International Journal of Electrical and Computer Engineering (IJECE)
Vol. 10, No. 4, August 2020, pp. 3651~3659
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i4.pp3651-3659

Detection of the botnets' low-rate DDoS attacks based on self-similarity

Sergii Lysenko[1], Kira Bobrovnikova[2], Serhii Matiukh[3], Ivan Hurman[4], Oleh Savenko[5]
[1,2,3,5]Department of Computer Engineering and System Programming, Khmelnytskyi National University, Ukraine
[4]Department of Software Engineer, Khmelnytskyi National University, Ukraine

Article Info

Article history:

Received Aug 4, 2019
Revised Jan 16, 2020
Accepted Feb 1, 2020

Keywords:

Botnet detection
Cyber attack
Hurst coefficient
Low-rate DDoS attack
Network traffic self-similarity

ABSTRACT

An article presents the approach for the botnets' low-rate a DDoS-attacks detection based on the botnet's behavior in the network. Detection process involves the analysis of the network traffic, generated by the botnets' low-rate DDoS attack. Proposed technique is the part of botnets detection system–BotGRABBER system. The novelty of the paper is that the low-rate DDoS-attacks detection involves not only the network features, inherent to the botnets, but also network traffic self-similarity analysis, which is defined with the use of Hurst coefficient. Detection process consists of the knowledge formation based on the features that may indicate low-rate DDoS attack performed by a botnet; network monitoring, which analyzes information obtained from the network and making conclusion about possible DDoS attack in the network; and the appliance of the security scenario for the corporate area network's infrastructure in the situation of low-rate attacks.

Corresponding Author:

Sergii Lysenko,
Department of Computer Engineering and System Programming,
Khmelnytskyi National University,
11 Instytutska str., Khmelnytskyi, Ukraine, 29016.
Email: sprlysenko@gmail.com

1. INTRODUCTION

Nowadays the cybercriminals implement different ways to obtain the profit from the legitimate businesses, which have become theirs target. Malware are one of the most powerful cybercriminals' tools for attaining such goals [1-2]. One the type of the malicious action against the users' computer systems, cloud infrastructure the distributed denial-of-service (DDoS) attacks–the attempt to disrupt normal traffic of a targeted server, service or network by overwhelming the target or its surrounding infrastructure with a flood of Internet traffic [3].

In the modern cyber world a botnets are the main tool for performing of such type of attacks [4]. The bots of botnets are compromised devices designed to attack a single server, network or application with an overwhelming number of requests, packets or messages. A low and slow attack is a another type of DoS or DDoS attack that relies on a small stream of very slow traffic with requests which can target application or server resources, thereby preventing genuine users from accessing the service. To carry out low and slow attacks cyber attackers can use HTTP headers, HTTP post requests, or TCP traffic.

Unlike a brute-force attacks, the low and slow attacks require very little bandwidth and can be hard to mitigate, as each bot is a legitimate Internet device and generate by them slow attack traffic is very difficult to distinguish from those of legitimate clients [5–6]. One of the way of the low-rate DDoS attacks detecting is the traffic analysis concerning its self-similarity of traffic. This method allows identifying the hidden malicious traffic in real-time.

2. RESEARCH METHOD

In recent years, the great number of the botnet detection approaches have been developed. The works [7, 8] are focused on analysis the DDoS attack issues at application layer. The classification of DDoS threats based on abnormal behavior at application layer and the summarized information about various DDoS tools were considered. Furthermore, it categorizes DDoS attack handling techniques based on monitoring, preventing, detecting, and mitigating concepts.

In [9] the network traffic pattern generated by different types of slow DoS attack which is targeted on HTTP application was analyzed. By monitoring and analyzing some significant network based parameters such as window size and delta time of the packet, which can be collected from the network gateway, a host machine independent early detection of slow DoS attack is derived and preventive action can be initiated from the network gateway itself.

In [10] a network-based slow HTTP DDoS attack defense method, which is assisted by a software-defined network that can detect and mitigate slow HTTP DDoS attacks in the network, was proposed. In [11] a comprehensive survey of DDoS attack, including a systematic analysis of this type of attacks, protection and mitigation techniques, possible limitations and challenges of existing research and some important future research directions are outlined.

In the paper [12] a defense method against the distributed slow HTTP DoS attack by disconnecting the attack connections selectively by focusing on the number of connections for each IP address and the duration time was developed. In [13] a framework for the detection and mitigation of the slow-running DDoS attacks within the network infrastructure without requiring access to servers under at-tack was proposed. In addition, the several schemes for the attackers identification in the network based on the measuring the packet rate and the uniformity of the packet distances were considered. In [14] the slow read DoS attack was analyzed and the secure setting of web server against such attack were derived. Authors found out that the efficient attack can be realized when the bandwidth is over 500 Kbps and that attacker can more effective attack by setting the connection rate equal to the process capability of web server.

The approach [15] have differentiated the legitimacy of any traffic irrespective of the network traffic protocol using the Hurst parameter and thus allows detecting DDoS attacks based on self-similarity property of network traffic. The method shows high-detection accuracy for both low-rate and high-rate DDoS attacks and minimum false positives, however only when the some assumptions are true. Otherwise, the range of Hurst parameter value for attack traffic may change, the false alarm percent-age may be higher, and the further fine-tuning may be needed. In [16-17] the machine learning-based methods for cyberattacks detection are presented. Network behavior-based detection techniques are presented in [18], where the investigation of a large amount of normal traffic and amount of malicious traffic is provided. In [19, 20] the systematic review of aspects of DDoS attacks detection and new frameworks are investigated.

3. RESULTS AND ANALYSIS

Due to high intension of the cyberattacks development a great amount of techniques devoted to this problem have been produces during last years. One of them is a botnet detection system–BotGRABBER. Presented system was developed from the idea to detect the botnets' attacks using the multi-agent system [21]. The next generations of the BotGRABBER system have obtained the possibility to detect the botnets that use DNS evasion techniques (cycling of IP mapping, "domain flux", "fast flux" and DNS-tunneling) via DNS traffic analysis, and the possibility to analyze the software's behavior in the host, which may indicate the possible presence of bot directly in the host [22-24].

The most important upgrade of BotGRABBER system was its transformation into the self-adaptive system for the corporate area networks' resilience in the presence of botnets' cyberattacks. Based on the gathered Internet traffic features inherent to cyberattacks, the BotGRABBER system was able to produce the security scenarios according to cyberattacks performed by botnets in order to mitigate the attacks and ensure the network's resilient functioning. The proposed approach used the semi-supervised fuzzy c-means clustering, where the objects of clustering were the feature vectors which elements may indicate the appearance of cyber threats in the corporate area networks [25]. This paper presents the approach for the botnet detection of the low rate DDoS attacks via the BotGRABBER system.

4. THE PROPOSED METHOD

The low-rate DDoS attacks identification based on the traffic self-similarity analysis is the part of botnets detection process performed by a self-adaptive system–BotGRABBER system [25]. It presents the framework for assuring the networks' resilience under the botnets' cyberattacks. In order to detect the botnets its main features are to be gathered and analyzed. The features are formed as feature vectors and are clustered and a result of the clustering is the assignment of each feature vector to a cluster, which is corresponding to a given cyberattack.

The low-rate DDoS attacks identification based on the traffic self-similarity analysis is the part of botnets detection process performed by a self-adaptive system–BotGRABBER system [25]. It presents the framework for assuring the networks' resilience under the botnets' cyberattacks. In order to detect the botnets its main features are to be gathered and analyzed. The features are formed as feature vectors and are clustered and a result of the clustering is the assignment of each feature vector to a cluster, which is corresponding to a given cyberattack.

This article presents detailed description of the botnets' detection process, which is based on the traffic self-similarity analysis, as this factor may indicate its presence in the network. Thus, the low-rate DDoS attacks detection includes learning and monitoring stages.

a. The learning stage consists of the following steps:
   - Knowledge formation based on the features that may indicate low-rate DDoS attack performed by a botnet;
   - Presentation of the knowledge about the low-rate DDoS attack as a set of feature vectors;
   - Labelling the obtained feature vectors of the low-rate DDoS attack for the purpose of clusters' formation, where each cluster corresponds to some type of the low-rate DDoS attack.
b. The monitoring stage includes the following steps:
   - Gathering the inbound and outbound network traffic;
   - Construction of the feature vectors based on the information obtained from the network, based on the botnet's features and the self-similarity of the traffic, generated by the botnets' low-rate DDoS attack.
c. The detecting stage includes the semi-supervised fuzzy c-means clustering of the obtained feature vectors for the purpose of its assignment to one of the clusters and choosing the proper security scenario for the attacks mitigation.
d. The appliance of the security scenario for the corporate area network's infrastructure. The subject of this paper is to present the approach for the botnet detection of the low rate DDoS attacks via the BotGRABBER system. Let us discuss this step in detail.

4.1. Presentation of the knowledge concerning the botnets' low-rate DDoS attacks the as the set of feature vectors

Let us define the of features, which are to be analyzed to identify the above-mentioned botnets' low-rate DDoS attacks as x_k, where x_1 – transmission protocol;
x_2 – an average payload length per connection;
x_3 – a number of a different size of packets transferred to a total number of frames per connection;
x_4 – a total number of bytes per connection excluding the header;
x_5 – a total number of bytes transmitted per connection;
x_6 – a duration of the connection;
x_7 – a number of bytes transmitted from origin to destination;
x_8 – a number of packages transmitted from origin to destination;
x_9 – a boolean feature that indicates whether the inbound traffic as an associated outbound traffic record;
x_{10} – a duration of the connection, observed from the earliest of the associated inbound or outbound traffic until the end of the latter traffic;
x_{11} – total size for the session in bytes;
x_{12} – total number of packets in the session;
x_{13} – self-similarity of the outbound/inbound packets in the session, determined by examining the variance in size of the outbound/inbound packets using the Hurst coefficient;
x_{14} – velocity of outbound/inbound traffic measured in packets per second;
x_{15} – velocity of outbound/inbound traffic measured in bits per second;
x_{16} – velocity of outbound/inbound traffic measured in bytes per packet;
x_{17} – standard deviation of packet size within the session measured in bytes;
x_{18} – invalid values of TCP flags seen in this session;
x_{19} – the ratio of the number of most common packet.

All aforementioned features are the base of the set of feature vectors $X = \{x_k\}_{k=1}^{Nk}, N_k = 19$, where each of feature vector x_k describes the specified low-rate attack, N-the number of the feature vectors. The main feature, that indicates the presence of the low-rate attack, is the self-similarity of the outbound/inbound packets in the session, determined by examining the variance in size of the outbound/inbound packets using the Hurst coefficient.

4.2. A self-similarity of network traffic and the Hurst coefficient

The main point of the botnets' low rate DDoS attack detection is assign the malicious traffic from legitimate one taking into account the self-similarity features of the attack and normal traffic. For this purpose, the proposed technique estimates the self-similarity features based on H values are used. In general, network traffic can be represented as a fractal–a figure whose small arbitrarily enlarged parts are similar to the base one. In other words, a certain object can be considered as self-similar if there is an exact or approximate coincidence of such an object with a part of itself. A network traffic is able to have the property of self-similarity. It can be manifested as the frequency of received data packets in different time scales, which at different scales looks like a fractal. Because of the self-similarity is a random process the self-similarity degree can be determined by the Hurst coefficient, which is able to analyze the time series during which network traffic was gathered.

In general, if the coefficient H takes a value of 0.5, this indicates that the events are random and there is no long-term dependence between them. In this case, network traffic is not self-similar. If the coefficient H takes values from 0.5 to 1, then this means that the observed time interval is a continuous series of time. Furthermore, the higher the value of the coefficient H, the greater the degree of long-term relationship between events and the greater the degree of self-similarity is observed. When the Hurst coefficient is close to value 1, network traffic takes the maximum value of the degree of self-similarity, which means that with any time series scaling, the frequency of data packets will receive the most similar form. This value is defined as a function of the time interval of the time series as follows:

$$E\left[\frac{R(n)}{S(n)}\right] = Cn^{H}, n \to \infty \tag{1}$$

$R(n)$ is the range of the first n cumulative deviations from the mean; $S(n)$ – a standard deviation; $E[x]$ is the expected value; n – the time span of the observation; C is a constant.

For the most accurate determination of the Hurst coefficient, the time interval should be sufficiently large. Therefore, the effectiveness of detecting low rate DDoS attacks based on the traffic self-similarity significantly depends on the time interval value during which the network traffic collection and analysis was carried out. In order to evaluate the network traffic self-similarity let us define it as a random process, which can be divided into the discrete time intervals as follows $X = (X_1, X_2, \dots)$. If the time intervals are equal to n, then this random process will have the form $X^{(n)} = (X_1^n, X_2^n, \dots)$, whose components are determined by the formula:

$$X_k^{(n)} \triangleq \frac{(X_{kn-n+1} + \cdots + X_{kn})}{n}, n, k \in N \tag{2}$$

To describe the dependence of random processes X and $X^{(n)}$, let us determine the correlation coefficients $r(q)$, which describes the dependence of the process X and the correlation coefficient $r_n(q)$, which describes the process $X^{(n)}$. In the general, the process X can be considered as self-similar if the Hurst coefficient takes values from 0.5 to 1 and the equality is be fulfilled:

$$r_n(q) = r(q), n \in Z, n \in \{2, 3, \dots\} \tag{3}$$

In this case, the self-similar process X is very similar to the process $X^{(n)}$, since the correlation coefficient $r(q)$ isn't changed after the time scaling of length n is performed. This means that the frequency of the received data packets for a certain time interval takes approximately the same form after the scaling was carried out.

In order to determine the Hurst coefficient, let us divide the length of the network traffic for fixed time intervals. To describe the time of arrival of traffic, let us define the time domain T, which is considered as an independent variable for the analysis of time phenomena. Obtained time intervals Xi are described as follows $X = (X_i \mid i = 0,1,2, \dots)$, where X is the total time of the traffic monitoring. Let us define the mean value of the packet receiving frequency as μt. The description of the value of the difference between the maximum and minimum frequencies for each of the time intervals can determined as the function $R(T)$ as [26]:

$$R(T) = max\, X\,(t, T) - min\, X(t, T),\ where\, 1 \le t \le T \tag{4}$$

To describe the average deviation of the data packets frequency from the mean value of the frequency, let us determine the mean square deviation $S(T)$, which is determined by the formula:

$$S(T) = \{\frac{1}{T}\sum_{t=1}^{T}[X_t - \mu_t]^2\}^{\frac{1}{2}}, where X(t,T) = \sum_{i=1}^{t}[X_i - \mu_t] \qquad (5)$$

In this case, the ratio of $\frac{R(T)}{S(T)}$ becomes:

$$\frac{R(T)}{S(T)} = cT^H \sim T^H, T \to \infty \qquad (6)$$

where H – the Hurst coefficient, c – constant. Then the Hurst coefficient will be evaluated as follows:

$$H = \frac{ln(\frac{R(T)}{S(T)})}{ln\, T} - \frac{ln\, c}{ln\, T} \qquad (7)$$

In the general, in order to determine the self-similarity rate, we are to calculate the value of the function $R_i(n)$ and the standard deviation for each of the time intervals of the length N. Further, for each of the time intervals the ratio $\frac{R_i(N)}{S_i(N)}$ as well as the mean value of $\frac{R(N)}{S(N)}$ are to be calculated, herewith:

$$\frac{R(N)}{S(N)} = \frac{1}{k}\sum_{i=1}^{k}\frac{R_i(N)}{S_i(N)} \qquad (8)$$

It is also worth mentioning, that the increasing the value of N, leads to recalculation of the formula (8) and the Hurst coefficient using the formula (7), since the change in the number of investigated time slots leads to the recalculation of the Hurst coefficient, and the new value of the degree of self-similarity of traffic. The constructed feature vector, which includes the H value, is to be clustered the semi-supervised fuzzy c-means clustering, where each cluster corresponds to the specified cyberattacks (and the security scenario to be applied) and one cluster corresponds to the absence of the attack [26-27].

5. EXPERIMENTS

5.1. Evaluation settings

In order to evaluate the efficiency of the approach for the detection of the botnets' low-rate DDoS attacks, a detection accuracy tests using real world network traffic were carried out. For this purpose, a Slowloris and R.U.D.Y attacks [4-5] were employed. The main abilities of the tools are the generating malicious low-rate attacks. On other hand, experiment included generated real traffic that mimics users' behavior (e.g. SSH, HTTP, and SMTP) using the malicious traffic dataset [28].

To carry out experiments, the university local area network of hosts including 50 hosts (hosts with Microsoft Windows operating system), one dedicated server (Linux openSUSE operating system with nginx HTTP server) and network devices (MikroTik CCR1009-8G-1S-1S+PC routers) was employed. Network traffic was captured by the means of tcpdump utility. All experiments were organized in real time and real networks, and lasted during from several seconds to one hour. To carry out the experiments, an attack on the mentioned web server was attacked by different attacks with different set of parameters.

The main parameters of low-rate DDoS-attack (e.g. R.U.D.Y. attack) are: a number of network connections to the server; a value of the Content-Length field of the corresponding POST HTTP requests; a frequency of sending packets from each open connection. The parameters of DDoS-attacks as in the case of the R.U.D.Y. attack used for conducting experiments are presented in Table 1.

Table 1. The main parameters of the low-rate DDoS-attack (e.g. R.U.D.Y. attack)

Parameter	Value
Number of network connections to the server	10500-1500
Content-Length, bytes	5100-11300
The frequency of sending packets, ms	3-15

The set of parameters involved into the traffic self-similarity detection are:

- Total time duration, *T*, sec;
- Number of the time intervals, *I*;
- Number of the data packets in each time interval, $k_1 \ldots k_i$;
- Scaling coefficient, *c*.

5.2. Results

The result of the experiment, which include different sets of parameters for malicious traffic samples are presented in the Table 2. As a data samples of the low-rate DDoS attacks the traffic samples, which include the self-similarity property, were used. Examples of the results for five different samples are presented in the Table 2. The results of the experiment, demonstrated that the obtained values of the Hurst coefficients for different malicious samples varied depend on the different parameters. Thus, the number of the time intervals affected greatly.

The largest values for the Hurst parameters were obtained when the total time duration was higher. At the same time, the influence of the Hurst coefficients de-creased when the total time number was lower and number of the time intervals was higher. For mentioned five samples the highest values of the Hurst coefficients were in the range of 0.713..0.804. It indicated that the traffic generated by low-rate DDoS attacks was self-similar, and it had made it possible to detect the malicious traffic of data packets among normal traffic. The results of the low-rate DDoS attacks detection via BotGRABBER with and without network traffic self-similarity analysis s presented in the Table 3, where the overall accuracy is 97.46% and 90.06% respectively. Thus, proposed approach is acceptable for its involvement into the BotGRABBER botnet detection system as the engine for low-rate attacks detection unit.

Table 2. The results of the obtained Hurst coefficients

Malicious traffic samples	Total time duration, sec	Number of the time intervals	Number of the data packets, $k_1 \ldots k_i$, bytes	Total time duration, sec	Total time duration, sec
Sapmle1	150	10	~8300	1.7	0.582..0.600
		100			0,627..0.659
		500			0.759..0.767
		1000			**0.753..0.784**
Sample2	150	10	~7500	1.7	0.564..0.592
		100			0.621..0.646
		1000			0.714..0.729
		10			**0.741..0.763**
Sample3	150	10	~11300	1.7	0.549..0.552
		100			0.628..0.662
		500			0.681..0.709
		1000			0.782..0.804
Sample4	150	10	~5100	1.7	0.530..0.545
		100			0.615..0.640
		500			**0.701..0.713**
		1000			0.671..0.680
Sample5	150	10	~5300	1.7	0.429..0.452
		100			0,621..0.540
		500			**0.693..0.707**
		1000			0.601..0.687

Table 3. Test results for low-rate DDoS attacks including sensitivity, specificity, overall accuracy, true positives (TP), true negatives (TN), false positives (FP), false negatives (FN)

	Number of malicious traffic samples	Evaluation set				Results			
		Malicious traffic samples		Benign traffic samples		Sensitivity, %	Specificity, %	Overall accuracy, (with network traffic self-similarity analysis), %	Overall accuracy, (without the network traffic self-similarity analysis), %
		TP	FN	TN	FP				
Low-rate DDoS attacks	1687	661	16	489	14	97.64	97.22	**97.46**	**90.06**

6. DISCUSSION

As BotGRABBER system involves the network traffic self-similarity analysis for the botnets' low-rate a DDoS-attacks detection there are several factors, which may affect the prediction accuracy. One of them is the diversity of training samples. Most conspicuously, that not all possible feature vectors, that describe different low-rate a DDoS-attacks, are may be adequately represented in the training set. Thus, system may be further improved by choosing more refined set of malicious traffic samples for different types of the low-rate DDoS-attacks.

The experiments demonstrated, that the BotGRABBER is able to achieve acceptable detection results, but the efficiency of the detection may be decreased in because the traffic flow of some attacks is very similar to users' ones and some of botnets' features were not taken into account for the detection process. On the other hand, the main to detect low-rate a DDoS-attack the system has to evaluate result taking into account several parameters with different values real time (various values of the total time duration, the number of the time intervals, the number of the data packets in each time interval, change the scaling coefficient, which leads to computational growth.

7. CONCLUSION

The article presents the approach for the botnets' low-rate a DDoS-attacks detection based on the self-similarity of network traffic. Proposed technique is the part of bot-nets detection system–BotGRABBER system. The novelty of the approach is that the low-rate DDoS-attacks detection is based on the network analysis concerning its self-similarity which is defined with the use of Hurst coefficient and the features inherent to the botnets. Experimental research demonstrated, that the Hurst parameters for the network traffic self-similarity analysis (the range of 0,713..0,804) were defined correctly, that made it possible to detect the low-rate a DDoS-attacks with high accuracy. Experimental research proved that the involvement of the network traffic self-similarity analysis is able to increase the botnet's detection efficiency up to 97%.

ACKNOWLEDGEMENTS

We thank the Khmelnytskyi National University for providing access to local network during the performance of the experimental research.

REFERENCES

[1] A. Dehghantanha, M. Conti, and T. Dargahi, "Cyber threat intelligence," *Springer International Publishing*, 2018.

[2] T. Kumar, S. Sharma, R. Dhaundiyal, and P. Jain, "Investigation of malware and forensic tools on internet," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 5, pp. 3179-3186, 2018.

[3] S. Alam, M. Muqeem, and S. A. Khan, "Review on security aspects for cloud architecture," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 5, pp. 3129-3139, 2018.

[4] CLOUDFLARE, "Low and slow attack," [Online], Available at: https://www.cloudflare.com/learning/ddos/ddos-low-and-slow-attack/ [accessed March 6, 2019].

[5] Imperva Incapsula, "R.U.D.Y. (R-U-Dead-Yet?)," [Online] Available at: https://www.incapsula.com/ddos/attack-glossary/rudy-r-u-dead-yet.html [accessed March 6, 2019].

[6] Radware, "DDoS attack definitions-DDoSPedia," [Online] Available at: https://security.radware.com/ddos-knowledge-center/ddospedia/slow-rate-attack/ [Accessed March 6, 2019].

[7] Mahadev, V. Kumar, and K. Kumar, "Classification of DDoS attack tools and its handling techniques and strategy at application layer," *International Conference on Advances in Computing, Communication and Automation, ICACCA*, pp. 1-6, 2016.

[8] S. Behal and K. Kumar, "Characterization and comparison of DDoS attack tools and traffic generators-a review," *Int. J. Network Security*, vol. 19, no. 3, pp. 383-393, 2017.

[9] N. Muraleedharan and B. Janet, "Behaviour analysis of HTTP based slow denial of service attack," *International Conference on Wireless Communications, Signal Processing and Networking, WiSPNET*, pp. 1851-1856, 2018.

[10] K. Hong, Y. Kim, H. Choi, and J. Park, "SDN-assisted slow HTTP DDoS attack defense method," *IEEE Communications Letters*, vol. 22, no. 4, pp. 688-691, 2018.

[11] T. Mahjabin, Y. Xiao, G. Sun, and W. Jiang, "A survey of distributed denial-of-service attack, prevention, and mitigation techniques," *International Journal of Distributed Sensor Networks*, vol. 13, no. 12, pp. 1-33, 2017.

[12] T. Hirakawa, K. Ogura, B. B. Bista, and T. Takata, "An analysis of a defense method against slow HTTP DoS attack," *International Symposium on Information Theory and Its Applications, ISITA*, pp. 316-320, 2019.

[13] T. Lukaseder, L. Maile, B. Erb, F. Kargl, "SDN-assisted network-based mitigation of slow DDOS attacks," *International Conference on Security and Privacy in Communication Systems, Springer*, Cham, pp. 102-121, 2018.

[14] S. Tayama and H. Tanaka, "Analysis of slow read DoS attack and communication environment," *International Conference on Mobile and Wireless Technology, Springer*, Singapore, pp. 350-359, 2017.

[15] R. K. Deka and D. K. Bhattacharyya, “Self-similarity based DDoS attack detection using Hurst parameter,” *Security and Communication Networks*, vol. 9, no. 17, pp. 4468-4448, 2016.

[16] R. Doshi, N. Apthorpe, and N. Feamster, “Machine learning DDoS detection for consumer internet of things devices,” *IEEE Symposium on Security and Privacy Workshops*, pp. 29-35, 2018.

[17] N. S. Selamat and F. H. M.Ali, “Comparison of malware detection techniques using machine-learning algorithm,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 16, no. 1, pp. 435-440, 2019.

[18] K. S. Yin and M. A. Khine, “Optimal remote access Trojans detection based on network behavior,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 3, pp. 2177-2184, 2019.

[19] S. Bravo and D. Mauricio, “Systematic review of aspects of DDoS attacks detection,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 14, no. 1, pp. 162-176, 2019.

[20] A. Saravanan, S. SathyaBama, Seifedine K., and L. Kumar Ramasamy, “A new framework to alleviate DDoS vulnerabilities in cloud computing,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 5, pp. 4163-4175, 2019.

[21] O. Savenko, S. Lysenko, and A. Kryschuk, “Multi-agent based approach of botnet detection in computer systems,” *International Conference on Computer Networks, Springer*, Berlin, Heidelberg, pp. 171-180, 2012.

[22] S. Lysenko, O. Pomorova, O. Savenko, A. Kryshchuk, and K. Bobrovnikova, “DNS based anti-evasion technique for botnets detection,” *8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, vol. 1, pp. 453-458, 2015.

[23] O. Pomorova, O. Savenko, S. Lysenko, A. Kryshchuk, and K. Bobrovnikova, “Anti-evasion technique for the botnets detection based on the passive DNS monitoring and active DNS probing,” *International Conference on Computer Networks: Springer International Publishing, Springer*, Cham, pp. 83-95, 2016.

[24] S. Lysenko, O. Savenko, K. Bobrovnikova, A. Kryshchuk, and B. Savenko, “Information technology for botnets detection based on their behaviour in the corporate area network,” *International Conference on Computer Networks, Springer*, Cham, pp. 166-181, 2017.

[25] S. Lysenko, O. Savenko, K. Bobrovnikova, and A. Kryshchuk, “Self-adaptive system for the corporate area network resilience in the presence of botnet cyberattacks,” *Communications in Computer and Information Science*, pp. 385-401, 2018.

[26] Chen Jian, Tan Xianhai, and Jia Zhen, “Performance analysis of seven estimate algorithms about the Hurst coefficient,” *Journal of Computer Applications*, 2006.

[27] O. Pomorova, O. Savenko, S. Lysenko, and A. Nicheporuk, “Metamorphic viruses detection technique based on the the modified emulators,” ICTERI, vol. 1614, pp. 375-383, 2016.

[28] Canadian Institute for Cybersecurity, “Botnet dataset,” University of New Brunswick, [Online], Available at: https://www.unb.ca/cic/datasets/botnet.html [access January, 2019].

BIOGRAPHIES OF AUTHORS

Sergii Lysenko is Associate Professor of the Department of Computer Engineering and System Programming, Khmelnytsky National University. He earned his B.Eng. Degree in Khmelnytsky National University in 2005 and his PhD Degree in Ternopil National Economic University in 2011. His main research interests are Self-adaptive detection systems for cyber-threats in computer networks, Methods of detecting cyberattacks in corporate networks, malware detection. Email: sirogyk@ukr.net.

Kira Bobrovnikova is Associate Professor of the Department of Computer Engineering and System Programming, Khmelnytskyi National University. She earned her M.Eng. Degree in Khmelnytsky National University in 2013 and her PhD Degree in Ternopil Ivan Puluj National Technical University in 2017. Her main research interests are network security, malware analysis and malware detection systems in corporate area networks.

Ivan Hurman is Associate Professor of the Department of Software Engineering, Khmelnytskyi National University. He earned his M.Eng. Degree in Khmelnytsky National University in 2005 and his PhD Degree in Khmelnytsky National University in 2013. His main research interests are network security and malware analysis.

Serhii Matiukh is vice-rector on scientific and pedagogical work of Khmelnytskyi national university, Assistant Professor, Ph.D. Associate Professor of the International economic Department of Computer Engineering, Khmelnytsky National University. His main research interests are the conceptual basics of efficiency formation in functioning of higher educational institutions.

Oleg Savenko is Professor and Dean of the Faculty of Programming and Computer and Telecommunication Systems, Khmelnytsky National University. He earned his B.Eng. Degree in Kamyanets-Podilsky State Pedagogical Institute in 1993 and his PhD Degree in Vinnitsa State Technical University in 1999. His main Areas of Research Interest are Methods for malware detecting, Operating Systems and Artificial Intelligence.