Technique for IoT Cyberattacks Detection Based on DNS Traffic Analysis

Kira Bobrovnikova[1[0000-0002-1046-893X]], Sergii Lysenko[1[0000-0001-7243-8747]], Piotr Gaj[2[0000-0002-2291-7341]], Valeriy Martynyuk[1[0000-0001-5758-4244]], Dmytro Denysiuk[1]

[1]Khmelnitsky National University, Khmelnitsky, Ukraine
[2] Silesian University of Technology
bobrovnikova.kira@gmail.com
sirogyk@ukr.net,
piotr.gaj@polsl.pl
martynyuk.valeriy@gmail.com
web.developer.den@gmail.com

Abstract. The dynamic growth of the number of cyberattacks, which perform destructive against the IoT devices, forces the developers of anti-virus software to implement new methods and algorithms for their search and disposal. The existing statistics prove the need of the novel cyberattacks detection approaches development. The paper presents a new technique for IoT cyberattacks detection based on DNS traffic analysis is presented. The method allows detecting IoT botnet cyberattacks. The method has the heuristic and proactive nature. It is based on the gathering of the set of features that may indicate the IoT cyberattacks presence. The mechanism of attack detection system is based on the cyberattacks' features gathering from network and feature vectors construction. As the classification algorithms a semi-supervised fuzzy c-means clustering, SVM and Artificial Immune System classification algorithms were employed.

Keywords: Internet of Things, Cyberattack, DNS, Network traffic, Network, Cybersecurity, Computer system, Host, Malicious traffic, Attacks Detection

1 Introduction

Every year, new types of devices are used in the Internet of Things (IoT) market: home automation, smart cities, medicine, and agriculture. The devices' firmware is being developed without taking into account the latest cybersecurity requirements [1] and, many IoT devices manufacturers are striving to make their products as cheap and expedite as possible by simplifying security features [2].

Despite the small computing power of individual IoT devices, their sheer number, combined into a single malicious bot-managed network, poor security (or even lack thereof) and permanent Internet connection make them a convenient tool for organizing powerful cyberattacks [1].

Malicious traffic volume generated by IoT botnets is usually much higher than the of botnet' traffic volume generated from personal computers [3].

Thus, cyberattacks against the IoT devices are a major important cybersecurity problem because they are difficult to detect, localize and mitigate [3].

2 Related work

In [4] various IoT botnet detection approaches are discussed. In [5] the Rustock IoT botnet which employs the evasion technique fast-flux to communicate with its bots and command and control (C&C) centers is investigated. For its detection the set of features were analyzed and number of classifiers were used.

In paper [6] an approach for botnet detecting of the on activity within consumer IoT devices and networks was presented. As a tool of making conclusion the kind of neural network (with the bidirectional long short term memory) was involved. As a tool of the communication detection between attackers the word embedding packets were employed. The proposed technique was compared with other ones, based on the usage of other kinds of neural networks. In order to determine the effectiveness of the detection, the Mirai IoT botnet was used. Experimental results demonstrated that the bidirectional approach increased the detection time, but improved its efficiency.

In [7] the functioning of the Mirai IoT bontet is presented. It is also demonstrated an approach for its detection using the network analysis.

In [8] a new behavior-based approach for DDoS detection in IoT network traffic was presented. It describes the specific IoT network's features, that may indicate the attacks presence in the network.

In [9] a novel IDS able to detect DNS IoT botnets' attacks. It is an effective mitigation tool against the attacks performed by the IoT botnets. Technique involves the detection of IoT attacks that employ DNS, HTTP and MQTT protocols. It is based on statistical processing and uses machine such learning algorithms as Artificial Neural Network, Naive Bayes, and decision tree. The experimental cases with usage of the known botnet datasets were presented.

In [10] a new approach IoT botnet DDoS attack detection which is able to mitigate the cyberattacks is presented. It is an event management-based approach and enables the possibility of the DDoS attack blocking. Approach monitors the network traffic concerning the compromised IoT devices taking into account specific network features.

In [11] a novel IoT botnet detecting approach based on the usage of the machine learning algorithms is presented. It is able to identify the botnet cyberattacks performed using the infected IoT devices. The detection process involved the "Grey Wolf" algorithm as well as the SVM and demonstrated promising results.

In [12] the aspects of the IoT cybersecurity concerning the smart cities infrastructure are presented. Furthermore, a new anomaly-based technique for the IoT attacks detection is proposed. It uses the Random Forest algorithms and demonstrated good detection effectively concerning infected IoT devices.

In [13] a new technique for DDoS detection was presented. It described the infected IoT devices' network traffic generation. Based on this, a new approach for anomaly detection was produced.

Nevertheless, the mentioned above approaches have common drawbacks: they don’t take into account a set of techniques that may be used by IoT botnets to perform the cyberattacks such as cycling of IP mapping, domain flux, fast flux, and DNS tunneling. In addition, techniques demonstrate low IoT botnets detection efficiency and have high false positives rate.

3 Technique for IoT Cyberattacks Detection Based on DNS Traffic Analysis

DNS is widely used to establish links between IoT botnets’ bots and their command and control centers (C&C) attackers [5]. It makes it possible to control the IoT botnet anonymously and flexibly. Various complex techniques are used to avoid the C&C servers tracking through DNS: cycling of IP mapping, domain flux, fast flux, and DNS tunneling [1, 2, 5].

In order to solve this problem, a new technique for IoT cyberattacks detection based on DNS traffic analysis was proposed. It is based on the detection of the IoT botnets’ communication with C&C over DNS protocol, and consists of steps:

1. Gathering of the incoming DNS traffic of the IoT network.
2. Domain names’ "white" and "black" lists checking.
3. DNS traffic features extraction that may indicate the malicious botnets activity in the IoT network.
4. Feature vectors analysis.
5. Localization and blocking of the infected IoT devices.

Let us present the IoT botnets detection process of based on the DNS traffic analysis as a tuple $M_{BN} = \left\langle \left\langle C, A, B, \Psi, Z, L, F \right\rangle, \chi^T, \vartheta_1^T, \vartheta_2^T, \Upsilon^T, \vartheta_3^T, T \right\rangle$, where $C = \left\{c_j\right\}_{j=1}^{N_C}$ - a set of botnet’s elements, N_C - the number of botnet’s elements; $A = \left\{a_j\right\}_{j=1}^{3}$ - IoT botnet architecture type; $B = \left\{b_j^p\right\}_{j=1}^{N_B}$ - a set of network protocols used to manage the IoT botnet, N_B - number of network protocols, $p \in P$, P={1.65535} - a set of ports used for the IoT botnet management; $\Psi = \left\{\psi_j\right\}_{j=1}^{4}$ - a set of evasion techniques of IoT botnet based on DNS; $Z = \left\{z_j\right\}_{j=1}^{N_Z}$ - a set of IoT devices infected by botnet, N_Z - a number of infected IoT devices; $L = \left\{l_j\right\}_{j=1}^{5}$ - a set of botnet’s life cycle stages; l_1 - infection; l_2 - initial registration or connection; l_3 - implementation of the malicious activity; l_4 - technical support; l_5 - termination of the botnet; stages l_2 - l_4 occur with the involvement of DNS; $F = \{f_j\}_{j=1}^{N_F}$ - the set of bot functions of the IoT botnet, determined by the corresponding botnet’s life cycle stage, N_F - the number of bot functions of the botnet IoT; IoT device infection function $l_1 \Rightarrow \Upsilon \xrightarrow{f_1} \{h_{\inf} | h_{\inf} \in H\}$, where Y - a set of botnet’s malicious actions, H - a set of infected IoT devices in the network, $h_{\inf}$ - an infected IoT device; the connecting function of the infected IoT device to botnet $l_2 \Rightarrow Z \cup \{h_{\inf} | h_{\inf} \in H\} \xrightarrow{f_2} Z'$; IoT botnet upgrade function to a new version $l_3 \Rightarrow z \times z' \xrightarrow{f_3} z'$; the set of botnet’s malicious commands $l_4 \Rightarrow Z \times \{p | p \in P\} \xrightarrow{f_4} \Upsilon$, where P - a set of commands that can be executed by bots of the IoT botnet; the deactivation function of the IoT botnet $l_5 \Rightarrow Z \setminus \{z | z \in Z\} \xrightarrow{f_5} Z'$; χ^T - the set of captured incoming DNS messages addressed to the set of network IoT devices H; ϑ_1^T - a function of domain names comparison with the "white" and "black" lists; ϑ_2^T - a function of the feature extraction from incoming DNS traffic, indicating the presence of malicious activity of the IoT botnets; Υ^T - a set of IoT botnets’ detecting algorithms based on the DNS traffic analysis; ϑ_3^T - the localization function of infected IoT devices, and blocking the bots’ actions; $T = \{t_m\}_{m=0}^{N_T}$ - the observation time interval, where N_T - the number of iterations of the observation.

Let present the command and control elements of an IoT botnet as $C = \{c_j\}_{j=1}^{N_C} = \{\langle\langle D, I\rangle, \langle N, E\rangle\rangle_j\}_{j=1}^{N_C}$, where $D = \{d_j\}_{j=1}^{N_D}$, $I = \{i_j\}_{j=1}^{N_I}$ - a set of domain names and IP addresses of IoT botnet control elements for d; $N = \{n_j\}_{j=1}^{N_N}$, $E = \{e_j\}_{j=1}^{N_E}$ - a set of domain names and IP addresses of authority name servers for d; N_D - a number of domain names corresponding to the controlling elements of the IoT botnet; N_I - the number of IPs mapped to domain names; N_N - the number of domain names of authority name servers; N_E - the number of IP addresses of authority name servers.

Let's present the type of IoT botnet architecture as $A = \{a_j\}_{j=1}^{3}$, where a_1 - centralized, a_2 - distributed, a_3 - hybrid.

Let us consider the steps of the method in more detail.

3.1 Gathering the Incoming DNS Traffic of the IoT Network

Let us represent DNS traffic as a tuple $\chi_N = \langle \chi, H, S, D \rangle$, where χ - the set of DNS messages sent from and to the set IoT network devices *H*, $\chi = \chi^O \cup \chi^I$, where χ^O - a set of outcoming DNS messages, χ^I - a set of incoming DNS messages; S - a set of DNS servers, $S = S^L \cup S^N$, where S^L - a set of local DNS servers, S^N - a set of non-local DNS servers; *D* - the set of requested domain names by IoT devices, $D = \{d_i\}_{i=1}^{N_D}$, where N_D - number of different domain names.

Let us present a set of IoT devices that have made DNS requests as $H = \bigcup_{j=d_1}^{d_{N_D}} \bigcup_{k=1}^{N_{TTL}} H_{j,k}$, where H_j - a subsets of MAC addresses of IoT devices that have sent DNS requests for a specific domain name; $H_{j,k}$ - subsets of MAC addresses of IoT network devices that have sent DNS requests to a specific domain name within a specific TTL period; N_{TTL} - the total number of such subsets; $H_{j,k} = \{h_{j,k,i}\}_{i=1}^{N_{H,j,k}}$, where $h_{j,k,i}$ - MAC address of a specific IoT network device; $N_{H,j,k}$ - the number of network IoT devices that have sent DNS requests within a specific TTL period.

Let us present the set of captured DNS messages as $\chi^T = \bigcup_{j=d_1}^{d_{N_D}} \bigcup_{k=1}^{N_{TTL}} \chi_{j,k}$, where χ_j - subsets of incoming DNS messages for a specific domain name; $\chi_{j,k}$ - subsets of incoming DNS messages for a specific domain name captured within a specific TTL period; $\chi_{j,k} = \{\chi_{j,k,i}\}_{i=1}^{N_{\chi,j,k}}$, where $\chi_{j,k,i}$ - DNS message captured within a TTL period, $N_{\chi,j,k}$ - the number of DNS messages captured within a TTL period.

Employing the incoming DNS message structure [14], let us present the captured DNS response for a specific domain name as a tuple $\chi_{j,k,i} = \langle \chi_{j,k,i,\mathrm{H}}, \chi_{j,k,i,TS}, \chi_{j,k,i,IP}, \langle \chi_{j,k,i,HD}, \chi_{j,k,i,ANS}, \chi_{j,k,i,ATH}, \chi_{j,k,i,ADD} \rangle \rangle$, $j = d_1, ..., d_{N_D}, k = \overline{1, N_{TTL}}, i = \overline{1, N_{\chi,j,k}}$, where $\chi_{j,k,i,\mathrm{H}}$ - MAC address of the IoT device that perform the DNS request; $\chi_{j,k,i,TS}$ - a time stampt of DNS packet; $\chi_{j,k,i,IP}$ - DNS packet source IP address; $\chi_{j,k,i,HD}, \chi_{j,k,i,ANS}, \chi_{j,k,i,ATH}, \chi_{j,k,i,ADD}$ - DNS message sections: Header, Answer, Authority, and Additional respectively.

The DNS message header can be presented as follows:

$\chi_{j,k,i,HD} = \left\langle \chi_{j,k,i,HD,ID}, \chi_{j,k,i,HD,OPC}, \chi_{j,k,i,HD,RC}, \chi_{j,k,i,HD,QDC}, \chi_{j,k,i,HD,ANC}, \chi_{j,k,i,HD,NSC}, \chi_{j,k,i,HD,ARC} \right\rangle, j = d_1, ..., d_{N_D}, k = \overline{1, N_{TTL}}, i = \overline{1, N_{\chi,j,k}}$, where $\chi_{j,k,i,HD,ID}$ - an identifier that allows associating a DNS request with DNS response (ID field); $\chi_{j,k,i,HD,OPC}$ - a request type (OPCODE field); $\chi_{j,k,i,HD,RC}$ - response code; $\chi_{j,k,i,HD,QDC}$ - number of entries in the query section; $\chi_{j,k,i,HD,ANC}, \chi_{j,k,i,HD,NSC}, \chi_{j,k,i,HD,ARC}$ - the number of resource records in the header, nameservers and additional information sections (fields ANCOUNT, NSCOUNT, ARCOUNT), respectively.

The Answer, Authority, and Additional sections have the same format and can be described as a set of the resource records as follows: $\chi_{j,k,i,S} = \left\{ \left(\chi_{j,k,i,S,NM}, \chi_{j,k,i,S,TP}, \chi_{j,k,i,S,TTL}, \chi_{j,k,i,S,RDL}, \chi_{j,k,i,S,RDT} \right)_n \right\}_{n=1}^{N_{RR,S}}$ $j = d_1, ..., d_{N_D}, k = \overline{1, N_{TTL}}, i = \overline{1, N_{\chi,j,k}}$, where $S \in \{ "ANS", "ATH", "ADD" \}$, $\chi_{j,k,i,S,NM}$ - NAME field; $\chi_{j,k,i,S,TP}$ - TYPE field; $\chi_{j,k,i,S,TTL}$ - TTL field; $\chi_{j,k,i,S,RDL}$ - RDATA field length; $\chi_{j,k,i,S,RDT}$ - RDATA field value; $N_{RR,S}$ - the number of resource records in the section (equal to $\chi_{j,k,i,HD,ANC}, \chi_{j,k,i,HD,NSC}, \chi_{j,k,i,HD,ARC}$ for the relevant section).

3.2 Usage of "white" and "black" Domain Names Lists

In order to detect DNS requests to known domain names of the IoT botnets and to reject legitimate DNS requests, the requested domain names of the IoT devices are compared with "white" and "black" domain names lists.

3.3 IoT malicious traffic extraction

At this stage, the inbound DNS messages are to be analyzed, and the features that may indicate the malicious IoT botnets activity are to be extracted.

Let us define a set of IoT botnets evasion techniques as $\Psi = \left\{ \psi_j \right\}_{j=1}^{4}$, where ψ_1 - cycling of IP mapping, ψ_2 – domain flux, ψ_3 - fast flux, ψ_4 - DNS tunneling.

If IoT botnet uses cycling of IP mapping C&C server $c \in C$ periodically changes its location, and the domain name d is associated with the C&C server is mapped to the some IP address from the set $i \in I$, $d \to \{i_1,...,i_n\}$. Botnet's architecture type is centralized, $\psi_1 \Rightarrow a_1$.

Let us define a set of features that indicate the usage cycling of IP mapping technique by IoT botnet as $G\Psi_1 = \{t_{mod}, t_{med}, t_{aver}, n_{IP}, s_{IP}\}$, where t_{mod} - t_{med} t_{aver} - TTL-period (mode, median, average respectively); n_{IP} and s_{IP} - the number of IPs and the average distance between IPs associated with the domain name respectively.

When IoT botnet uses domain flux technique, the C&C server $c \in C$ periodically migrates to the new domain names from a list formed using the domain name generation algorithm (DGA). Thus, within specified TTL period a new name $d \in D$ may correspond to IP address of the C&C server $i \in I$, $\{i\} \to \{d_1,...,d_n\}$. If the C&C server also changes location, then $\{i_1,...,i_n\} \to \{d_1,...,d_m\}$. Botnet architecture type is centralized, $\psi_2 \Rightarrow a_1$.

Let us define a set of features that indicate the use of "domain flux" evasion technique as $G\Psi_2 = \{t_{mod}, t_{med}, t_{aver}, f_S, n_D\}$, where f_S - binary sign of success of DNS request; n_D - the number of domain names with shared IP addresses.

Within the time interval defined by the TTL DNS period, a single-flux network domain name *d*, which is used to connect with the infected IoT devices to control elements $\{c_1,...,c_n\}$, is mapped to a new set of IPs. These IPs are changing cyclically $d \to \{i_1,...,i_n\}$. Also, the IP addresses are geographically distributed by the infected botnet's nodes that redirect traffic to the control elements $\{c_1,...,c_n\} := \{x|x \in Z \wedge x \in C\}$. For the double-flux network the domain name of each authority name server *n* is matched to a subset of cyclically changing IPs $d \to \{i_1,...,i_n\}$, $n \to \{e_1,...,e_m\}$. These IPs are also the IP addresses of the geographically distributed infected botnet's nodes, i.e. $\{n_1,...,n_m\} := \{x|x \in Z \wedge x \in N\}$. As the number of name servers for such botnets is usually more than one, then $\{n_1,...,n_m\} \to \{e_1,...,e_n\}$. Botnet's architecture type is distributed, $\psi_3 \Rightarrow a_2$.

Let us define the set of features that indicate the usage fast-flux technique changing as $G\Psi_3 = \{t_{mod}, t_{med}, t_{aver}, n_A, s_A, n_{UA}, s_{UA}\}$, where n_A - the number of A-records corresponding to the domain name in the incoming DNS message; s_A, n_{UA} and s_{UA} - the average distance between IPs, the number of unique IPs and the average distance between unique IPs in multiple A-records corresponding to a domain name in incoming DNS messages respectively.

Attacker uses the DNS tunneling to transmit C&C traffic to a fake DNS server. It enables the possibility of the IoT infected devices to send the encrypted messages to the attacker's server and receive the commands from him. In this case, the set of domain names *D* actually is an analogue of the domain names of the C&C server of the

IoT botnet. IP address e of the fake DNS server usually stays stable, that is $\{d_1,...,d_n\} \to e$. Type of botnet architecture - centralized or hybrid, $\psi_4 \Rightarrow a_1 \vee a_3$.

Let us define a set of features of DNS tunneling as $G_{\Psi_4} = \{l_N, n_U, e_N, e_{R,} f_{UR}, l_P\}$, where l_N - a length of the domain name; n_U - a number of unique characters in the domain name; e_N - a domain name entropy; e_R - a maximum entropy value of DNS resource records contained in DNS messages; f_{UR} - a sign of a rare DNS records usage; l_P - an average size of DNS messages for a domain name.

Let us define a set of features that can obtained by the usage of the active DNS probing as $G_\Psi = \{n_{NS}, s_{NS}, v_{retry}, n_{ASN}, n_{ASA}\}$, where n_{NS} - the number of NS records in the DNS response; s_{NS} - the average distance between IPs for multiple NS records for a domain name; v_{retry} - the value of the retry field obtained in the DNS response of the SOA request; n_{ASN} - the number of different ASNs for name servers' IPs; n_{ASA} - the number of different ASNs for domain name.

From these features extracted from the incoming DNS traffic, feature vectors are generated for each domain name requested by the network IoT devices:

$$\overline{W_d} = \{t_{mod}, t_{med}, t_{aver}, n_{IP}, s_{IP}, n_A, s_A, n_{UA}, s_{UA}, l_N, n_U, e_N, e_{R,} f_{UR}, l_P, f_S, n_D\} . \quad (1)$$

3.4 The Feature Vectors Analysis

The feature vectors obtained after the feature gathering and extraction are to be assigned to specified classes. The results of classification are the memberships of the feature vectors to IoT botnet malicious classes or benign class of uninfected IoT devices. The task of classification can be described as a function $f_{classifier} : \overline{W_d} \to \varsigma$, where $f_{classifier}$ - a classification function, ς - IoT botnet malicious class or benign class of uninfected IoT devices.

3.5 Localization and Block of the Infected IoT Devices

Localization and blocking of IoT devices infected with botnets is performed based on the log files analysis that contain lists of domain names requested by IoT devices of the network and MAC addresses of these devices.

4 Experiments

In order to determine the efficiency of the proposed technique a number of experiments were carried out. As the classification algorithms a semi-supervised fuzzy c-means clustering [15-17], Support Vector Machine (SVM) [18], Artificial Immune System (AIS) [19] classification algorithms were used. For the purpose of training the classifiers 16804 samples of the labeled DNS data of the real modern normal traffic and synthetic contemporary IoT botnet attack traffic of two data set were used: BoT-IoT dataset [20, 21] and the UNSW-NB15 dataset [22].

In order to compare the effectiveness of the classification algorithms the test samples of IoT malware infections and IoT benign traffic from IoT-23 dataset [23] were employed. The test data contains 32 415 samples of IoT DNS traffic. 15611 DNS samples of them were the samples of DNS traffic flows of different version's IoT botnets, in particular such as Mirai, Torii, IoT Trojan, Kenjiro, Okiru, Haji me and other [24, 25, 26, 27]. Also test data contains 16804 samples of uninfected IoT devices.

Test result of experiments are presented in the Table 1.

Table 1. Test result of experiments: true positives (TP), false positives (FP)

Botnet`s name	Number of malicious DNS samples	Semi-supervised fuzzy c-means clustering		Support Vector Machine		Artificial Immune System	
		TP	FP	TP	FP	TP	FP
Mirai	2308	2195	94	2236	52	2213	81
Linux Mirai	1795	1709	74	1741	49	1691	50
Torii	1386	1318	32	1331	24	1321	34
IoT Trojan	1762	1674	64	1711	29	1676	32
Kenjiro	1822	1694	35	1771	3	1729	2
Okiru	1080	961	46	1051	8	1036	12
IRC Bot	1521	1427	49	1478	32	1427	24
Linux Hajime	1162	1106	47	1118	17	1112	18
Muhstik	1284	1217	34	1258	16	1226	15
Hide&Seek	1491	1405	52	1449	26	1430	54
Total	15611	14706/**94,2%**	527/**3,14%**	15144/**97%**	256/**1,5%**	14861/**95,19%**	322/**1,9%**

The experimental results show that usage of the SVM demonstrated better results than the other two methods. The effectiveness of the method involving SVM is in the range of 96,06 to 98,01% with the false positives in the range of 0,015 to 0,31%. Involving of AIS demonstrated effectiveness in the range of 93,8 to 95,9% with the false positives in the range of 0,01 to 0,48%. And the worst results were shown by involving semi-supervised fuzzy c-means clustering - in the range of 89 to 95,2% with false positives in the range of 0,19 to 0,56%.

5 Conclusion

The new technique for IoT cyberattacks detection based on DNS traffic analysis is presented. The method allows detecting IoT botnet cyberattacks. The method has the heuristic and proactive nature. It is based on the gathering of the set of features that may indicate the IoT cyberattacks presence.

The mechanism of attack detection system is based on the cyberattacks' features gathering from network and feature vectors construction.

As the classification algorithms a semi-supervised fuzzy c-means clustering, SVM and Artificial Immune System classification algorithms were employed. The proposed method has demonstrated the ability to detect unknown IoT cyberattacks with high efficiency in the range of 96,06 to 98,01% with the false positives in the range of 0,015 to 0,31%.

The further work may be devoted to the development of the techniques that involve machine learning algorithms and new IoT attacks' features analysis.

6 References

1. Vignau, B., Khoury, R., & Hallé, S. 10 Years of IoT Malware: a Feature-Based Taxonomy. In 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 458-465, IEEE (2019).
2. Murphy, M. The Internet of Things and the threat it poses to DNS. Network Security, 2017.7, pp.17-19 (2017).
3. Angrishi, K. Turning internet of things (iot) into internet of vulnerabilities (iov): Iot botnets. arXiv preprint arXiv:1702.03681 (2017).
4. Alieyan, K., Almomani, A., Abdullah, R., Almutairi, B., & Alauthman, M. Botnet and Internet of Things (IoTs): A Definition, Taxonomy, Challenges, and Future Directions. In Security, Privacy, and Forensics Issues in Big Data, IGI Global, pp. 304-316 (2020).
5. Li, W., Jin, J., & Lee, J. H. Analysis of Botnet Domain Names for IoT Cybersecurity. IEEE Access, 7, 94658-94665 (2019).
6. McDermott, C. D., Majdani, F., Petrovski, A. V. Botnet detection in the internet of things using deep learning approaches. In 2018 international joint conference on neural networks (IJCNN), IEEE, pp. 1-8 (2018).
7. De Donno, M., Dragoni, N., Giaretta, A., Spognardi, A. DDoS-capable IoT malwares: Comparative analysis and Mirai investigation. Security and Communication Networks (2018).
8. Doshi, R., Apthorpe, N., Feamster, N. Machine learning ddos detection for consumer internet of things devices. In 2018 IEEE Security and Privacy Workshops (SPW), IEEE, pp. 29-35 (2018).
9. Moustafa, N., Turnbull, B., Choo, K. K. R. An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things. IEEE Internet of Things Journal, 6(3), 4815-4830 (2018).
10. Al-Duwairi, B., Al-Kahla, W., AlRefai, M. A., Abdelqader, Y., Rawash, A., & Fahmawi, R. SIEM-based detection and mitigation of IoT-botnet DDoS attacks. International Journal of Electrical & Computer Engineering (2088-8708), 10 (2020).

11. Al Shorman, A., Faris, H., Aljarah, I. Unsupervised intelligent system based on one class support vector machine and Grey Wolf optimization for IoT botnet detection. Journal of Ambient Intelligence and Humanized Computing, pp. 1-17 (2019).
12. Alrashdi, I., Alqazzaz, A., Aloufi, E., Alharthi, R., Zohdy, M., Ming, H. AD-IoT: anomaly detection of IoT cyberattacks in smart city using machine learning. In 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, pp. 0305-0310 (2019).
13. Cvitić, I., Peraković, D., Periša, M., Botica, M. Novel approach for detection of IoT generated DDoS traffic. Wireless Networks, pp. 1-14 (2019).
14. Mockapetris P. RFC-1035. Domain names – implementation and specification. ISI, 1987. Available online: http://www.ietf.org/rfc/rfc1035.txt?number=1035 (accessed on March 20, 2020).
15. Pomorova, O., Savenko, O., Lysenko, S., Kryshchuk, A., Bobrovnikova, K. A technique for the botnet detection based on DNS-traffic analysis. Communications in Computer and Information Science, Vol. 522, pp. 127-138 (2015).
16. Lysenko, S., Bobrovnikova, K., Savenko, O. A botnet detection approach based on the clonal selection algorithm. The 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies: Proceedings. Vol. 1, pp. 424-428 (2018).
17. Lysenko, S., Savenko, O., Kryshchuk, A., Klyots, Y. Botnet detection technique for corporate area network. The IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Proceedings, Berlin. Vol. 1, pp. 315-320 (2013).
18. Lysenko S., Bobrovnikova K., Savenko O., Kryshchuk A. BotGRABBER: SVM-based self-adaptive system for the network resilience against the botnets' cyberattacks. Communications in computer and information science, pp. 127-143 (2019).
19. Lysenko S., Savenko O., Bobrovnikova K., Kryshchuk A. 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).
20. The BoT-IoT Dataset. Available online: https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/bot_iot.php (accessed on March 20, 2020).
21. Koroniotis N., Moustafa N., Sitnikova E., Turnbull B. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset", https://arxiv.org/abs/1811.00701 (2018).
22. The UNSW-NB15 Dataset. Available online: https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/ (accessed on March 20, 2020).
23. Stratosphere Laboratory. Aposemat IoT-23. A labeled dataset with malicious and benign IoT network traffic. Parmisano, A., Garcia, S., Erquiaga, M. J. Available online: https://www.stratosphereips.org/datasets-iot23 (accessed on March 20, 2020).
24. Securelist. New trends in the world of IoT threats. Available online: https://securelist.com/new-trends-in-the-world-of-iot-threats/87991/ (accessed on March 20, 2020).
25. Cloudflare. What is the Mirai Botnet? Available online: https://www.cloudflare.com/learning/ddos/glossary/mirai-botnet/ (access.March 20, 2020).
26. Kharchenko V., Kondratenko Y., Kacprzyk J. (eds). Concepts of Green IT Engineering: Taxonomy, Principles and Implementation. Green IT Engineering: Concepts, Models, Complex Systems Architectures. Studies in Systems, Decision and Control, Springer, Cham, 2017, Vol. 74. pp. 3-19
27. Singh, K., Singh Dhindsa, K., & Bhushan, B. (2018). Performance analysis of agent based distributed defense mechanisms against ddos attacks. International Journal of Computing, 17(1), 15-24. Retrieved from http://computingonline.net/computing/article/view/945.