DOI: 10.17725/rensit.2020.12.287

Detection of DoS attacks caused by CONNECT messages of MQTT protocol

Dmitrii I. Dikii

St. Petersburg National Research University of Information Technologies, Mechanics and Optics (University ITMO), http://www.itmo.ru

St. Petersburg 197101, Russian Federation

E-mail: dimandikiy@mail.ru

Received September 25, 2019, reviewed on March 10, 2020, after finalization on March 30, 2020 accepted April 13, 2020

Abstract. **Detecting DoS attacks within the Internet of Things is an urgent task to ensure the security of this infrastructure. The malefactor, undertaking the attack, generates a large number of connection requests to the Internet of Things network based on the MQTT protocol. This makes the gateway unavailable for other users. The author discusses the approaches and methods of detecting DoS attacks within the Internet, in general, as well as within the Internet of Things, in particular. The method of feature vector generation for detecting DoS attacks based on the network traffic analysis was suggested. The feature vector consists of parameters of message transmission frequency within a time interval from a device with the same IP-address. The multilayer perceptron, the random forest algorithm, the support vector machine are classifiers in this study. The author constructed an experimental assembly to generate training and testing sets with the supplied parameters. The experiment results showed: in order to achieve maximum classification accuracy, the dimension increase of the feature vector is not required. A comparison of the mentioned above algorithms by the F1-score value was carried out, which proved the artificial neural network – the multilayer perceptron – to be the best classifier. At that, the time interval, on which the feature vector generation is based, must be higher than 1.5 seconds for the accuracy to be over 0.99 under the legal device connection frequency once per second. The research gave positive results of implementing the reviewed classifiers based on the suggested feature vector to detect DoS attacks.**

Keywords: **Internet of things, DoS, MQTT, machine learning, random forest, multilayer perceptron, support vector machine, telecommunication, attack detection**

UDC 004.052.3

Acknowledgments. The reported study was funded by RFBR, project number №19-37-90051

For citation: Dmitrii I. Dikii. Detection of DoS attacks caused by CONNECT messages of MQTT protocol. *RENSIT,* 2020, 12(2):287-296. DOI: 10.17725/rensit.2020.12.287.

Contents

1. INTRODUCTION

Recently, there has been a significant popularity increase of technologies employed in the Internet. One of those technologies is the Internet of Things [1]. The main characteristic of the technology that allows uniting a variety of projects under a single name – the Internet of Things – is a possibility for a large number of devices, functioning without an operator, to communicate to carry out a single common task. The devices mentioned below must have only the essential capabilities. This makes them significantly cheaper than common workstation computers (personal computers, smartphones, etc.). Certain devices in the Internet of Things network function on an independent power supply. This imposes limits on the employment of such devices from the energy saving point of

view. Data transfer technologies and protocols that significantly reduce energy demand of the terminal device are developed to increase its operational life from an independent power supply. Research in the field of the Internet of Things networks encloses the whole protocol stack of the OSI model [2]. One of those protocols is the MQTT (message quality telemetry transport), which was developed by the OASIS alliance [1]. Currently, the most widespread version is the MQTT protocol 3.1.1.

Along with the tendency to simplify protocols for Internet of Things devices, there is an increase in information security threats. Information circulating in the Internet of Things network remains plaintext. This may lead to negative consequences from the information owner's side. The most striking case in point is the medical field. Papers [4, 5] present arguments on security enhancement namely in healthcare. One of presented methods is to enhance device authentication.

Apart from threats common to the Internet of Things networks, threats typical to all devices connected to the Internet have to be noted. Generally, those are attacks such as a man-in-the-middle, phishing, viruses, trojans, etc. Another threat is the distributed DoS attack. Its main feature is that a large number of network devices send requests to the victim device. Due to the exceedance of the request-processing maximum per time interval, the victim does not handle the load and becomes unavailable to other devices. According to the Kaspersky Laboratory analytics, the malware based on the Mirai botnet has become the most widespread around the world by the end of first quarter of 2019 [6]. As part of the botnet behavior, the DoS attack becomes an immediate threat.

Thus, deploying the Internet of Things network infrastructure in an organization or at home, one should consider classical threats of information security as well as specific ones of the Internet of Things networks and their combinations.

In this paper, the author discusses the problem of the Internet of Things network devices' excessive use of the MQTT protocol capabilities to employ the DoS attacks.

Nowadays, the following attack classification is given:

- Bandwidth exhaustion attack;
- Victim resource exhaustion attack;
- Infrastructure attack;
- Zero day attack [7].

Malefactors most frequently use two types of DoS attacks. The first is the bandwidth saturation with information until the legitimate source signal does not reach the recipient. The second type of attack exploits the vulnerability of other protocols to exhaust the server's resources: memory space, CPU usage time. Moreover, one can implement the network and transport layer protocols as well as the application ones, such as the HTTP, for the second type of attack. A striking example is the TCP SYN attack, when the malefactor sends a request to establish a connection via the TCP protocol, but instead of specifying his own IP address, a nonexistent one is specified. The server stands by to establish a connection, but does not receive feedback overlong. However, the information concerning unestablished connection is saved on the server side, thus, leading to the victim bandwidth exhaustion.

Preventive measures to secure information systems from such attacks can be divided into two stages:

- Detection;
- Counter acting.

Detection is carried out through network traffic analysis. "Hop count" packet filtering method has gained the most popularity [8]. In this method, the number of TCP packets and statistical parameters is assessed: SYN flag, TTL, the source and destination addresses, etc. Paper

[9] reviews the DoS attack detection method and the security against it, consisting of MAC-addresses filtering and cryptographic processing.

Methods based on artificial intelligence and machine learning are more rapidly suggested to detect attacks. Thus, the authors propose to employ swarm algorithms in paper [10]. The accuracy of DoS attack detection by the suggested method is 0.75-0.80.

TCP traffic is most often analyzed by estimating the server response time for normal traffic and for attacks to protect Internet resources. During an attack, the server response time significantly increases. This fact is the basis for traffic classification. For example, using the LS-SVM algorithm made it possible to achieve classification accuracy of over 0.92 [11].

Many other papers reviewed the support-vector machines (SVM) for the DoS attacks detection. For example, the authors of [12] were able to gain 99% of positive attack identification from the TCP traffic employing the SVM on the DARPA database. The authors of [13, 14] were able to gain similar abnormal traffic detection accuracy employing the support-vector machines. The authors of paper [15] review the variations of this method. In this case, the accuracy of the presented methods is more than 0.92. The difference between the studies dedicated to employing SVM to detect DoS attacks is in the distinctive solutions to form the feature vector. Paper [16] presents a study on the influence of various features on the classification accuracy. Thus, the choice of the feature vector is the main factor affecting accuracy. The SVM algorithm or its variations showed positive results in detecting attacks under the TCP traffic analysis.

Another approach to detect DoS attacks is to employ artificial neural networks (ANN). There is a great number of artificial neural networks variations. The most widespread model, the multilayer perceptron (MLP), is reviewed in paper [11]. The comparison of the SVM and the ANN has shown that the latter has lower accuracy and requires more time to make a decision [17]. Paper [18] presents the results of the experiment on employing the ANN to detect anomalies in traffic via the TCP and ICMP protocols. The approach suggested by the authors achieved a 0.98 accuracy of detecting attacks. Moreover, the ensemble of recurrent artificial networks is used [16].

The random forest (RF) algorithm class and the decision trees are used to detect attacks. This approach shows good results. For example, the detection accuracy in paper [20] is over 0.96. Similar studies [21-23] present high detection accuracy. Fuzzy logic approaches are also used to detect DoS attacks [24].

Methods of abnormal traffic detection in the Internet of Things networks are based on data analysis of transport and network protocols, as described in [25, 26]. However, the Internet of Things networks use other layer protocols that are vulnerable to DoS attacks. Such as application layer protocols (CoAP, MQTT), and protocols of lower layers (for example, LoRa). Attacks on the physical and data-link layers are most widespread in the wireless sensor networks. For example, an attack aimed at resource exhaustion is described in [27]. Another attack common to the Internet of Things network is "blackhole". In this case, a device communicates to other devices in the network that its node has the shortest route to deliver a packet. However, all packets delivered to this node will be dropped [28]. In addition, there are attacks that create jamming for information transmission via radio channels, thus, causing DoS [29].

2. METHODS AND MATERIALS

In relation to the application layer protocol MQTT, one can note its prepossession to DoS attacks. Generally, this is employed by increasing the load on the network elements to disrupt the communication between devices. The protocol functions on the "publish-subscribe" pattern. Thus, the network has a key element called

the gateway. It is responsible for redirecting messages from the sender to the recipient. As all messages pass the gateway it is the most vulnerable element to the attack. Research was carried out concerning the influence of message parameters (flags, message amount, etc.) on gateway sustainability under high loads. Authors in most studies reviewed only messages of PUBLISH type with the following parameters:

- Quality of service (QoS) [30, 31];
- Number of subscribers [32];
- Message payload size [33, 34];
- Cryptographic processing of messages [32].

On the contrary, the process of device connection to the gateway is not taken into account. When simultaneously sending a large amount of connection requests (CONNECT messages), the gateway may not be able to handle the load. As a result, legal devices will not be able to connect to the gateway to send or receive messages [35]. Thus, in the networks that operate on the MQTT protocol one must detect abnormal device behavior on all stages of protocol operation.

The purpose of this work is to develop a method of detecting a denial-of-service attack caused by abnormal behavior of network devices by exploiting CONNECT messages of the MQTT Protocol using machine learning algorithms. To achieve this purpose, first, the task of choosing the optimal feature vector is solved. The second task to be solved is to determine the most effective classification method. The following algorithms were considered as classifiers in this study: multilayer perceptron, random forest, and support vector machine with a radial basis function of the kernel, programmatically implemented on the basis of the WEKA project [36]. To generate training and testing data sets, an experimental assembly was created (**Fig. 1**), consisting of a gateway, communication equipment, and several computers that simulate the behavior of many Internet of things devices using the Paho-mqtt

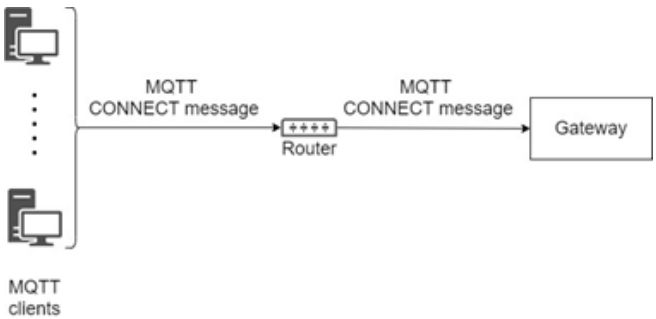

Fig. 1. *Experimental assembly scheme.*

framework [37]. The gateway was a Raspberry Pi 3 model B microcomputer with Moquette project software on JAVA language [38].

A feature vector must be formed, for the message to be correctly classified. As the malefactor merely needs the gateway address and the port number to send a connection request, thus, service information as device ID and username can be generated automatically and are not considered. Therefore, the main parameters describing CONNECT message via the MQTT protocol are:

- Sender address as the IP address. It is required to keep a block list of addresses from which the attack originates. In this case, the IP address is used as a tag;
- Number of connection requests per a time interval;
- Connection-time mean between connection requests per a time interval;
- Binary value determining the employment of cryptographic processing via TLS protocol, which significantly affects time of connecting to the gateway: 0 – TLS protocol is not employed, 1 – TLS protocol is employed.

Therefore, the feature vector of the connection message consists of three main parameters per a single analyzed time interval (hereafter m).

The choice of a time interval plays an important role in forming a feature vector. Moreover, it can be not a single time interval, but a complex. Thus, the feature vector dimension can be increased and be estimated by the formula:

$$W = 1 + 3k, \quad (1)$$

where k – the number of time intervals m.

A time interval m is a period between the moment when the gateway received a message and the moment of the predetermined number of milliseconds to this instance. An example of forming a body of three time intervals is depicted on **Fig. 2**.

Legal traffic was generated based on device behavior simulation of a real network with consideration to installed secure and insecure connections via the TLS protocol. The simulation of legal traffic depends on practical application conditions of the network, thus, the training data set will vary depending on the estimated maximum network load. The abnormal traffic was simulated by generating a high message flow with requests to connect via an insecure channel as well as a secure one by the TLS protocol. Channel security selection is determined randomly for each connection.

A test data set consisting of legal and abnormal traffic examples was collected to determine the best classifier. A network operation scenario in a standard mode consisting of ten thousand connections was used to collect the data set. In addition, there was a scenario with a potential attack consisting of five thousand sequentially sent CONNECT messages.

The question of selecting time intervals m, based on which the feature vector will be formed remains. The following methodology for selecting optimal time intervals was suggested.

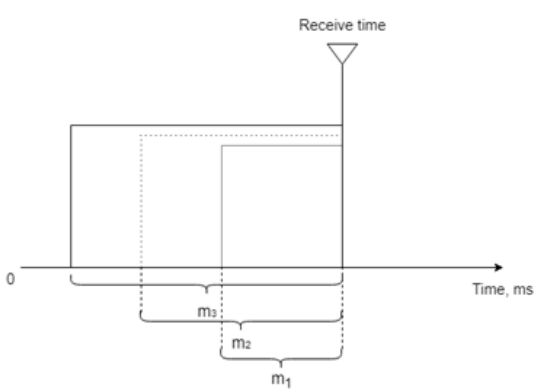

Fig. 2. *Time interval definition scheme for feature vector,* $m_3 > m_2 > m_1$.

As a first step, expert analysis determines a finite set of time intervals M with natural values ($M \in N$).

On the second step, three classifiers (MLP, RF, SVM) are trained for each value of $m \in M$ and those classifiers undergo proving on the test data set.

In the third step, the classification quality is evaluated by calculating the F1-score, which is a weighted average of precision and recall. This metric is widely used in evaluating the quality of binary classification for machine learning methods, as is shown in [24, 39]. To calculate this value, the classification results of the number of correctly and incorrectly classified messages on the test dataset are used (**Table 1**, where TP – the number of legitimate messages recognized correctly; TN – the number of attack messages recognized correctly; FP – the number of abnormal messages recognized incorrectly; FN-the number of legal messages recognized incorrectly).

Then classification precision is calculated by the formula:

$$\text{Precision}=TP/(FP+TP) \quad (2)$$

and recall by the formula:

$$\text{Recall}=TP/(TP+FN) \quad (3)$$

Knowing those values, one can calculate the F1-score using the formula:

$$F=(2\times\text{Precision}\times\text{Recall})/(\text{Precision}+\text{Recall}) \quad (4)$$

On the fourth step, a finite set S is formed with unique combinations of nonrecurring elements $m \in M$ of the length l, such that $2 \leq l \leq |M|$.

Table 1

Confusion matrix

	Legal messages	Abnormal messages
Correctly classified messages	TP	TN
Incorrectly classified messages	FN	FP

On the fifth step, steps 2 and 3 are repeated with the determined time intervals of the set *S*. The best time interval combination, under which the highest F1-score value is reached.

3. RESEARCH RESULTS

The following initial time interval set for *M* was established for the simulated network {20, 50, 100, 150, 200, 250, 500, 1000, 1500, 2000, 3000} ms.

The training data set consists of two arrays. The first one includes information about legal message flow of the connection to the gateway. Whereas the second has information about a flow similar to the DoS attack. The following model was established to generate a legal data flow. A time interval *I*, during which at least a single connection message is sent guaranteed, is established. The dispatch time *i* is randomly determined (via equal probability distribution so that the data set contains examples with both small and close to maximum values of the delay between messages) from the interval *I* ($i \in I$). Therefore, the time difference between the dispatches of two sequential messages is defined as follows:

$$\measuredangle T = i + t, \quad (5)$$

where *t* – the time required to process the messages and receive a response from the gateway (via an insecure channel not over 50 ms in average, via a secure one – not over 700 ms for the considered experimental assembly), *i* – random time delay, not over the maximum value of *I* interval.

Two intervals were considered, $I = [0, 1000]$ ms and $I = [0, 500]$ ms, to determine the impact of a training sample containing legitimate traffic on the quality of classification,.

The author simulated a situation where a large CONNECT message flow was sent to the gateway over a short period to generate the second data array containing data about abnormal data flow. The training data set

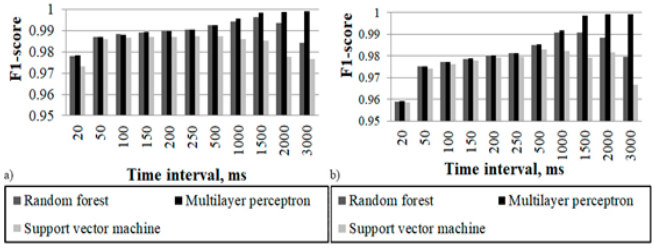

Fig. 3. *Classification results of legal messages generated from the interval a)* $I = [0, 1000]$ *ms, b)* $I = [0, 500]$ *ms.*

consists of ten thousand messages for legal data flow and five thousand messages for simulating the DoS attack. The classification results by the testing data set under one time interval $m \in M$ is presented on **Fig. 3**.

The results of the conducted experiment show the classifier based on the multilayer perceptron to be the best one. An increase of the time interval, during which traffic statistics is collected, leads to the increase of the F1-score value. For example, the value reaches 0.9989 ± 0.0001 under the interval of two seconds.

There is a complex dynamic to the random forest algorithm. The F1-score increases as the time interval increases from 20 ms to 1500 ms. However, further increase of the time interval to three seconds leads to the classifiers characteristics degradation. The F1-score value maximum is achieved at interval m = 1500 ms and is equal to over 0.9934 ± 0.0027.

The support-vector machine was the worst algorithm to cope with classification. The F1-score value was lower than that of other algorithms under every considered time interval. The F1-score value maximum is achieved at m = 500 ms and the classification accuracy equals 0.985 ± 0.0021. The classification accuracy decreases, when increasing the time interval to three seconds.

The employment of a set of analyzed intervals $m \in M$ did not bring a significant positive effect. **Figure 4** presents F1-score values of the considered algorithms under the following interval sets: {200, 250, 500},

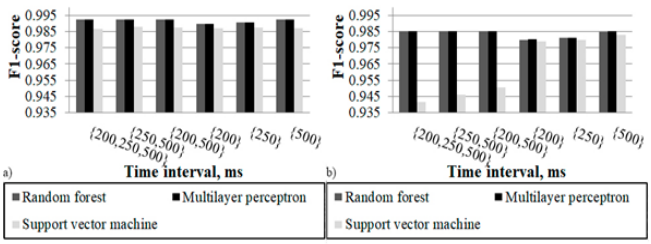

Fig. 4. *Classification results for legal messages generated from the interval a) I = [0, 500] ms, b) I = [0, 1000] ms.*

{250, 500}, {200,500}. Intervals {200}, {250}, {500} are shown for comparison purposes. Thus, the employment of a set of intervals does not increase classification quality, but often decreases it. For example, when employing the support-vector machine under the legal request frequency of the I = [0, 1000] interval, one can observe evident negative dynamics of the F1-score in cases of feature space increase. The F1-score values are lower or insignificantly higher than the values under classification by a single widest interval in most other cases.

Therefore, the use of a set of intervals to form a feature vector of a larger dimension is inefficient.

4. DISCUSSION

In the scope of the conducted study, the detection methods of DoS attacks employed in the Internet networks as well as the Internet of Things networks were analyzed. The suggested feature vector for CONNECT messages by the MQTT protocol consists of three main parameters: the amount of messages per time interval, mean connection-time between two sequential messages, the mean value of the parameter responsible for the TLS protocol employment when creating a cryptographically secure channel per a time interval, and a tag-parameter – sender's IP address.

The following algorithms were considered as classifiers: multilayer perceptron, random forest algorithm, support-vector machine. The experiment based on generated training and testing data sets showed that all algorithms cope with the traffic classification task with accuracy over 0.90. The multilayer perceptron model had the best classification quality. The F1-score values increased according to the time interval increase, during which traffic statistics was collected and the feature vector was formed. The random forest algorithm coped with the classification with worse values. The time interval increase up to 1.5 seconds has a positive effect on the dynamics F1-score. However, further increase of the time interval leads to the F1-score value decrease. The worst to cope with the classification was the support-vector machine. The dynamics of F1-score is similar to the one of random forest algorithm. The F1-score maximum is gained at the 500 ms interval. The employment of feature vectors of larger dimension is inefficient as the classification characteristics may not only remain the same but also degrade.

5. CONCLUSION

Therefore, among all considered approaches and algorithms of detecting DoS attacks by the suggested feature vector (the feature vector in this case will have dimension 4) it is recommended to employ the multilayer perceptron. That model showed the best results in contrast of other considered methods. However, the classification quality increases with the time interval during which traffic statistics is collected. However, it is worth noting that increasing this interval will lead to a large computational and time-consuming cost of training the model and making a decision. The quality of classification based on the random forest algorithm or the support vector machine with the radial basis function of the core is worse than that of the multilayer perceptron, but the values of the F1-score are high enough to be used.

Further research will be dedicated to studying DoS attacks caused by the abuse of other types of MQTT protocol messages.

REFERENCES

1. Ashton K. That 'Internet of Things' Thing. *RFID Journal*, 2009, 22:97–114.
2. Standart "ISO/IEC 7498-1:1994 [ISO/IEC 7498-1:1994] Information technology — Open Systems Interconnection — Basic Reference Model: The Basic Model". *ISO/IEC Information Technology Task Force (ITTF) web site*, 1994.
3. Standart ISO/IEC 20922:2016 Information technology – Message Queuing Telemetry Transport (MQTT) v3.1.1. *ISO/IEC Information Technology Task Force (ITTF) web site,* 2016.
4. Albalawi U, Joshi S. Secure and Trusted Telemedicine in Internet of Things IoT. *Proceedings of 2018 IEEE 4th World Forum on Internet of Things* (WF-IoT), 2018, pp. 30-34. DOI: 10.1109/WFIoT.2018.8355206.
5. Wazid M, Kumar Das A, Khurram Khan M, Al Dhawailie AlGhaiheb A, Kumar N, Vasilakos AV. Secure Authentication Scheme for Medicine Anti-Counterfeiting System in IoT Environment. *IEEE Internet of Things Journal,* 2017, 4(5):1634-1646. DOI: 10.1109/JIOT.2017.2706752.
6. Chebyshev V, Sinitsyn F, Parinov D, Larin B, Kupreev O, Lopatin E. Development of information threats in the first quarter of 2019. *Statistics. Kaspersky Security Bulletin 2019. Statistics.* Threat reports URL: https://securelist.ru/it-threat-evolution-q1-2019-statistics/94021/ (дата обращения: 20.08.2019).
7. Mahjabin T, Xiao Y, Sun G, Jiang W. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. *International journal of distributed senssor networks*, 2017, 13(12):1-32. DOI: 10.1177/1550147717741463.
8. Jin C., Wang H., Shin K.G Hop-Count Filtering: An effective defense against spoofed DDoS traffic. *Proc. of the ACM Conf. on Computer and Communications Security,* 2003, pp. 30-41. DOI: 10.1145/948109.948116.
9. Prakash A, Satish M, Sri Sai Bhargav T, Bhalaji N. Detection and Mitigation of Denial of Service Attacks Using Stratified Architecture. *Proc. of the 4th Intern. Conf. on Recent Trends in Computer Science & Engineering Detection Procedia Computer Science,* 2016, 87:275-280. DOI: 10.1016/J.PROCS.2016.05.161.
10. Sharma S, Gupta A, Agrawal S. An Intrusion Detection System for Detecting Denial-of-Service Attack in Cloud Using Artificial Bee Colony. *Proc. of the Intern. Congress on Information and Communication Technology, Advances in Intelligent Systems and Computing,* 2016, pp. 137-145. DOI: 10.1007/978-981-10-0767-5_16.
11. Sahi A, Lai D, LI Y, Diykh M. An Efficient DDoS TCP Flood Attack Detection and Prevention System in a Cloud Environment. *IEEE Access,* 2017, 5:6036-6048. DOI: 10.1109/ACCESS.2017.2688460.
12. Mukkamala S, Sung AH. Detecting Denial of Service Attacks Using Support Vector Machines. *Proc. of the 12th IEEE Intern. Conf. on Fuzzy Systems,* 2003, pp.1231-1236. DOI: 10.1109/FUZZ.2003.1206607.
13. Manuel S. Hoyos Ll, Gustavo AIE, Jairo IV, Castillo OL. Distributed Denial of Service (DDoS) Attacks Detection Using Machine Learning Prototype. *Proc. of the 13th Intern.Conf., Advances in Intelligent Systems and Computing,* 2016, pp. 33-41. DOI: 10.1007/978-3-319-40162-1_4.
14. Kim D, Lee KY. Detection of DDoS Attack on the Client Side Using Support Vector Machine. *Intern. J. of Applied Engineering Research,* 2017, 12(20):pp. 9909-9913.
15. Xu X, Wei D, Zhang Y. Improved Detection Approach for Distributed Denial of Service Attack Based on SVM. *Proc. of the 3th Pacific-Asia Conference on Circuits, Communications and System (PACCS)*, 2011, pp. 1-3. DOI: 10.1109/PACCS.2011.5990284.
16. Chan APF, Ng WWY, Yeung DS, Tsang ECC. Refinement of rule-based intrusion detection system for denial of service attacks by support vector machine. *Proc. of the 13rd*

Intern. Conf. on Machine Learning and Cybernetics, 2004, pp. 4252- 4256. DOI: 10.1109/ICMLC.2004.1384585.

17. Tsang GCY; Chan PPK; Yeung DS; Tsang ECC. Denial of service detection by support vector machines and radial-basis function neural network. *Proc. of Intern. Conf. on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, 2004, pp. 4263-4267. DOI: 0.1109/ICMLC.2004.1384587.
18. Saied A, Overill RE, Radzik T. Detection of known and unknown DDoS attacks using Artificial Neural Networks. *Neurocomputing,* 2016, 172:385-393. DOI: 10.1016/j.neucom.2015.04.101.
19. AI Islam ABMA, Sabrina T. Detection of various Denial of Service and Distributed Denial of Service Attacks using RNN Ensemble. *Proc. of 12th Intern. Conf. on Computer and Information Technology (ICCIT 2009),* 2009, pp. 603-608. DOI: 10.1109/ICCIT.2009.5407308.
20. Lakshminarasimman S; Ruswin S; Sundarakantham K. Detecting DDoS attacks using decision tree algorithm. *Proc. of 4th Intern. Conf. on Signal Processing, Communication and Networking (ICSCN)*, 2017, pp.1-6. DOI: 10.1109/ICSCN.2017.8085703.
21. Chen L, Zhang Y, Zhao Q, Geng G, Yan Z. Detection of DNS DDoS Attacks with Random Forest Algorithm on Spark. *Proc. of 2nd Intern. Workshop on Big Data and Networks Technologies Procedia Computer Science,* 2018, 134:310-315. DOI: 10.1016/j.procs.2018.07.177.
22. Idhammad M, Afdel K, Belouch M. Detection System of HTTP DDoS Attacks in a Cloud Environment Based on Information Theoretic Entropy and Random Forest. *Security and Communication Networks*, 2018, 2018:1-13. DOI: 10.1155/2018/1263123.
23. Cheng J, Li M, Tang X, Sheng VS, Liu Y, Guo W. Flow Correlation Degree Optimization Driven Random Forest for Detecting DDoS Attacks in Cloud Computing. *Security and Communication Networks,* 2018, 2018:1-14. DOI: 10.1155/2018/6459326.
24. Haripriya AP, Kulothungan K. Secure-MQTT: an efficient fuzzy logic-based approach to detect DoS attack in MQTT protocol for internet of things. *Journal on Wireless Communications and Networking*, 2019, Vol. 90. DOI: 10.1186/s13638-019-1402-8.
25. Doshi R, Apthorpe N, Feamster N. Machine Learning DDoS Detection for Consumer Internet of Things Devices. *Proc. of IEEE Symposium on Security and Privacy Workshops,* 2018, pp. 29-35. DOI 10.1109/SPW.2018.00013.
26. Meidan Y, Bohadana M, Mathov Y, Mirsky Y, Breitenbacher D, Shabtai A, Elovici Y. N-BaIoT: Network-based Detection of IoT Botnet Attacks Using Deep Autoencoders. *IEEE Pervasive computing,* 2018, 13(9):1-8. DOI: 10.1109/MPRV.2018.03367731.
27. Mallikarjunan KN, Muthupriya K, Shalinie SM. A survey of Distributed Denial of Service attack. *Proc. of 10th Intern. Conf. on Intelligent Systems and Control (ISCO)*, 2016, pp. 1-6. DOI: 10.1109/ISCO.2016.7727096.
28. Cetinkaya A, Ishii H, Hayakawa T. An Overview on Denial-of-Service Attacks in Control Systems: Attack Models and Security Analyses. *Entropy*, 2019, 21:1-29. DOI:10.3390/E2102021029.
29. Wood AD, Stankovic JA. Denial of Service in Sensor Networks. *Computer,* 2002, 35(10):54-62. DOI: 10.1109/MC.2002.1039518.
30. Chifor B, Patriciu V. Mitigating DoS attacks in publish-subscribe IoT networks. *Proc. of Conf.: Electronics, Computers and Artificial Intelligence,* 2017, pp. 1-6. DOI: 10.1109/ECAI.2017.8166463.
31. Handosa M, Gracanin D. Performance evaluation of mqtt-based internet of things system. *Proc. of Winter Simulation Conference,* 2017, pp. 4544-4545. DOI: 10.1109/WSC.2017.8248196.
32. Fehrenbach P. Messaging Queues in the IoT Under Pressure-Stress Testing the Mosquitto MQTT

Broker. *Fakultät Informatik. Hochschule Furtwangen University*, 2017. URL: https://blog.it-securityguard.com/wp-content/uploads/2017/10/IOT_Mosquitto_Pfehrenbach.pdf.

33. Firdous SN, Baig Z, Valli C, Ibrahim A. Modelling and Evaluation of Malicious Attacks against the IoT MQTT Protocol. *Proc. IEEE Intern.Conf. on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2017, pp. 748-755. DOI: 10.1109/ITHINGS-GREENCOM-CPSCOM-SMARTDATA.2017.115.
34. Bao C, Guan X, Sheng Q, Zheng K, Huang X. A Tool for Denial of Service Attack Testing in IoT. *Proc. 8th Intern. Conf. on Information Technology in Medicine and Education (ITME)*, 2016, pp. 1-6.
35. Official web-site WEKA project. URL: https://www.cs.waikato.ac.nz/ml/weka/32 (date of the application: 11.03.2020).
36. Dikiii D.I. MQTT protocol analysis for denial of service attacks. *Scientific and technical. information bulletin technology, mechanics and optics of ITMO*. 2020, 2(2). DOI: 10.17586/2226.1494.2020.20.2.
37. Official web-site of client paho-MQTTt. URL: https://pypi.org/project/paho-MQTT/1.3.0/ (date of the application: 20.08.2019).
38. Brokers official website Moquette. URL: https://projects.eclipse.org/projects/iot.moquette (date of the application 20.08.2019).
39. Hasan M, Islam M, Zarif I, Hashem M. Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Internet of Things,* 2019, 7:1–14. DOI: 10.1016/J.IOT.2019.100059.