Iraqi Journal of Science

ISSN: 0067-2904

A Decision Tree-Aware Genetic Algorithm for Botnet Detection

Thurayaa B. Alhijaj*, Sarab M. Hameed, Bara'a A. Attea
Department of Computer Since, Collage of Since, University of Baghdad, Baghdad, Iraq

Received: 25/7/2020 Accepted: 14/10/2020

Abstract

In this paper, the botnet detection problem is defined as a feature selection problem and the genetic algorithm (GA) is used to search for the best significant combination of features from the entire search space of set of features. Furthermore, the Decision Tree (DT) classifier is used as an objective function to direct the ability of the proposed GA to locate the combination of features that can correctly classify the activities into normal traffics and botnet attacks. Two datasets namely the UNSW-NB15 and the Canadian Institute for Cybersecurity Intrusion Detection System 2017 (CICIDS2017), are used as evaluation datasets. The results reveal that the proposed DT-aware GA can effectively find the relevant features from the whole features set. Thus, it obtains efficient botnet detection results in terms of F-score, precision, detection rate, and number of relevant features, when compared with DT alone.

Keywords: Botnet; decision tree; feature selection; genetic algorithm.

شجرة القرار مع خوارزمية جينية لاكتشاف شبكة الروبوت

ثريا بريسم*، سراب مجيد حميد , براء علي عطية
قسم علوم الحاسوب ، كلية العلوم ، جامعة بغداد ، بغداد، العراق.

الخلاصة

في هذا البحث ، يتم تعريف مشكلة اكتشاف البوت نت كمشكلة اختيار الميزة ويتم استخدام الخوارزمية الجينية (GA) للبحث عن أفضل تركيبة مهمة من الميزات من مساحة البحث الكاملة لمجموعة الميزات.علاوة على ذلك ، يتم استخدام مصنف شجرة القرار (DT) كوظيفة موضوعية لتوجيه قدرة الخوارزمية الجينية GA المقترحة على تحديد مجموعة الميزات التي يمكن أن تصنف الأنشطة بشكل صحيح في عمليات النقل العادية وهجمات الروبوتات. يتم استخدام مجموعتي بيانات UNSW-NB15 ومجموعة بيانات CICIDS2017كمجموعات بيانات تقييم. تكشف النتائج أن الخوارزمية الجينية مع شجرة القرار يمكن ان يجدان بشكل فعال الميزات ذات الصلة من مجموعة الميزات الكاملة ، وبالتالي الحصول على نتيجة فعالة للكشف عن الروبوتات عن طريق تطبيق مجموعة من مقاييس التقييم F-score ، والدقة ومعدل الكشف وكذلك الميزات ذات الصلة عند المقارنة مع شجرة القرار لوحدها.

1. Introduction

Botnets, as a set of the words "ro-*bot*" and "*net*-work", is a network of computers infected with malicious software and remotely commanded and controlled by cybercriminals, or the so-called

*Email: prog.th88@gmail.com

botmasters. Any host on the Internet that is flawed by a botmaster becomes a zombie in this botnet and joins tens or even hundreds of thousands of zombies in the botnet's army.

The main Command and Control (C&C) servers of the intruders in this army will, in turn, gain huge amount of money while accessing large numbers of infected machines. Due to the vast usage of network-based services and applications in our daily digital world, such cybercriminal activities significantly affect the economic cost. In April 2012, e.g., the Symantec cybercrime report reported that cyber-attacks cost US$114 billion each year. However, if the time lost by companies trying to recover from cyber-attacks is also counted, the total cost of cyber-attacks would reach US$385 billion [1]

A Botnet can emerge from several families; namely Internet Relay Chat (IRC), Hypertext Transfer Protocol (HTTP), and Peer-to-Peer (P2P). Based on the C&C infrastructure, a botnets network can be classified into centralized and decentralized. In centralized botnets, the botmaster usually uses the C&C server to send a command to the bots. The centralized botnet is popularly adopted by various botnet families because of its simplicity. However, its main limitation is its single point of the failure C&C server. To evade this defect, decentralized C&C infrastructures, such as the P2P, have been developed by botnet attackers [2].

Although various machine learning and data mining algorithms have been proposed in the literature to build several botnet detection models, almost all these models and algorithms are based on feature extraction (or feature construction), where different feature sets are extracted from the available high dimensional dataset based on some expertise and skills [3]. On the other hand, feature selection, which engages an essential role in developing different machine learning models, is found to be of little attention in the literature of botnet detection.

The contribution of this paper is to present the botnet detection problem as an optimization problem and to adopt a Genetic Algorithm (GA) to tackle it. The problem is formulated as feature selection, where the GA has to search, from a set of different scenarios, for the best possible combination of features that can correctly discriminate botnet traffic data from normal ones. The proposed GA is designed to be within DT-aware search algorithm, where the DT classifier is used to rank the individual solutions according to the F-score rate of botnets.

The remainder of this paper is organized as follows: Section 2 presents some botnet detection related works. Section 3 presents the preliminary concepts. The proposed DT-aware GA for botnet detection is introduced in Section 4. Experimental results and the related discussions are reported in Section 5. Finally, conclusions and possible future ideas are commented in Section 6.

2. Related work

Recently, many works have been presented in the literature to describe different botnet detection techniques for botnet detection problems. An anomaly detection model based on the genetic neural feed-forward network is proposed in [4]. Searching for the best setting for the initial weights of back-propagation feed-forward neural networks is the main concept of this model. The work in [5] introduced a botnets detection method to distinguish between two behaviors: bots having non-periodic and normal traffics which normally reveal periodic behavior. Principal components analysis (PCA) is used for feature selection and J48 algorithm is used for classification. Feature selection from PCA helps in increasing the detection rate of the bot traffic.

In [6] the UNSW-NB15 dataset is analyzed statistically and practically. Five different classifiers: Naïve Bayes (NB), Decision Tree (DT), Artificial Neural Network (ANN), Logistic Regression (LR), and Expectation-Maximization (EM) Clustering are used to assess the complexity in terms of accuracy and False Alarm Rrate (FAR).

To investigate the impact of feature selection on the performance of a botnet detection technique, the study in [2] is suggested. Three different feature selection methods have been developed in this study, namely linear models penalized with the L1 norm, Recursive Feature Elimination (RFE), and Tree-based feature selection (i.e. random forest feature ranking). Four main steps constituted the detection methodology. Feature extraction using Tranalyzer is applied to the raw network captures. The statistical feature set being extracted from the first step is then delivered to the selection process to select the most deputy features subset influencing accurate classification of traffic category into botnet and normal ones. The adopted classifiers are used in the third step, while the evaluation of the classification methods is computed in the final step.

In [7], a single objective GA is used to explore the high space of candidate features and attempt to

locate the best subset that could improve the performance of the GA-based botnet detection system.
In [8], a botnet flow classification system based on C4.5 algorithm is described. A set of features is deterministically selected from the input Packet Capture (PCAP) file using the greedy search mechanism of Consistency Subset Evaluation. To extract the behavior pattern of each flow traffic, the classifier algorithm classifies similar flow traffic into groups. The proposed system not only can analyze P2P botnets, but further it extracts the patterns to an application layer and can analyze botnets using HTTP protocols.
A P2P botnet detection method based on four-layer network traffic classification is proposed in [9]. Feature reduction is gained using a decision tree algorithm. While the first layer filters out non P2P packets, the second layer characterizes the captured network traffic into non P2P and P2P. At the third layer, the selected features are further reduced and at the final layer, P2P botnets are detected using decision tree classifier. Little attention has been paid to the role of feature selection in perceiving only the feature set that could get a lower detection rate than other features.

3 .Preliminary concepts

3.1 Genetic algorithm

Genetic algorithm (GA) is a class of adaptive search algorithms in which the main idea is based on natural genetics. GA starts by initializing a population of several individuals. Then, the individuals are evaluated according to the objective function for the given problem. After that, the selection, crossover, and mutation operators are applied to the entire population for generating new, and hopefully better, individuals. Different canonical selection, crossover, and mutation operators are provided in the literature. Examples are binary tournament selection, roulette wheel selection, *n*-point crossover, inversion mutation, and substitution mutation operators. The process of generating new individuals by selection, crossover, and mutation operators continues until a termination condition is justified [10, 11] .

3.2 Decision tree

Decision tree (DT) is a popular method in machine learning. The decision tree models are determined to be the most valuable in the domain of data mining, considering reasonable accuracy with relatively inexpensive computation. The decision tree is defined as a supervised learning that classifies the labeled trained data into a tree or rules. It divides a data domain (node) recursively into two subdomains such that the subdomains have a higher information gain than the node that it was split from. Finding the best split that gives the maximum information gain is the goal of the optimization algorithm in the decision tree-based supervised learning [12, 13] .

4. Proposed decision tree-aware genetic algorithm for botnet detection

In an abstract language, the proposed GA is designed to be as a feature selector algorithm. It should select the most suitable combination of features from a mixed dataset of both normal and botnet traffics. Further, the DT is used as a supervised learning algorithm to direct the GA for the promising set of features in the search space. Feature selection, however, is proved to be a non-deterministic polynomial time hard (NP-hard) problem. The complexity of the problem increases exponentially with the problem dimension.
In this section, the characteristic components of the proposed GA are presented, while tackling the botnet detection problem. The problem is formulated as a feature selection problem. Furthermore, DT is used to judge the ability of the proposed GA to locate the combination of features that can correctly classify the activities into normal and botnet activities. Figure- 1 depicts the general layout of the proposed botnet detection.

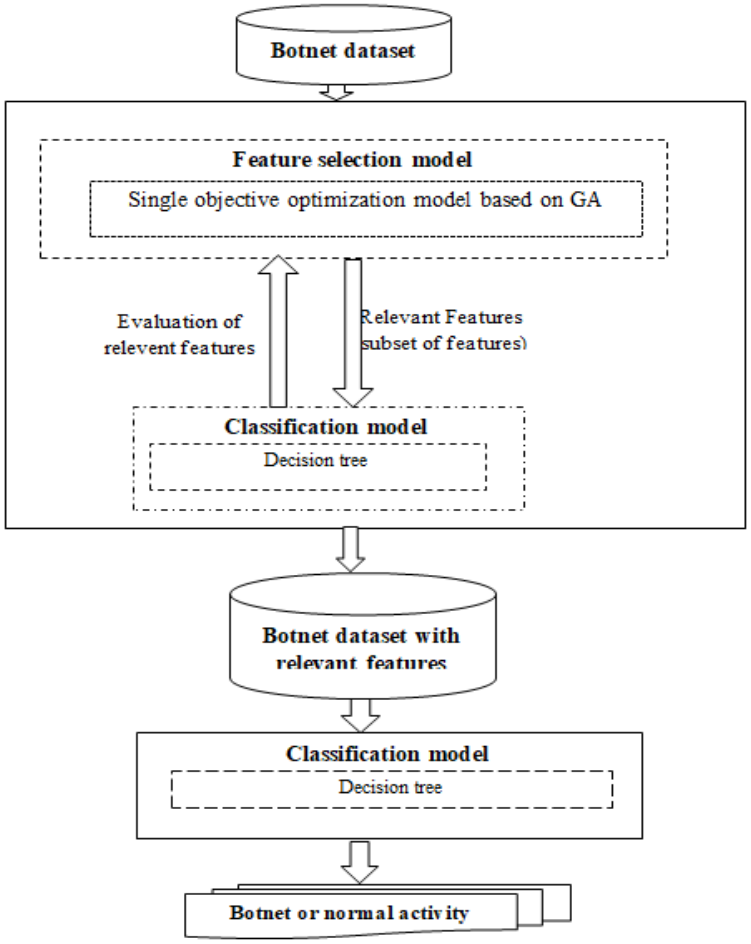

Figure 1- General layout of the proposed botnet detection

4.1 Individual's genotype encoding and population initialization

Each individual (solution) in the proposed GA is represented as a fixed length vector of n genes, where n is the total number of features. The locus of each gene maps to the corresponding feature while its allele value can be either 0 or 1 to indicate the absence or presence of this feature. The total search space size for the modeled botnet detection problem can, thus, be calculated as the Cartesian product of presence and absence of all features (i.e., 2^n).

GA is a population based algorithm, i.e. it is composed of a population $\mathbb{P}$ of N individuals. The mathematical formulation of $\mathbb{P}$ can be expressed as $\mathbb{P} = \{I_1, I_2, \dots I_N\}$. The formulation of the individual I can then be expressed as $I_{1 \le i \le N} = \{I_{i,1}, I_{i,2}, \dots, I_{i,n}\}$. The genes of each individual $I_{i,1 \le j \le n}$ are first generated randomly to ensure unbiased search in the beginning. As expressed in Equation 1, r refers to a uniform random number in the range [0,1] generated for each gene in each individual.

$\forall 1 \le i \le N$

$$I_{i,1 \le j \le n} = \begin{cases} 1 \; if \; r \le 0.5 \\ 0 \; otherwise \end{cases} \quad (1)$$

4.2 Objective function

In GA, the objective function $\Phi: \mathrm{I} \rightarrow \mathbb{R}$ is used to determine the quality of the individuals during the evolution process. The objective function regarding the feature selection problem within botnet detection ($\Phi: \mathrm{I} \rightarrow \mathbb{R}^+$) is designed to maximize the F-score. F-score is expressed in Eq.2, reflecting the harmonious mean within DR and P. Each individual I_i in $\mathbb{P}$ is evaluated according to Equation 2. The details of evaluating the objective function are sketched in Algorithm 1.

$\forall i = 1, \dots, N$

$$\Phi(I_i) = \frac{2*\text{Precision} \times DetectionRate}{\text{Precision} + Detection\ Rate} \quad (2)$$

Precision (P) as expressed in Eq. 4 measures the truly positive exemplars predicted from the whole number of examples in the positive class.

Detection rate (DR) as expressed in Eq. 3 measures the ratio of positive exemplars that are accurately categorized,

where

$$Precision\ (P) = \frac{TP}{TP+FP} \quad (3)$$

$$Detection\ Rate\ (DR) = \frac{TP}{TP+FN} \quad (4)$$

$True\ positive\ (TP)$ represents botnet that is correctly classified,

$True\ negative\ (TN)$ represents normal that is correctly classified,

$False\ positive\ (FP)$ represents normal that is misclassified as an botnet , and

$False\ negative\ (FN)$ represents botnet that is misclassified as normal.

Algorithm 1: Objective Function Evaluation

Input:

- I: Individual
- $\mathbb{T}$: training dataset
- $\mathbb{V}$: validation set
- L: label set of $\mathbb{V}$
- D: No. of traffic data in $\mathbb{V}$
- n: individual length

Output: $\Phi(I)$**:** Objective Function value for *I*

1: **Parameters Setting**

$FilteredTrainingSet$=[];

$FilteredValidationSet$=[];

OnFeaturesCounter = 0;

$TP = 0$;

$FP = 0$;

$FN = 0$;

2: $For\ i = 1\ to\ n$

$\boldsymbol{If}\ \ I(i) = 1$

OnFeaturesCounter = OnFeaturesCounter+1;

// **Generate Filtered Training set and Filtered Validation set**

$FilteredTrainingSet$ = $FilteredTrainingSet \cup \mathbb{T}$(OnFeaturesCounter)

$FilteredValidationSet$= $FilteredValidationSet \cup \mathbb{V}$(OnFeaturesCounter)

$\boldsymbol{Endif}$

$\boldsymbol{Endfor}$

3: Apply decision tree classifier for $FilteredTrainingSet$ and assign label for each data in $FilteredValidationSet$ stored in $ValidationLabel$

4: **Compute** TP **,** $FP\ and\ FN$

$\boldsymbol{For}\ j = 1\ to\ D$

$\boldsymbol{if}\ ValidationLabel_{\,j} = 1\ \ \boldsymbol{and}\ \boldsymbol{L_j} = 1$

$TN = TN + 1$

$\boldsymbol{else}$

$\boldsymbol{if}\ ValidationLabel_{\,j} = 0\ \boldsymbol{and}\ \boldsymbol{L_j} = 1$

$FP = FP + 1$

$\boldsymbol{else}$

$\boldsymbol{if}\ ValidationLabel_{\,j} = 1\ \boldsymbol{and}\ \boldsymbol{L_j} = 0$

$FN = FN + 1$

Endif
Endfor
5: Evaluate individual *I* using Equation 2

4.3 GA evolution operators

The proposed GA can be described as an iterative transformation function $\Psi: \mathbb{P} \rightarrow \mathbb{P}'$ with $\Psi(\mathbb{P}_t) = \mathbb{P}_{t+1}$, where $\mathbb{P}_t$ and $\mathbb{P}_{t+1}$ are the population at iteration t and $t+1$, respectively. The population begins within an initial set of individuals and continues till a maximum number of iterations max_{gen} have been reached. The evolution (transformation) function Ψ is composed of three operators: selection, crossover, and mutation. Binary tournament selection is adopted in this paper, in which two random individuals are selected and the one with the largest fitness function (largest F-score) is passed to the mating pool for the next generation.

Uniform crossover operator with probability $p_c \in [0,1]$ and mutation operator are used. In uniform crossover, two parents, I^{p1} and I^{p2}, are crossed over to produce a child, $'$, as formulated in Equation (5).

$\forall i, 1 \leq i \leq n$

$$I_i' = \begin{cases} I_i^{p1} \; if \; r \leq 0.5 \\ I_i^{p2} \; otherwise \end{cases} \quad (5)$$

where

r is a uniform random number in [0,1].

For the mutation operator, a random number $p_m \in [0,1]$ is generated. The allele of the mutated gene can be flipped from 1 to 0 or vice versa with probability p_m.

4.4 Botnet detection based on DT aware GA

Figure- 1 depicts the general layout incorporating the components of GA mentioned previously. The botnet detection using GA-based feature selection can be recapitulated in Algorithm 2.

Algorithm 2: Botnet detection-aware GA

Input :

- $\mathbb{T}$: training dataset.
- $\mathbb{T}'$: testing dataset
- N: Population size.
- p_c: Crossover probability.
- p_m: Mutation probability.
- max_{gen}: maximum number of generations

Output:

- labeled data $\{normal, attack\}$ for $\mathbb{T}'$

1. Generate initial population P^0
2. Calculate the objective function (F-Score) for each individual *I* in P^0 using algorithm 1
3. ***for i = 1 to max_{gen}***
4. Apply elitism by copying the best GA individual to the next generation.
Apply tournament selection
Apply uniform crossover with p_c
Apply mutation operator with p_m
Calculate objective function for each individual *I* using algorithm 1.
***End*for**
5. Take out the best individual *I* that has the highest F-score
6. Apply the decision tree classifier for $\mathbb{T}$ and assign label for each data $\mathbb{T}'$in

5. Performance Evaluation

UNSW-NB15 dataset [6] is used as an evaluation dataset for the assessment of the proposed GA-based botnet detection. The dataset is designed by XIA PerfectStorm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) to generate real modern normal and nine types of attack activities. The types are namely Analysis, Backdoor, Denial of service (DoS), Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms. The dataset is divided into two groups: the first group is the training set containing 175,341 activities and the second is the testing group containing 82,332 activities. The percentage of normal in the training and testing sets is 44.94% and 31.94%, respectively. While the percentage of the attack in the training and testing sets is 55.06% and 68.06%, respectively. Each instance in the dataset has 42 features in addition to class labels (0 for normal and 1 for attack). Figure- 2 depicts the distribution of the normal and the attack categories in the training and testing sets in UNSW-NB15 dataset.

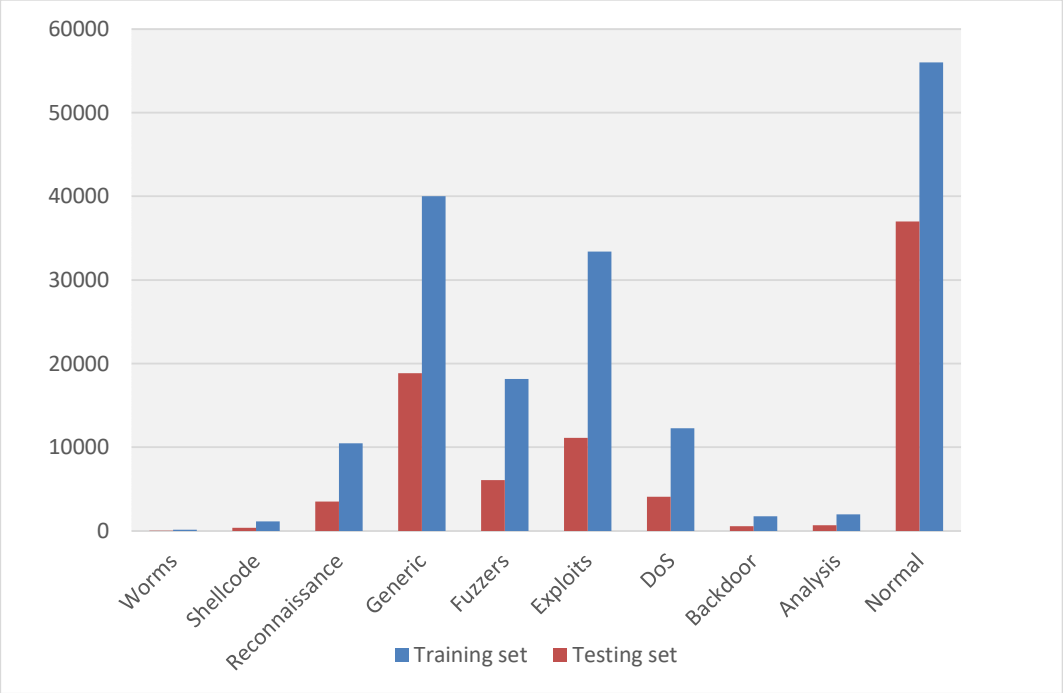

Figure 2- The distribution of the normal and the attack categories in the training and testing sets in UNSW-NB15 dataset.

Furthermore, the CICIDS2017 dataset [14] is used for evaluating the proposed GA botnet detection. The training dataset contains three types of the botnet and the testing dataset includes three types of botnets. Each instance in the dataset has 78 features in addition to class labels (0s for normal traffics and 1s for attacks). The percentage of malicious in the training dataset is 38.33% while the percentage of malicious flow in the testing dataset is 29%.

Several experiments have been conducted to assess the performance of the proposed GA botnet detection regarding five percentages of validation set (5%, 10%, 15%, 20% and 25%). All the experiments have been conducted with the same setting of GA parameters. The parameters are set as follows: a population of $N = 100$ individuals was used and evolved over a sequence of $max_{gen} = 100$, $p_c = 0.9$, and $p_m = 0.1$ have been chosen.

Table-1 reports the $F - score$, P, and DR values of the proposed model, which are evaluated using UNSW-NB15 dataset and represented by an average of 30 runs. Table-2 reports the average $F - score, P,$ and DR values of the proposed model, that are evaluated using CICIDS2017 dataset and represented by an average of 5 runs with the same parameters.

The results in the tables reveal that the proposed GA can effectively find the relevant features from the whole features set, and thus obtaining an efficient botnet detection result. Furthermore, the results highlight that increasing the size of the validation set has a positive impact on achieving high F-score with less number and relevant features.

Table 1- Comparison of the results of DT against DT-aware GA within UNSW-NB15 dataset

Model	Avg. FS	Avg. DR	Avg. P	Avg. # Features
DT-42	0.9214	0.8883	**0.9572**	42
DT-GA-%5	0.9374	0.9921	0.8883	14.5667
DT-GA-%10	0.9368	0.9935	0.8863	**14.444**
DT-GA-%15	0.9363	0.9928	0.8859	14.5000
DT-GA-%20	0.9373	0.9933	0.8873	15.6333
DT-GA-%25	**0.9383**	**0.9942**	0.8884	16.966

Table 2- Comparison of the results of DT against DT aware GA within CICIDS2017 dataset

Model	Avg. FS	Avg. DR	Avg. P	Avg. # Features
DT-78	0.9257	0.9645	0.8900	78
DT-GA-%5	0.9715	0.9516	0.9921	36
DT-GA-%10	0.9732	0.9516	0.9959	34
DT-GA-%15	0.9751	0.9515	**0.999**	37
DT-GA-%20	0.9716	0.9516	0.9923	40
DT-GA-%25	**0.9800**	**0.9685**	0.9917	**33**

6. Conclusions

In this paper, a botnet detection model is proposed. The proposed model uses a DT-aware GA to select the best possible subset of features to discriminate botnet traffics from normal traffics. The results reveal the ability of the proposed model to improve the F-score and the detection rate of botnet detection, with less number of features when compared against a DT based botnet detection model. In the future, a multi-objective evolutionary algorithm, such as the decomposition based multi-objective evolutionary algorithm (MOEA/D), can be used to maximize both precision and detection rate, instead of applying a single objective GA.

References

1. Internet Security Threats Report. Symantec, http://www.symantec.com/threatreport/, last accessed: June **2013**.
2. Alauthaman, M., Aslam, N., Zhang, L., Alasem, R., & Hossain, M. A. **2016**. A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks. *Neural Computing and Applications*, **29**(11): 991–1004. doi: 10.1007/s00521-016-2564-A.
3. Pektaş, T. Acarman. **2017**. Effective Feature Selection For Botnet Detection Based On Network Flow Analysis, International Conference Automatics and Informatics'2017.
4. Yin, C., Awlla, A. H., Yin, Z., & Wang, J. **2015.** Botnet detection based on genetic neural network. *International Journal of Security and Its Applications*, **9**(11): 97-104.
5. Harsha, T., Asha, S., and Soniya, B. 2016. Feature selection for effective botnet detection based on periodicity of traffic. *Information Systems Security Lecture Notes in Computer Science*, pp. 471–478.
6. Moustafa, N., & Slay, J. **2016**. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective*, **25**(1-3): 18–31. doi: 10.1080/19393555. 2015.1125974.
7. Alejandre, F. V., Cortés, N. C., & Anaya, E. A. **2017**. Feature selection to detect botnets using machine learning algorithms. In*ternational Conference on Electronics, Communications and Computers*. https://doi.org/10.1109/CONIELECOMP.2017.7891834.
8. Hung, C., & Sun, H. **2018**. A Botnet Detection System Based on Machine-Learning using Flow-Based Features. *The twelfth international conference on emerging security information, systems*

and technologies,pp. 122–127.

9. Khan, R. U., Zhang, X., Kumar, R., Sharif, A., Golilarz, N. A., & Alazab, M. **2019**. An Adaptive Multi-Layer Botnet Detection Technique Using Machine Learning Classifiers. *Applied Sciences*, **9**(11): 2375. doi: 10.3390/app9112375

10. Aissa, N. B., & Guerroumi, M. **2015.** A genetic clustering technique for Anomaly-based Intrusion Detection Systems. 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing. https://doi.org/10.1109 /SNPD.2015.7176182.

11. Dhuha I. M. and Sarab M. H. **2016**. A Feature Selection Model based on Genetic Algorithm for Intrusion Detection, *Iraqi Journal of Science, Special Issue*, Part A, pp. 168-175

12. Suthaharan, S. **2016**. Machine Learning Models and Algorithms for Big Data Classification. *In Integrated Series in Information System*s, **36**. https://doi.org/10.1007/978-1-4899-7641-3

13. Lavanya, D. **2012.** Ensemble Decision Tree Classifier For Breast Cancer Data. *International Journal of Information Technology Convergence and Services*. **2**(1): 17–24. https://doi.org/ 10.5121/ijitcs.2012.2103.

14. Canadian Institute for Cybersecurity, https://www.unb.ca/cic/datasets/ids-2017.html