Simulation Model of a Fuzzy Cyber Attack Detection System

Ihor Subach[1,2], Vitalii Fesokha[2], Artem Mykytiuk[1], Volodymyr Kubrak[1],
and Stanislav Korotayev[1]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", st.Verkhnoklyuchova, 4, Kyiv, 03056, Ukraine*
[2]*Military Institute of Telecommunications and Information Technologies named after Heroes of Kruty, st. Moscow, 45/1, Kyiv, 01011, Ukraine*

Abstract
The method of applying a simulation model of a fuzzy cyberattack detection system is considered. The functional diagram of the simulation model is given. The block diagram of the simulation model is considered and the purpose of its elements is described. The main steps of using a simulation model for conducting an experimental study of evaluating the effectiveness of models and methods for detecting cyber attacks based on the theory of fuzzy sets and fuzzy inference are described. The procedure for generating initial data is given, the classes of cyberattacks to be detected are defined, the vectors of cyberattack features are identified, the parameters of the studied traffic are described, the types of membership functions are defined to formalize expert knowledge and represent it in the knowledge base in the form of fuzzy production rules. The issue of parametric adaptation of membership functions to clarify the subjective judgments of experts are considered. To implement the possibility of detecting polymorphic cyberattacks, the procedure for determining the required number of the most important features for each known class of cyberattacks, represented by fuzzy sets and linguistic variables that characterize them quite fully, is described. A comparative analysis of the results of modeling the process of detecting cyber attacks based on the proposed approach with existing methods for detecting cyber attacks was carried out, based on the theory of fuzzy sets and fuzzy logic, artificial immune systems and neural networks in terms of accuracy.

Keywords
Cybersecurity, cyber attack, IDS, simulation model, fuzzy set theory

1. Introduction

In [1, 2] the architecture of a promising fuzzy intelligent system for detecting cyber attacks was proposed. It allows the security operations centers (SOC) operational personnel to make decisions on their detection promptly and reasonably. It is based on the technologies of data mining, machine learning, big data processing and artificial intelligence (Figure 1).

The proposed architecture consists of the following components [2-10]:

information collection subsystem - a set of sensors on network nodes that collect and process primary data on network activity;

analyzer - the module of the first echelon of cyber attack detection - analyzes security events (incidents) and on the basis of signature analysis classifies harmful activity as cyber attack;

signature database - a dictionary of signatures of classified cyberattacks used by the component - analyzer;

XXI International Scientific and Practical Conference "Information Technologies and Security" (ITS-2021), December 9, 2021, Kyiv, Ukraine
EMAIL: igor_subach@ukr.net (I. Subach); vitaliifesokha@gmail.com (V. Fesokha); mukuta8888@gmail.com (A. Mykytiuk); volodymyr.kubrak@ukr.net (V. Kubrak); meduha1998@gmail.com (S. Korotayev)
ORCID: 0000-0002-9344-713X (I. Subach); 0000-0001-6612-1970 (V. Fesokha); 0000-0002-8307-9978 (A. Mykytiuk); 0000-0001-8877-5289 (V. Kubrak); 0000-0003-3823-8375 (S. Korotayev)

statistical database - statistics of telemetry of network traffic for a certain period for its further use in order to improve the system to adapt the parameters of the mechanism of detection of cyber attacks to existing changes;

scheduler - plans further actions of the system after processing by the analyzer (in case it detects a cyberattack, the scheduler gives control to the response module to take measures to stop malicious activity; otherwise, the scheduler gives control to the data analysis module detection cyber attacks second tier, built on the basis of the developed model of cyberattack detection [8-11];

fasificationblock - clear values of the studied parameters are turned into fuzzy (the degree of their belonging to the term sets of linguistic variables specified by experts is determined). Fuzzy network activity is taken into account based on the application of the model proposed in [8-11];

fuzzy rule base: contains fuzzy production rules built by experts at the stage of preparing the system for operation;

adaptation module: performs primary parametric adjustment of membership functions by means of genetic algorithms on the basis of a certain training sample and further training of the system if necessary on the basis of statistical data to clarify the values of cyber attack detection mechanism;

fuzzy inference block - a module for making decisions about the state of the network based on determining the relationship between input data (telemetry of network traffic) and expert opinions by means of fuzzy logic;

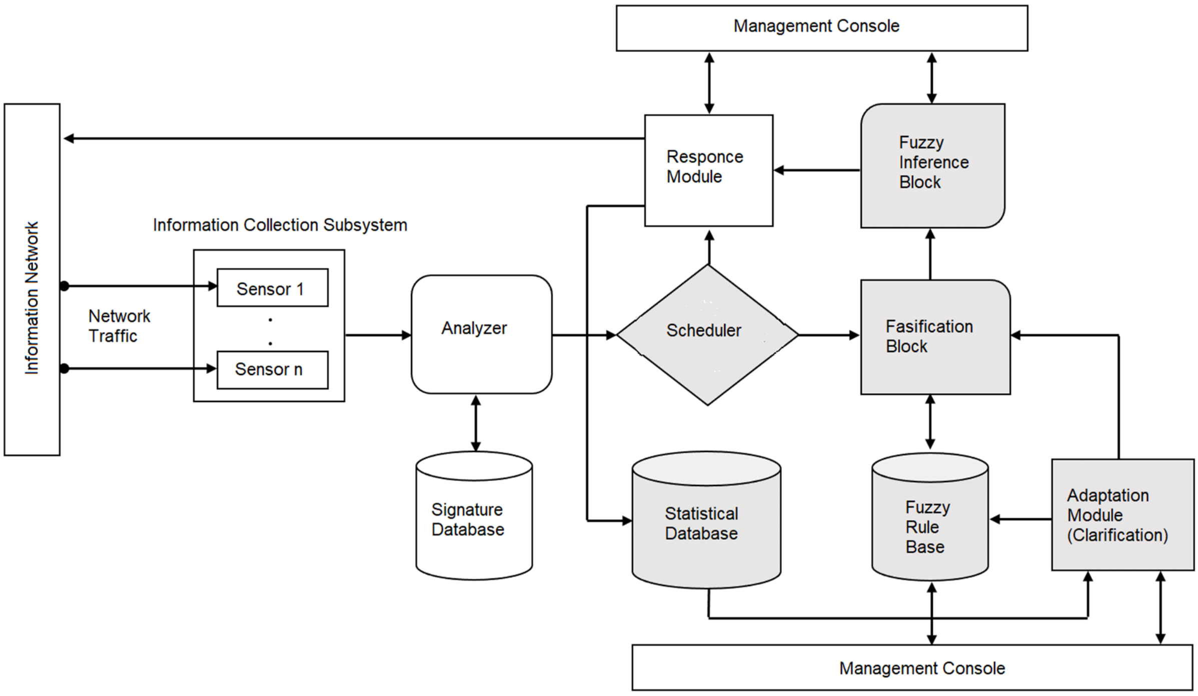


Figure 1. Architecture of a promising fuzzy intelligent system for detecting cyber attacks

response module - generates requests and notifications to the console, takes protective measures to block detected cyberattacks, as well as fills the database of statistical decisions on their decisions for further use by the adaptation module;

management console: module for configuring the cybersecurity officer of the system parameters.

The application of the proposed IDS architecture allows to increase the efficiency of detecting cyberattacks in near real time, based on the application of a multi-tiered approach to their detection. In addition, it becomes possible to adapt the system to the detection of unknown types of cyber attacks (zero day), as well as increase the efficiency of the cybersecurity officer on such indicators as efficiency and soundness of decision-making.

In [8-11] – models and methods for detecting known and polymorphic cyber attacks based on the theory of fuzzy sets and fuzzy inference. Formally, the task of fuzzy identification of a cyber attack is to find a solution to analytical expression that connects a set of parameters of the state of the system, on the basis of which its anomalous behavior is determined and an expert solution that meets them, taking into account the weighting coefficients for fuzzy rules (Figure 2):

$$X^* = \left(x_1^*, x_2^*, \ldots, x_n^*\right) \to y = \left(a_1^{jk_j}, a_2^{jk_j}, \ldots, a_n^{jk_j}\right) \in D = \left(d_1, d_2, \ldots d_n\right), i = \overline{1,n}, j = \overline{1,m}, \qquad (1)$$

where $X = \left(x_1, x_2, \ldots, x_n\right)$ is a set of parameters of the information and communication system (ICS) which are analyzed;

y is a linguistic description of the expert decision (opinion) $d_j \in D$ on the state of ICS;

jk_j is a numbers of combinations of values of x_i of the parameters of ICS state description, corresponding to the value of d_j .

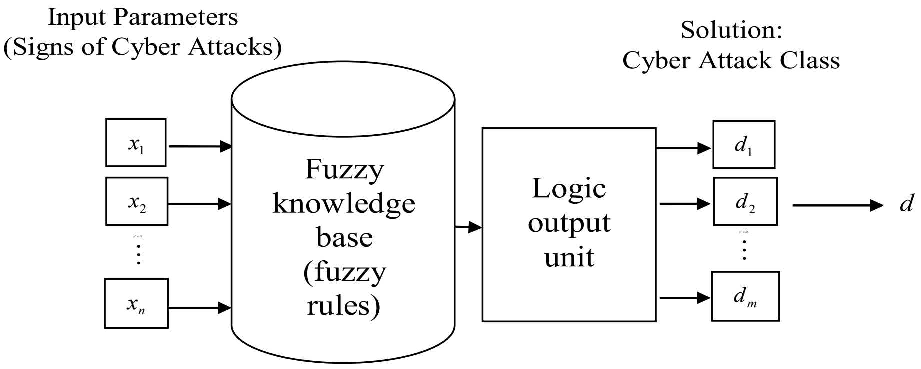


Figure 2. Graphical interpretation of the problem of cyber attack identification

To evaluate the effectiveness of the proposed solutions, a system simulation model (FIDM – Fuzzy Intrusion Detection Model) was developed using the Fuzzy Logic Toolbox™ package,which provides MATLAB® functions and the Simulink® block [13, 14, 15] for designing and modeling systems based on fuzzy logic.

2. Functional and structural diagram of the simulation model

The functional diagram of the simulation model is shown in Figure 3, where *x1..xn* are input parameters and *y* is an output variable.

The basis of the proposed simulation model is a modular diagram for organizing sequential iterative interaction between its components: a data input module for analysis, a fuzzification module, a knowledge base (KB), a fuzzy inference and defuzzification module [13, 14, 15]. The structure diagram of the developed simulation model is shown in Figure 4.

For the operation of the test data input module [16], is a set of statistical data on cyberattacks KDD Cup 1999 Data (xlsx files) was used.

The purpose of the fuzzification module is to represent the quantitative and qualitative values of the studied parameters using term sets and linguistic variables. Incomplete and uncertain data on network activity were taken into account by applying the developed model of cyber attack detection.

Additional rules generation module was used to create new fuzzy production rules in the knowledge base by intersecting fuzzy sets of linguistic variables of previously existing rules and the most significant linguistic variables predefined by an expert for each class of cyberattacks.

The knowledge base is a set of fuzzy production rules built by an expert.

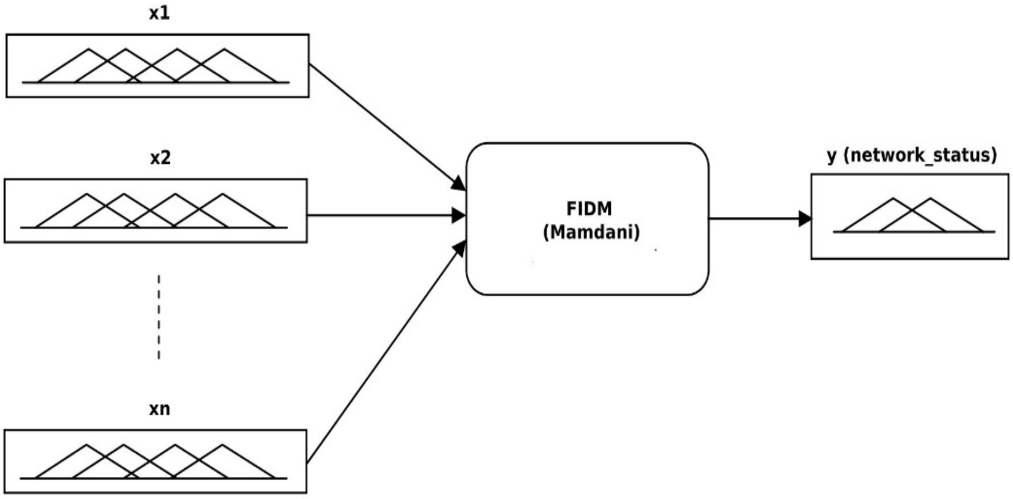


Figure 3. Functional diagram of the FIDM simulation model

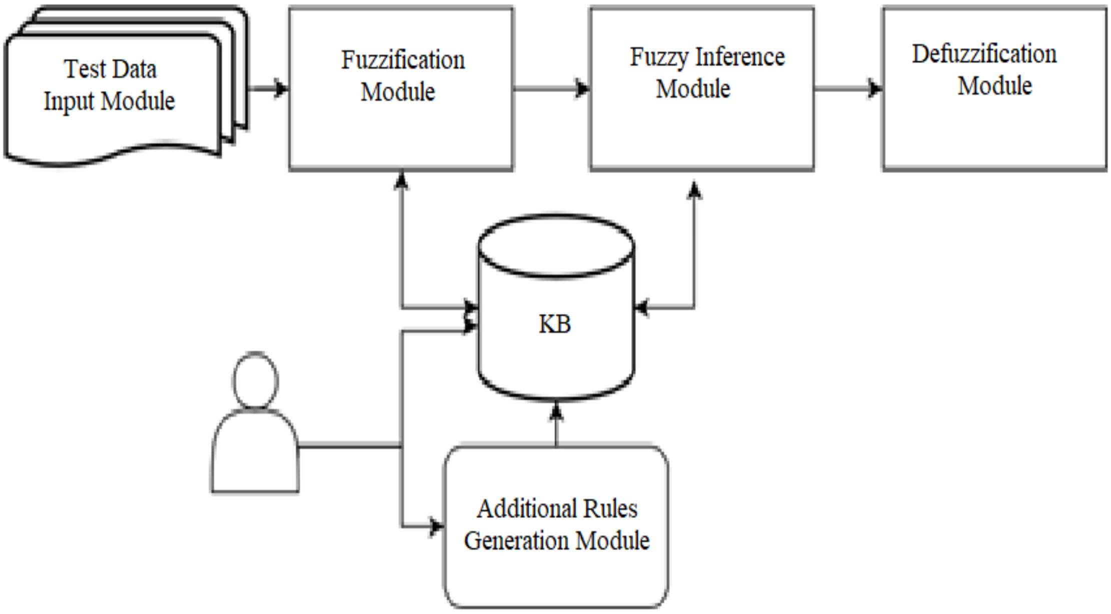


Figure 4. Structure diagram of the developed simulation model

The purpose of the fuzzy inference module is to generate a decision about the state of the information and communications network based on determining the relationship between input data and expert conclusions using fuzzy logic.

The defuzzification module was used to convert the obtained values of the fuzzy inference into crisp ones.

3. Methodology of applying the simulation model

Step 1: Defining is a set of cyber attackstatistics: KDD Cup 1999 Data [16].

Step 2. Defining the classes of cyberattacks to be detected [16]: Denial of Service, Remote to Local, User to Root, Probe and normal states of the ICS.

Step 3. The input of the simulation model was fed with vectors of cyber attacks of the KDD Cup 1999 Data set in the number of: known – Denial of Service – 4264, Remote to Local – 1020, User to

Root – 52, Probe – 3231; normal states of the information system – 1000; polymorphic cyberattacks built on the basis of known ones – 100. Thus, the total number of attribute vectors was 9667.

For the study, 38 parameters of network traffic telemetry were selected based on the classification proposed in the chosen data set on cyberattacks (Table 1).

Table 1
The studied parameters of network traffic

Code	Parameter	Description
x1	duration	Connection time in seconds
x2	src_bytes	Number of bytes from source to destination
x3	dst_bytes	Number of bytes in the response to the client
x4	land	1 if the connection is from/to the same host/port
x5	wrong_fragment	Number of false fragments
x6	urgent	Number of urgent packages
x7	hot	Number of hot indicators
x8	num_failed_logins	Number of failed registration attempts
x9	logged_in	1 if successful login; 0 unsuccessful
x10	num_compromised	Number of compromising conditions
x11	root_shell	1 if a root shell is obtained; otherwise 0
x12	su_attempted	1 if su root was executed; otherwise 0
x13	num_root	Number of root accesses
x14	num_file_creations	Number of file creation operations
x15	num_shells	Number of shell requests
x16	num_access_files	Number of operations to access file control
x17	num_outbound_cmds	Number of FTP session output commands
x18	is_host_login	1 if the login belonged to the hot list
x19	is_guest_login	1 if guest login
x20	count	Number of connections in the current session in the last 2 sec.
x21	srv_count	Number of connections to the same service in the last 2 sec.
x22	serror_rate	% of connections that had SYN errors
x23	srv_serror_rate	% of connection with an error in SYN packet
x24	rerror_rate	% of connections that had REJ errors
x25	srv_rerror_rate	% of connections with REJ errors
x26	same_srv_rate	% of connections having the same service
x27	diff_srv_rate	% of connections to different services
x28	srv_diff_host_rate	% connections from other hosts
x29	dst_host_count	Number of connections to the host established by the remote party

x30	dst_host_srv_count	Number of connections to the host established by the remote party that use one service
x31	dst_host_same_srv_rate	% of connections to the local host established by the remote party using one service
x32	dst_host_diff_srv_rate	% of connections to the local host established by the remote party using different services
x33	dst_host_same_src_port_rate	% of connections to the host at the current source port number
x34	dst_host_srv_diff_host_rate	% of connections to the service of different hosts
x35	dst_host_serror_rate	% of connections with SYN error for the destination host
x36	dst_host_srv_serror_rate	% of connections with SYN error for the receiver service
x37	dst_host_rerror_rate	% of connections with REJ error for the destination host
x38	dst_host_srv_rerror_rate	% of connections with REJ error for the receiver service

Step 4. Defining the type of membership functions to describe the ranges of values of the studied parameters and the power of term sets for input and output linguistic variables: triangular membership functions (2) due to the ability to undergo parametric adaptation (refinement) while maintaining an acceptable level of computational complexity [17-22], term set power – 7 (number: VS is acronym for "very small", S is acronym for "small", BA is acronym for "below average", A is acronym for "average", AA is acronym for "above average", L is acronym for "large", VL is acronym for "very large").

$$f_{\Delta}(x,a,b,c)=\left\{\begin{matrix} 0, x<a \\ \dfrac{x-a}{b-a}, a\le x\le b \\ \dfrac{c-x}{c-b}, b\le x\le c \\ 0, x>c \end{matrix}\right\}, \tag{2}$$

where *a, b, c* is a some numeric parameters that take arbitrary real values and are ordered by relations: $a \le b \le c$.

Step 5. Defining input and output linguistic variables: 38 input linguistic variables that correspond to the number of studied parameters of network traffic and one output – an indicator of the state of the information and communications system. Each input value (x_i) corresponds to the network traffic parameter according to the KDD Cup 1999 Data, and the membership functions configured by the expert are represented by term sets.

x_1 – {VS – very small [0, 250, 510], S –small [500, 1000, 1500], BA – below average [1400, 2000, 2500], A – average [4000, 5250, 6500], AA – above average [6400, 8500, 10500], L – large [10400, 15500, 20500], VL – very large [20000, 31000, 42500]} on the universe [0, 42500];

x_2 – {VS – very small [0, 250, 550], S –small [520, 1000, 1500], BA – below average [1400, 5500, 10500], A – average [10000, 12500, 25500], AA – above average [25000, 40000, 54550], L – large [2500000, 77000000, 150000000], VL – very large [120000000, 350000000, 700000000]} on the universe [0, 700000000];

x_3 – {S –small [0, 500, 1000], BA – below average [1000, 5000, 9999], A – average [10000, 60000, 100000], AA – above average [125000, 600000, 1000000], L – large [1100000, 1800000, 2500000], VL – very large [2400000, 3500000, 5250000]} on the universe [0, 5250000];

$x_4, x_9, x_{11}, x_{12,}$ $x_{17}, x_{18}, x_{19}, x_{24}, x_{25}$ – {S –small [0, 0.25, 0.5], L – large [0.5, 1, 1.5]} on the universe [0, 1.5];

x_5 – {S –small [0, 0.25, 0.5], A – average [0.5, 1, 1.5], L – large [2.5, 3, 3.5]} on the universe [0, 3.5];

x_6 x_{15}, x_{16} – {S –small [0, 0.25, 0.5], A – average [0.5, 1, 1.5], L – large [1.5, 2, 2.5]} on the universe [0, 3];

x_7 – {S –small [0, 5, 10], A – average [21, 25, 30], L – large [10, 15, 22]} on the universe [0, 30];

x_8 – {S –small [0, 0.25, 0.5], A – average [0.5, 1, 1.5], L – large [4, 5, 6]} on the universe [0, 6];

x_{10} – {S –small [0, 5, 10], L – large [15, 27.5, 40]} on the universe [0, 40];

x_{13} – {S –small [0, 5, 10], A – average [10, 15, 20], L – large [30, 42.5, 55]} on the universe [0, 55];

x_{14} – {S –small [0,2,5], L – large [20,22.5,25]} on the universe [0,25];

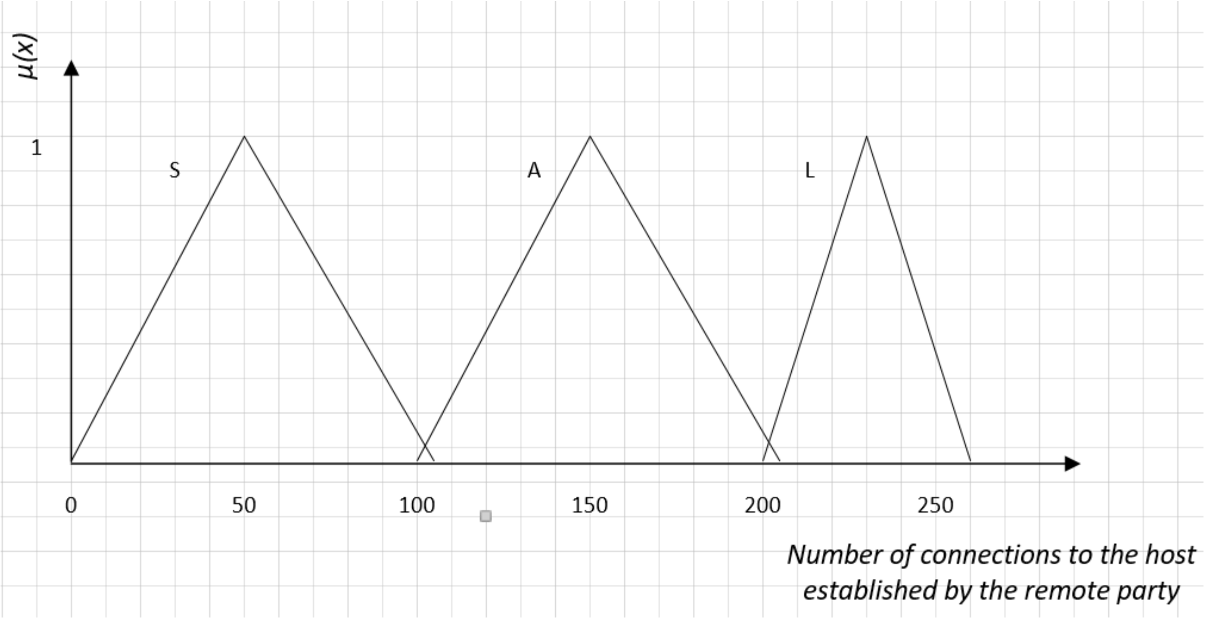


Figure 5. Graphical representation of the described linguistic terms of membership function

x_{15}, x_{16} – {S –small [0, 50, 101], A – average [99, 200, 305], L – large [300, 400, 515]} on the universe [0, 515];

$x_{22}, x_{23}, x_{26}, x_{27}, x_{28}$ – {S –small [0, 0.25, 0.6], L – large [0.5, 0.8, 1.2]} on the universe [0, 1.2];

x_{29}, x_{30} – {S –small [0, 50, 105], A – average [100, 150, 205], L – large [200, 230, 260]} on the universe [0, 260];

$x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}, x_{37}, x_{38}$ – {S –small [0, 0.25, 0.6], L – large [0.5, 0.8, 1.2]} on the universe [0, 1.2].

Step 6. Preparing test data format of KDD Cup 1999 Data for the Fuzzy Logic Toolbox™ software.

Step 7. Creating fuzzy production rules for the KB based on the KDD Cup 1999 Data set using association rule search algorithms (Fig. 6) [23, 24].

Step 8. Obtaining expert conclusions about the state of IP according to the classification of cyber attacks presented in the KDD Cup 1999 Data: Denial of Service, Remote to Local, User to Root, Probe and normal state: total number of rules: 335, of which Denial of Service – 68; Remote to Local – 55; User to Root – 18; Probe – 107; Normal – 87.

Step 9. Parametric adaptation of constructed membership functions [25, 26, 27] in order to clarify the subjective point of view of the expert by means of the Optimization Tool package of MATLAB® software [13, 14, 15]: on the basis of the frequent data sets found at the previous stage, parametric optimization of membership functions was performed for the above terms of each studied variable (search for the optimum of the parameter vector of the system of equations of the analytical model of the triangular membership function).

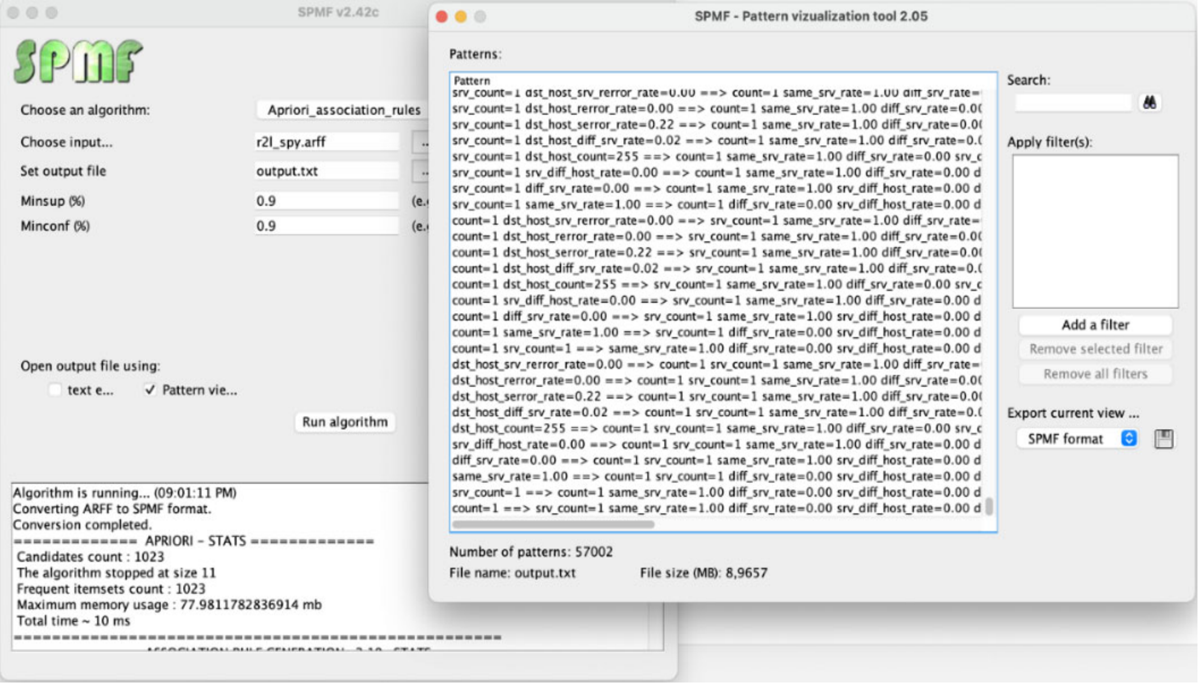
Step 10. Determining the required number of the most important (informative) parameters (features) for each known class of cyberattacks, represented as fuzzy sets of linguistic variables that characterize them quite fully in order to be able to identify polymorphic modifications of known cyberattacks:

$$X_{dos} = \{x_1, x_3, x_5, x_{20}, x_{21}, x_{22}, x_{23}, x_{26}, x_{28}, x_{29}, x_{30}, x_{31}, x_{32}, x_{33}\};$$

$$X_{r2l} = \{x_1, x_2, x_3, x_7, x_9, x_{20}, x_{21}, x_{26}, x_{28}, x_{29}, x_{30}, x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}\};$$

$$X_{u2r} = \{x_2, x_3, x_9, x_{26}, x_{27}, x_{29}, x_{31}, x_{32}, x_{33}, x_{37}, x_{38}\};$$

$$X_{probe} = \{x_{20}, x_{21}, x_{24}, x_{25}, x_{26}, x_{27}, x_{29}, x_{30}, x_{32}, x_{37}, x_{38}\}.$$



Choose an algorithm: Apriori_association_rules
Choose input... r2l_spy.arff
Set output file output.txt
Minsup (%) 0.9
Minconf (%) 0.9

Open output file using:
□ text e... ✓ Pattern vie...

```
Algorithm is running... (09:01:11 PM)
Converting ARFF to SPMF format.
Conversion completed.
============= APRIORI - STATS =============
 Candidates count : 1023
 The algorithm stopped at size 11
 Frequent itemsets count : 1023
 Maximum memory usage : 77.9811782836914 mb
 Total time ~ 10 ms
================================================
```

Patterns:

```
Pattern
srv_count=1 dst_host_srv_rerror_rate=0.00 ==> count=1 same_srv_rate=1.00 diff_srv_rate=
srv_count=1 dst_host_rerror_rate=0.00 ==> count=1 same_srv_rate=1.00 diff_srv_rate=0.0(
srv_count=1 dst_host_serror_rate=0.22 ==> count=1 same_srv_rate=1.00 diff_srv_rate=0.0(
srv_count=1 dst_host_diff_srv_rate=0.02 ==> count=1 same_srv_rate=1.00 diff_srv_rate=0.(
srv_count=1 dst_host_count=255 ==> count=1 same_srv_rate=1.00 diff_srv_rate=0.00 srv_c
srv_count=1 srv_diff_host_rate=0.00 ==> count=1 same_srv_rate=1.00 diff_srv_rate=0.00 d
srv_count=1 diff_srv_rate=0.00 ==> count=1 same_srv_rate=1.00 srv_diff_host_rate=0.00 d
srv_count=1 same_srv_rate=1.00 ==> count=1 diff_srv_rate=0.00 srv_diff_host_rate=0.00 d
count=1 dst_host_srv_rerror_rate=0.00 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=
count=1 dst_host_rerror_rate=0.00 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.0(
count=1 dst_host_serror_rate=0.22 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.0(
count=1 dst_host_diff_srv_rate=0.02 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.(
count=1 dst_host_count=255 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.00 srv_c
count=1 srv_diff_host_rate=0.00 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.00 d
count=1 diff_srv_rate=0.00 ==> srv_count=1 same_srv_rate=1.00 srv_diff_host_rate=0.00 d
count=1 same_srv_rate=1.00 ==> srv_count=1 diff_srv_rate=0.00 srv_diff_host_rate=0.00 d
count=1 srv_count=1 ==> same_srv_rate=1.00 diff_srv_rate=0.00 srv_diff_host_rate=0.00 d
dst_host_srv_rerror_rate=0.00 ==> count=1 srv_count=1 same_srv_rate=1.00 diff_srv_rate=
dst_host_rerror_rate=0.00 ==> count=1 srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.0(
dst_host_serror_rate=0.22 ==> count=1 srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.0(
dst_host_diff_srv_rate=0.02 ==> count=1 srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.(
dst_host_count=255 ==> count=1 srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.00 srv_c
srv_diff_host_rate=0.00 ==> count=1 srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.00 d
diff_srv_rate=0.00 ==> count=1 srv_count=1 same_srv_rate=1.00 srv_diff_host_rate=0.00 d
same_srv_rate=1.00 ==> count=1 srv_count=1 diff_srv_rate=0.00 srv_diff_host_rate=0.00 d
srv_count=1 ==> count=1 same_srv_rate=1.00 diff_srv_rate=0.00 srv_diff_host_rate=0.00 d
count=1 ==> srv_count=1 same_srv_rate=1.00 diff_srv_rate=0.00 srv_diff_host_rate=0.00 d
```

Number of patterns: 57002
File name: output.txt File size (MB): 8,9657

Figure 6. Associative rules search box in Open-Source Data Mining Library SPMF

Step 11. Obtaining additional fuzzy production rules for KB based on the actions taken in the previous step and removing duplicate rules. As a result, new rules were obtained in the following quantity: Denial of Service – 48; Remote to Local – 14; User to Root – 13; Probe – 16. The total number of rules in the KB – 426.

Step 12. Application of the developed program code for the correct input of data from the cyberattack data set for further analysis by the Fuzzy Logic Toolbox™ library.

Table 2.
Comparative analysis of simulation results

Cyber attack class	Immune systems	Neural networks	Fuzzy logic	Suggested method
DoS	0,98	1,0	0,94	1,0

R2L	0,90	0,95	0,99	0,99
U2L	0,97	0,36	0,99	0,99
Probe	0,96	0,99	0,90	1,0
Normal	0,97	0,99	0,91	0,99
Polymorphic (DoS)	-	-	-	1,0
Polymorphic (R2l)	-	-	-	0,98
Polymorphic (U2r)	-	-	-	1,0
Polymorphic (Probe)	-	-	-	0,98

Step 13. Conducting experimental studies on cyber attack detection by a developed simulation model, the functioning of which is based on the application of models and methods and comparative analysis of the results of modeling the process of detecting cyber attacks based on the proposed approach with existing methods for detecting cyber attacks: based on the theory of fuzzy sets and fuzzy logic, artificial immune systems and neural networks in terms of accuracy.

$$Accuracy = \frac{TD}{TD + FN}, \qquad (3)$$

where TD (True Detection) is the number of correctly detected cyber attacks;
FN (False Negative) is a type II errors (classifying a cyber attack as a normal state).

A comparative analysis of the results of modeling the process of cyber attack detection based on the approach proposed in the study and existing solutions in terms of accuracy are presented in Table 2.

The results of the study were included in the methodology of rational choice of security incident management system for building operational security center [28].

4. Conclusion

The practical application of the developed simulation model of a fuzzy cyber attack detection system showed the expediency of using it to evaluate models and methods of cyber attack detection based on the theory of fuzzy sets and fuzzy inference. Thus, the comparison of the developed scientific and methodological apparatus with the already available ones shows that its use makes it possible to increase the effectiveness of information systems cyber protection in terms of the accuracy of detecting known cyber attacks by an average of 10%, as well as to ensure the detection of polymorphic cyber attacks in terms of accuracy of at least 98 %.

References

[1] I. Subach, V. Kubrak, A. Mykytiuk, Architecture and functional model of a promising proactive intelligent system SIEM-system for cyber protection of critical infrastructure, Information Technology and Security Vol. 7 Iss. 2 (2019) 208-215. doi:10.20535/2411-1031.2019.7.2.190570.

[2] I. Subach, V. Fesokha, N. Fesokha, Analysis of existing intrusion prevention solutions in information and telecommunication networks, opened on the basis of publicly available licenses, Information Technology and Security Vol. 5 Iss. 1 (2017) 29–41. doi:10.20535/2411-1031.2017.5.1.120554.

[3] IDS architecture, 2015. URL: https://studfile.net/preview/1665659/page:28.

[4] Intrusion detection system architecture, 2020 URL: https://studref.com/521846/informatika/arhitektura_sistem_obnaruzheniya_vtorzheniy.

[5] D. Levonevsky, R. Fatkieva, Development of a system for detecting network traffic anomalies, Scientific Bulletin of NGTU Vol. 56 Iss. 3 (2014) 108-114.

[6] J. Rabatel, S. Bringay, P. Poncelet, Fuzzy anomaly detection in monitoring sensor data: IEEE International Conference on Fuzzy Systems, Inc., 2010.

[7] M. Dodonov, N. Dodonova, Automated detection system of insider attacks using fuzzy logic: Information Technology and Nanotechnology (ITNT-2015), Inc., 2015, pp. 376–380.

[8] I. Subach, I. Subach, V. Kubrak, A. Mykytiuk, S. Korotaiev, Zero-day polymorphic cyberattacks detection using fuzzy inference system, Austrian Journal of Technical and Natural Sciences Vol. 5-6, (2020) 8-13. doi:10.29013/AJT-20-5.6-8-13.

[9] I. Subach, Y. Zdorenko, V. Fesokha. Methods of detecting JS(HTML)/Scrinjectcyber attacks based on the application of the mathematical apparatus of fuzzy set theory, Proceedings of the Heroes of Kruty Military Institute of Telecommunications and Informatization Iss. 4 (2018) 125–131.

[10] I. Subach, V. Fesokha., A model for detecting cyber attacks on information and telecommunication systems based on the description of anomalies in their operation by weighted fuzzy rules, Information Technology and Security Vol. 5. Iss. 2 (2017) 145–152. doi:10.20535/2411-1031.2017.5.2.136984.

[11] I. Subach, V. Kubrak, A. Mykytiuk, S. Korotaev, Rule-oriented method of cyber incidents detection by SIEM based on fuzzy logical inference: CEUR Workshop Proceedings (CEUR-WS.org), Vol. 2859, 2021, pp. 210-219.

[12] M. Beshley, S. Toliupa, V. Pashkevych, R. Kolodiy, Development of software system for network traffic analysis and intrusion detection: International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo, Inc., 2018.

[13] Matlab documentation. Exponent, 2020 URL: https://docs.exponenta.ru/matlab.

[14] S. Sivanandam, S. Sumathi, S. Deepa, Introduction to Fuzzy Logic using MATLAB: Springer-Verlag Berlin Heidelberg, Inc., 2007.

[15] A.V. Leonenkov, Fuzzy modeling in MATLAB and fuzzyTECH, St. Petersburg, BHV-Petersburg, 2003.

[16] UCI Knowledge Discovery in Databases Archive. University of California, Irvine, CA 92697–3425. KDD Archive, 2020 URL: http://kdd.ics.uci.edu/databases/kddcup99/task.html.

[17] A. Rothstein, Intelligent identification technologies: fuzzy sets, genetic algorithms, neural networks, Vinnytsia: UNIVERSUM, 1999.

[18] S. Shtovba, Fuzzy model tuning based on a training set with fuzzy model output values, Cybernetics and Systems Analysis Vol. 43 (2007) 334–340.

[19] A. Rothstein, Medical diagnostics on fuzzy logic, Vinnytsia: Continent-PRIM, 1996.

[20] Y. Mityushkin, B. Mokin, O. Rothstein, Soft Computing: identification of patterns of fuzzy knowledge bases: a monograph. Vinnytsia: UNIVERSUM-Vinnytsia, 2002.

[21] O. Rothstein, G. Chernovolyk, E. Laryushkin, Method of constructing membership functions of fuzzy sets. Bulletin of VPI, Vol. 3 (1996) 72–75.

[22] A. Piegat, Fuzzy Modeling and Control, Physica-Verlag, Heidelberg. 2001.

[23] SPMF, 2022 URL: http://www.philippe-fournier-viger.com/spmf/index.php.

[24] P. Fournier-Viger, A. Gomariz, T. Gueniche, A. Soltani, C. Wu, V. S. Tseng, SPMF: a Java Open-Source Pattern Mining Library, Journal of Machine Learning Research Vol. 1 (2014) 1-5.

[25] Y. Zaichenko, Operations Research: Fuzzy Optimization: Kiev, High school, 1991.

[26] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. 1989.

[27] F. Herrera, M. Lozano, Adaptation of genetic algorithm parameters based on fuzzy logic controllers. Genetic Algorithms and Soft Computing: Physica-Verlag, Heidelberg. 1996. pp. 95-124.

[28] I. Subach, V. Kubrak, A. Mykytiuk, Methodology of rational choice of security incident management system for building operational security center: CEUR Workshop Proceedings (CEUR-WS.org), Vol. 2577, 2019, pp. 11-20.