УДК 004.056

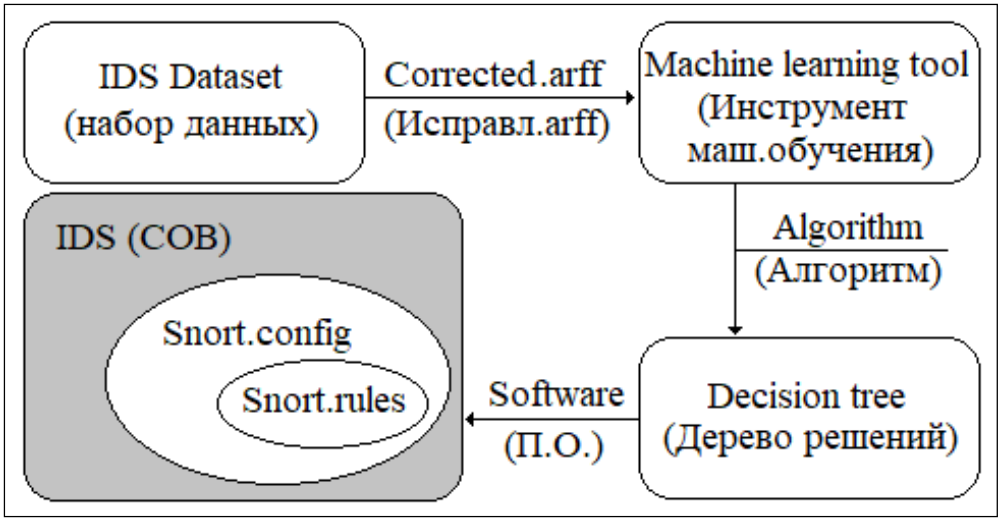
REALIZATION OF EXPERT INTRUSION DETECTION SYSTEM BASED ON THE RESULTS OF DATASETS AND MACHINE LEARNING ALGORITHM ANALYSIS

The article was received by the editorial board on 05.12.2019, in the final version – 06.05.2020.

Ivkin Andrey N., Samara National Research University, 34, Moskovskoye shosse, Samara, 443086, Russian Federation,
postgraduate student, e-mail: Ivkin.92@bk.ru
Burlakov Michael E., Samara National Research University, 34, Moskovskoye shosse, Samara, 443086, Russian Federation,
Cand. Sci. (Engineering), Associate Professor, e-mail: knownwhat @gmail.com

Intrusion detection system is one of the most important devices for the protection of computing systems. The system is enabled to detect and investigate packets of network traffic. IDS Snort is an open source with free software that is used to protect your network. Snort detects only confirmed attacks using predefined signatures. In order to detect new, previously unknown network attacks and reduce false positives, this work has developed advanced rules for Snort, obtained using the WEKA machine learning tool and the j48 algorithm. In the article, for experimental research, the CICIDS dataset is used. The main goal of this research is the realization of IDS with embedded machine learning tool rules. The main stages of research are comparative analysis of different publicly available datasets, data preparation, application and comparison of 8 different algorithms, extraction of expert rules, implementation of Snort rules and attacks identification. The proposed system provides effective detection rates.
Keywords: intrusion detection system, Snort, machine learning, WEKA, j48 algorithm, CICIDS dataset, signatures

Graphical annotation (Графическая аннотация)



РЕАЛИЗАЦИЯ ЭКСПЕРТНОЙ СИСТЕМЫ ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ, ОСНОВАННОЙ НА РЕЗУЛЬТАТАХ АНАЛИЗА НАБОРОВ ДАННЫХ И АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ

Статья получена редакцией 05.12.2019, в окончательном варианте – 06.05.2020.

Ивкин Андрей Николаевич, Самарский национальный исследовательский университет имени академика С. П. Королева, 443086, Российская Федерация, г. Самара, ул. Московское шоссе, 34,
аспирант, e-mail: Ivkin.92@bk.ru
Бурлаков Михаил Евгеньевич, Самарский национальный исследовательский университет имени академика С. П. Королева, 443086, Российская Федерация, г. Самара, ул. Московское шоссе, 34,
кандидат технических наук, доцент кафедры «Безопасности информационных систем», e-mail: knownwhat@gmail.com

Система обнаружения вторжений выявляет факт применения различного вредоносного программного обеспечения, всевозможных аномальных действий, наносящих ущерб целостности и безопасности компьютерной системы. Главной задачей системы обнаружения вторжений является корректное и своевременное оповещение об атаках на информационную систему. Ключевым моментом успешного обнаружения вторжений является выбор набора правил для детектирования угроз. С целью улучшения эффективности работы СОВ в данной исследовательской работе предлагается использовать экспертные правила, полученные из инструмента машинного обучения, что также даст возможность детектировать атаки, не представленные в базах сигнатур. Для реализации системы обнаружения вторжений, основанной на экспертных правилах, необходим корректный набор данных. Существует целый ряд наборов данных, таких как KDD99, ISC2012, ADFA2013 и т.д., которые используются для оценки эффективности предлагаемых методов обнаружения и предотвращения вторжений. Большинство наборов данных содержат устаревшую информацию, недостаточно разнообразный трафик, однотипные атаки, сильно урезанную информацию о пакетах, также имеет место нехватка некоторых атрибутов. В статье проведен сравнительный анализ 15-ти общедоступных наборов данных для поддержки разработки/тестирования СОВ. Также оценивается эффективность применения 8-ми различных алгоритмов машинного обучения к набору данных CICIDS. Мерами оценки эффективности рассмотренных алгоритмов выступает следующее: точность, полнота, Fмера. Предлагаемая система обнаружения вторжений, основанная на системе обнаружении Snort, обеспечивает высокие (свыше 98 %) показатели обнаружения вторжений.

Ключевые слова: система обнаружения вторжений, наборы данных, машинное обучение, *CICIDS, WEKA, Snort,* сигнатуры

Introduction. There are many means of information protection such as firewall, intrusion detection systems (IDS), antivirus systems etc. Each of them is directed to the defense from specific threat. Intrusion detection plays a vital role in the network defense process by aiming security administrators in forewarning them about malicious behaviors such as intrusions, attacks, and malware. IDS is a specialized software and hardware tool designed to detect unauthorized access attempts to system resources [6], which will not allow an attacker to disrupt the system, steal confidential information, delete or modify data and so on. IDS can detect attacks that the firewall missed, because firewall restricts certain traffic to a host or subnet to prevent intrusions and does not monitor intrusions from within the network. IDS passes traffic, analyzing it and signaling when it detects suspicious activity. Security breach detection is usually done using heuristic rules and signature analysis of known computer attacks.

According to the approach to detection, IDS are classified as systems based on signature and behavioral analysis. The last of this is based on models of the normal functioning of the information system, while signature analysis is based on clearly defined tabular values of signatures [10]. IDS based on behavioral analysis [3] include: machine learning systems, anomaly-based systems and based on the detection of violations in the protocol. The approach proposed in this paper based on the expert rules of a machine learning tool and has the following advantages:

- a small number of false positives (number of incorrectly defined attacks);
- increased attack detection speed (quick comparisons with submitted rules);
- low resource costs (high capacity computing are not required);
- the ability to detect unknown attacks (applied machine learning methods will identify modified attacks).

The basis for the implementation of the proposed method was the SNORT system. IDS «Snort» [7] is open source software, which allows creating your own system, implement machine learning methods, apply borrowed code in the firewall and etc. As a rule, Snort is deploying on the router as a network IDS. Snort detects attacks based on rules written in a defined format and syntax. Snort is a multi variant packet investigation tool which works in multiple modes. The main Snort advantage over analogues is the flexibility and simple of modifying rules compared to other commercial IDS. The Snort architecture is represented in figure 1.

Snort rules are recorded in one line. They consist of a header and options, as shown in the example below (fig. 2). The rule header contains the rule actions, protocol name, IP addresses and port numbers. Rule options include rule execution criteria and additional responsive actions; they are used to implement more stringent traffic filtering. Rule options are optional; more information is provided in [7]. An example of a simple Snort rule which determines the presence of a SYN flag is shown in figure 2. This Flag, in the header of the TCP segment, used for synchronization of session numbers of data. SYN flag is used for SYN-flood attacks (form of DOS attack)

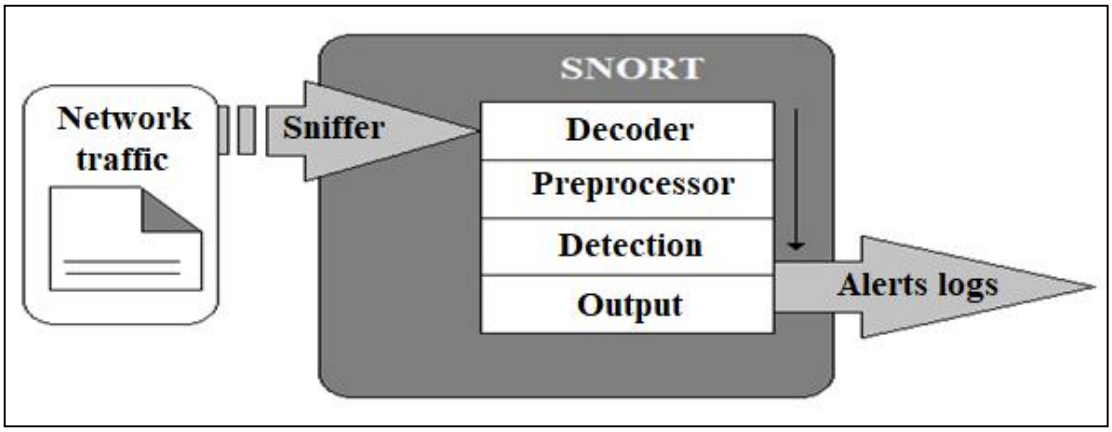


Figure 1 – Architecture of IDS Snort

Rule Header	*Rule Options*
alert tcp any any -> $MY_NET any	**(flags: S; msg: "SYN packet";)**

Figure 2 – Example of a Snort rule

The action "alert" is selected on the presented rule, which generates an alert and sends information to the logging system. According to the rule, an alert will be generated if packets with the presence of SYN flag (flags: S) are sent to our network ($MY_NET any – IP of our computer and all ports) from any source (any – all IP, any – all ports), using the TCP transfer protocol (tcp). In the logging system, the message presented in the rule will be recorded (msg: «SYN packet»).

The purpose of this research is to implement adaptive intrusion detection system based on expert rules obtained from the optimal dataset using the most suitable machine learning algorithm.

Comparative analysis of data sets for IDS. For testing and evaluating the effectiveness of various methods used in IDS, a correct dataset with real traffic and modern network attacks is required. A large number of datasets cannot be distributed in the public domain, because they contain confidential information. The other part of the datasets contains a lack of traffic and a variety of attacks. It should also be noted that good datasets should be updated periodically. According to research [1, 2, 11] proposed many ways to evaluate quality of datasets for IDS. In this paper we use the most modern evaluation system [2]. This system consists of 11 criteria: «Attack Diversity», «Anonymity», «Available Protocols», «Complete Capture», «Complete Interaction», «Complete Network Configuration», «Complete Traffic», «Feature Set», «Heterogeneity», «Labelling», and «Metadata» are critical for a comprehensive and valid IDS dataset.

Further, the disadvantages of each of the considered datasets are demonstrated:

- DARPA (Lincoln Laboratory 1998–99). DARPA is the first dataset for the evaluation of intrusion detection system. Within two weeks, about 201 cases of about 56 types of attacks were distributed. This set does not correspond to the real modern network traffic and contains various errors. Dataset is outdated for effective evaluation of IDSs;
- KDD'99 (University of California, Irvine 1998–99). KDD'99 is the subset of DARPA dataset. KDDCup dataset contains about 4,9 million single instances which are described by 41 features. They are classified as either normal or an intrusion. Also outdated dataset. Contains a large number of unnecessary records, as well as corrupted data;
- DEFCON (The Shmoo Group 2000, 2002). DEFCON datasets uses port scan and sweeps, bad packets, administrative privilege, and FTP by telnet protocol attacks. The dataset was generated during an open information security competition, which leads to a mismatch with real traffic;
- CAIDA (Center of Applied Internet Data Analysis 2002–2016). Contains 3 different datasets: CAIDA OC48, CAIDA DDOS and CAIDA Internet 2016. These datasets are very specific for machine learning and contain anonymous information, which makes them not effective in comparative analysis;
- LBNL (Lawrence Berkeley National Laboratory 2015). Network traffic with full packet headers recorded on an average site. There is no payload and too much anonymous information;
- CDX (United States Military Academy 2009). This dataset contains network traffic from cybersecurity competitions. The set contains Internet traffic, email traffic, DNS lookup and other necessary services.

Network attack tools such as Nikto, Nessus, and WebScarab were also used. Insufficient traffic diversity (quite common problem);

- Kyoto (Kyoto University 2009). The Kyoto 2006 dataset was created from November 2006 to August 2009. This dataset contains 50 million regular sessions and 43 million attack sessions. It does not coincide with the actual network traffic. The normal data in this dataset contains only DNS and mail traffic. Thus, the total number of false positives is reduced;
- Twente (University of Twente 2009). This dataset contains three services such as OpenSSH, Apache web server and Proftp using auth/ident on port 113 and captured data from a honeypot network by Netflow. The dataset contains unmarked traffic. Also contains small amount of total traffic and monotonous network attacks;
- UMASS (University of Massachusetts 2011). The dataset includes trace files, which are network packets, and some traces in wireless applications (from Massachusetts, Amherst, 2011) (Nehinbe, 2011). UMASS was created using one attack scenario; therefore, there is no diverse traffic, which makes it useless for testing IDS methods;
- ISCX2012 (University of New Brunswick 2012). A set with two profiles: a profile of various attack scenarios, and a profile for generating benign traffic. It includes network traffic for the HTTP, SMTP, SSH, IMAP, POP3, and FTP protocols with packet payloads. Important modern network protocols are not presented in ISCX;
- HTTPCSIC2010 (Spanish Research National Council 2010). The HTTP dataset CSIC 2010 contains 223 thousands of web requests automatically generated with 18 attributes. HTTP is a narrowly specialized data set with a lack of total information, network attacks, and the types of protocols;
- ADFA (University of New SouthWales 2013). This dataset includes normal training and validating data and 10 attacks per vector (Creech and Hu, 2013). It contains FTP and SSH password brute force, Java based Meterpreter, Add new Superuser, Linux Meterpreter payload and C100Webshel attacks. ADFA contains mislabeled data and insufficient network attacks.

According to a comparative analysis of this work (as shown in table 1) CICIDS selected for further research, because it meets all criteria of the standard dataset [2].

Table 1 – Results of comparative analysis

	Attacks	\Protocols	Features	Heterogeneity	Net. Config.	Interaction	Traffic	Anonymity	Labelling	Metadata	Capture	Result
DARPA	-	-	-	-	+	+	-	-	+	+	+	**5/11**
KDD	-	-	+	-	+	+	-	-	+	+	+	**6/11**
Kyoto	+	+	+	-	+	+	-	-	-	+	+	**7/11**
CDX	-	-	-	-	-	+	-	-	-	-	+	**2/11**
LBNL	-	-	-	-	+	-	+	+	-	-	-	**3/11**
CAIDA	-	-	-	-	+	-	+	+	-	+	-	**4/11**
DEFCON	-	-	-	-	-	+	-	-	-	-	+	**2/11**
UMASS	-	-	-	-	+	-	-	-	+	-	+	**3/11**
Twente	-	-	-	-	+	+	+	-	+	+	+	**6/11**
ISCX	+	-	-	+	+	+	-	-	+	+	+	**7/11**
ADFA	-	-	-	-	+	+	+	-	+	+	+	**6/11**
HTTPCSIC	-	-	+	-	+	+	-	-	+	-	+	**5/11**
CICIDS	+	+	+	+	+	+	+	+	+	+	+	**11!**

CICIDS dataset analysis. This dataset [8] contains the most common attacks and protocols. First, duplicated information is removed from the dataset, such as «Fwd Packet Header». Extra spaces and other input errors are also removed. Then, using the implemented software, the dataset is reformatted into a format compatible with the tool WEKA (.arff format). This format has a clear structure that is implemented in the CICIDS dataset. Presented in a tabular format, the set has 79 different attributes for each package (such as Destination port, Flow duration, Total fwd packets, Total backward packets and so on [9]). To generate traffic, two networks were emulated, an attacker network and a victim network as shown in figure 3.

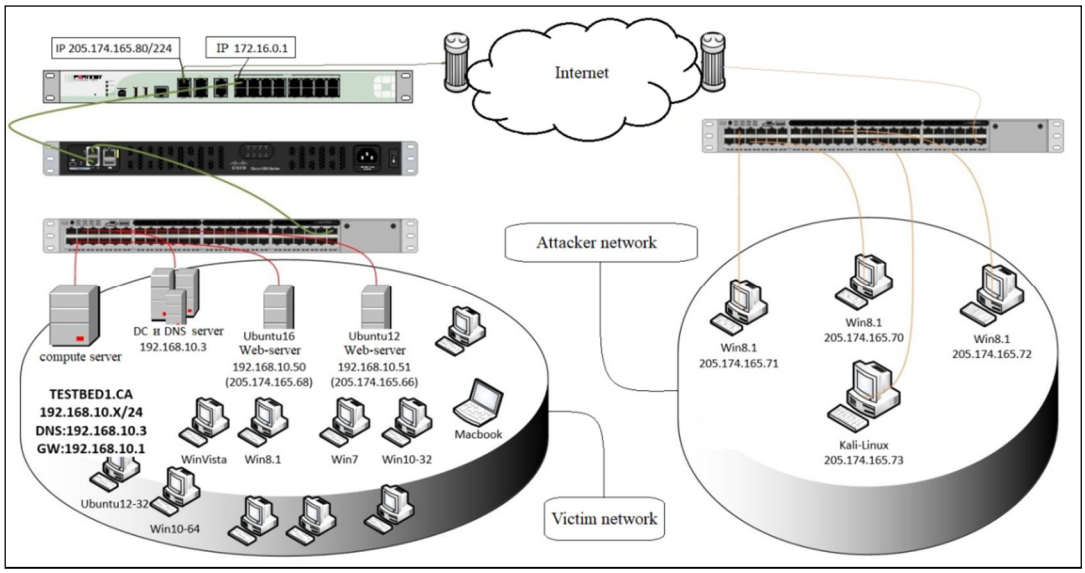


Figure 3 – CICIDS traffic generation architecture

It is worth noting that the heterogeneity of the set is achieved by capturing the network traffic from the main switch, and records of all system alerts and memory dumps of victim network computers. The network topology includes a modem, a firewall, switches, routers, and the presence of different operating systems such as Windows, Ubuntu and Macintosh. This dataset used software to emulate the real network user's behavior. The traffic generation lasted five days, and contains the following data as shown in table 2.

Table 2 – Daily CICIDS dataset traffic

Day	Traffic
Monday	Benign
Tuesday	Benign, BForce: (SFTP and SSH)
Wednesday	Benign, DoS and Hearbleed Attacks: Slowloris, Slowhttptest, Hulk and GoldenEye
Thursday	Benign, Web and Infiltration Attacks: Web BForce, XSS and Sql-injection Infiltration Dropbox Download and Cool disk
Friday	Benign, DDoS LOIT, Botnet ARES, Infiltration Attacks: PortScan

In this table «Benign» is the normal user behavior. «BForce» (Brute Force) is basically a hit and try attack, then the victim succeeds. «DoS» (Denial of Service attacks) – this type of attack creates difficult conditions for user access or a complete denial of service (for example, by creating a large number of requests). «Hearbleed Attacks» – attacks based on a bug in the OpenSSL cryptographic library, which is a widely used implementation of the TLS protocol. «Web» attacks on web systems, for example, embedding malicious code in a web page issued by a web system. «Infiltration attacks» probe attack which receiving information about the target system. «Botnet ARES» computer network consisting of a number of hosts with bots running that can provide remote shell, file upload/download, capturing screenshots and key logging.

Analysis of machine learning algorithms. One of the goals of this work was to choose the best machine learning algorithm and create expert rules for IDS based on it. To achieve the goal we need a machine learning tool and evaluation measures.

A large number of excellent tools available today, such as RapidMiner, Apache Mahout, Cafee, PyTorch and so on. The choice of a machine learning tool depends on the programming language, the possibility of implementing program code, or the need to obtain graphical results. The machine learning tool WEKA was used in this work. WEKA is tried and tested open source machine learning software [4]. The software can be used through a graphical user interface, Java API or a standard terminal. WEKA is widely used for research and industrial applications, contains built-in tools for machine learning tasks, and additionally gives transparent access to well-known toolboxes such as scikit-learn, R, and Deeplearning4j. An excellent choice for Python and Java programmers involved in research in the field of machine learning.

The metrics used to measure the effectiveness of the proposed methods are precision, recall and Fmeasure. Precision is the fraction of relevant attacks among the retrieved attacks (1). Recall is the fraction

of the total amount of relevant attacks that were actually retrieved (2). To determine the balance between precision and recall there is metric combining two concepts. Fmeasure is the harmonic mean of precision and recall (3).

$$\Pr ecision = TP / (TP + FP), \quad (1)$$

$$\mathrm{Re}call = TP / (TP + FN), \quad (2)$$

$$Fmeasure = (2 \times \Pr ecision \times \mathrm{Re}call) / (\Pr ecision + \mathrm{Re}call). \quad (3)$$

In this formula TP (truepositives) is the number of correctly defined attacks, FP (falsepositives) is the number of incorrectly defined attacks, FN (falsenegatives) is the number of undefined attacks.

The dataset CICIDS was tested on various machine learning algorithms with the presentation of the classification results in table 3. The calculations were made on the Intel Core i7-8700T processor, with 4 gigabytes of memory heap size.

Table 3 – CICIDS algorithm testing results

Algorithm	Precision	Recall	Fmeasure
KNN	0.96	0.96	0.96
RF	0.98	0.97	0.97
ID3	0.98	0.98	0.98
Adaboost	0.77	0.84	0.80
MLP	0.77	0.83	0.76
Naive-Bayes	0.88	0.84	0.86
QDA	0.97	0.88	0.92
J48	0.98	0.98	0.98

The presented classification outputs show that some algorithms cope poorly with the task and are not suitable for further research. As a result of the evaluation of the effectiveness of the algorithms, which coped with the task, the «j48 algorithm» was chosen [5]. As an example in figure 4, the output of the J48 algorithm on the CICIDS dataset (day: Wednesday) is presented.

```
Time taken to build model: 992.18 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      692325               99.9454 %
Incorrectly Classified Instances       378                0.0546 %
Kappa statistic                          0.9989
Mean absolute error                      0.0003
Root mean squared error                  0.0132
Relative absolute error                  0.1714 %
Root relative squared error              4.6527 %
Total Number of Instances           692703

=== Detailed Accuracy By Class ===

                         Precision  Recall   F-Measure     Class
                         1,000      1,000    1,000         BENIGN
                         0,990      0,994    0,992         DoS slowloris
                         0,992      0,990    0,991         DoS Slowhttptest
                         0,999      1,000    1,000         DoS Hulk
                         0,995      0,995    0,995         DoS GoldenEye
                         0,833      0,909    0,870         Heartbleed
Weighted Avg.            0,999      0,999    0,999

=== Confusion Matrix ===

      a      b      c      d      e      f   <-- classified as
 439844     22     24    115     24      2 |      a = BENIGN
     21   5763     10      1      1      0 |      b = DoS slowloris
     17     34   5443      3      2      0 |      c = DoS Slowhttptest
     26      0      0 231026     21      0 |      d = DoS Hulk
     25      0      8     21  10239      0 |      e = DoS GoldenEye
      1      0      0      0      0     10 |      f = Heartbleed
```

Figure 4 – J48 algorithm WEKA output for CICIDS (Wednesday)

This output of the program contains information about the operating time of the model used classification methods and general information about the set. Also it compiled confusion matrix and calculated averages, which greatly simplifies the task of data analysis

Creation and implementation of expert rules. Rules created using the WEKA tool were saved as a decision tree j48. In this work, software for creating expert rules was written in the python programming language, consists of 200 lines of code. The program monitors «end nodes» and compiles a set of rules for each of presented attacks (fig. 5).

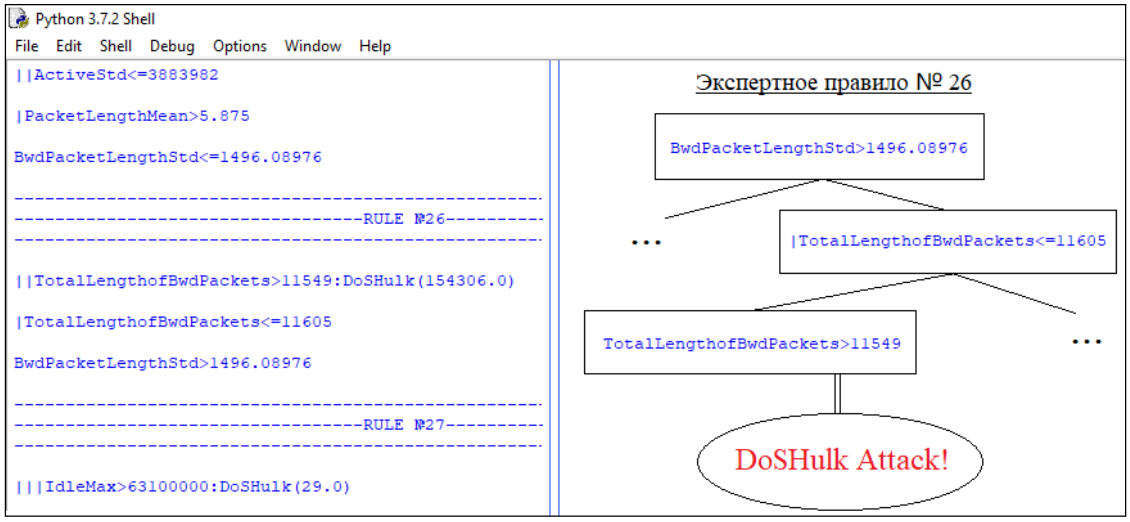


Figure 5 – An example of creating an expert rule for a DoS attack

The figure 5 shows one of the rules for detecting DoS attacks. Thus, if standard deviation size of packet in backward direction is more than 1496.08976 bytes and total size of packet in backward direction in the interval from 11549 to 11605 bytes then the packet is attack (type DoSHulk). As a result of the work, it was revealed that some attributes are more interest for a particular attack. For example, for GoldenEye attacks (type of DoS attacks), the attributes «B. Packet Len Std»,» Flow IAT Min», «Fwd IAT Min», «Flow IAT Mean» are more interesting than the others (for a detailed description of the attributes see [8]).

Then the extracted expert rules are drawn up according to the syntax of the Snort rules language and written to the Snort.rules file. Snort runs in network IDS mode and accesses the rules file, as specified in snort.config. The implemented IDS shows effective indicators of correctly classified information, over 98 %.

Moreover, this system allows detecting new attacks that are not represented in signatures, since the obtained rules are drawn up on the identified patterns [4]. Thus, any modified old attack is detected.

Conclusion. In this research paper, the results of a comparative analysis of 15 datasets for IDS are obtained. Based on the analysis one best dataset CICIDS was selected. Duplicated information is removed from the dataset, such as «Fwd Packet Header». Extra spaces and other input errors are also removed. Then, using the implemented software, the data set is reformatted into a format compatible with the tool WEKA. Next, the implementations of 8 machine learning algorithms are presented. As a result of the evaluation of the effectiveness of the algorithms, the j48 algorithm was chosen. In this work, software for creating expert rules was written. Further, based on the results of classification of the j48 algorithm, expert rules were created that were able to respond to network intrusions.

In the work reproduced IDS with embedded expert rules. All expert rules are implemented in the Snort rules language. Snort runs in network IDS mode and accesses the rules file, as specified in snort.config. The proposed system provides effective detection rates (over 98 %)

In the future, the task will increase the detection and implementation of an artificial immune system for comparative analysis.

Библиографический список

1. Ali L. W. Network intrusion detection and prevention: Concepts and techniques / L. W. Ali, T. Mahbod. – Springer Science, LLC, 2010. – 205 p.

2. Gharib A. An evaluation framework for intrusion detection dataset // A. Gharib, I. Sharafaldin, A. A. Ghorbani // 2016 International Conference on Information Science and Security (ICISS). – 2016. –p. 6.

3. Gong Fengmin. Deciphering Detection Techniques: Part II Anomaly-Based Intrusion Detection / Gong Fengmin. – White Paper from McAfee Network Security Technologies Group, 2003. – 30 p.

4. Hall Mark. The WEKA data mining software: an update / Hall Mark // ACM SIGKDD explorations newsletter_11. – 2009. – № 1. – P. 10–18.

5. Hssina B. A comparative study of decision tree ID3 and C4.5 / Badr Hssina, Abdelkarim Merbouha, Mohammed Erritali // International Journal of Advanced Computer Science and Applications. – 2014. – P. 1–19.

6. Holland T. Understanding IPS and IDS: Using IPS and IDS together for Defense in Depth, Information Security Reading Room / T. Holland. – SANS Institute, 2004. – P. 14.

7. Holmes G. Snort open source network intrusion prevention and detection system (ids/ips) / G. Holmes. – Режим доступа: https://snort.org, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 01.01.2020).

8. Lashkari. A. H. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization // A. H. Lashkari, S. Iman and A. A. Ghorbani // 4th International Conference on Information Systems Security and Privacy (ICISSP). – Portugal, 2018. – P. 108.

9. Lashkari A. H. Network traffic flow analyzer / A. H. Lashkari, Gerard Draper-Gil, Mohammad Saiful Islam Ma mun and Ali A. Ghorbani. – Режим доступа: http://www.netflowmeter.ca/netflowmeter.html, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 01.01.2020).

10. Sagar N. Shah. Signature-Based Network Intrusion Detection System Using SNORT And WINPCAP / Sagar N. Shah. Purnima Singh // IJERT. – December, 2012. – Vol. 1, issue 10. – P. 7.

11. Shiravi A. Toward developing a systematic approach to generate benchmark datasets for intrusion detection // Ali Shiravi, M. T. Hadi Shiravi, and A. A. Ghorbani // Computers and Security. – 2012. – № 31. – P. 357.

References

1. Ali L.W., Mahbod T. *Network intrusion detection and prevention: Concepts and techniques*. Springer Science, LLC 2010. 205 p.

2. Gharib A., Sharafaldin I., Ghorbani A. A. An evaluation framework for intrusion detection dataset. *2016 International Conference on Information Science and Security (ICISS)*, 2016, p. 6.

3. Gong Fengmin. *Deciphering Detection Techniques: Part II Anomaly-Based Intrusion Detection*. White Paper from McAfee Network Security Technologies Group, 2003. 30 p.

4. Hall Mark. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter_11*, 2009, no. 1, pp.10–18.

5. Hssina Badr, Abdelkarim Merbouha, Mohammed Erritali. A comparative study of decision tree ID3 and C4.5. *International Journal of Advanced Computer Science and Applications*, 2014, pp. 1–19.

6. Holland T. *Understanding IPS and IDS: Using IPS and IDS together for Defense in Depth, Information Security Reading Room.* SANS Institute, 2004, p. 14.

7. Holmes G. *Snort open source network intrusion prevention and detection system (ids/ips)*. Available at: https://snort.org (accessed 01.01.2020).

8. Lashkari. A. H., S. Iman, Ghorbani A. A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. *4th International Conference on Information Systems Security and Privacy (ICISSP)*, Portugal, 2018, p. 108.

9. Lashkari A. H., Gerard Draper-Gil, Mohammad Saiful Islam Ma mun and Ali A. Ghorbani. Network traffic flow analyzer. Available at: http://www.netflowmeter.ca/netflowmeter.html (accessed 01.01.2020).

10. Sagar N. Shah, Purnima Singh: Signature-Based Network Intrusion Detection System Using SNORT And WINPCAP, (IJERT), ISSN: 2278-0181, Vol. 1 Issue 10, December- 2012, 7 p.

11. Shiravi A., Hadi Shiravi M. T., Ghorbani A. A. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers and Security*, 2012, no. 31, p. 357.