Detecting Unauthorized Network Intrusion based on Network Traffic using Behavior Analysis Techniques

Nguyen Tung Lam
Information Assurance Department
FPT University, Hanoi
Vietnam

***Abstract*—Nowadays, network intrusion detection is an essential problem because cyber-attacks are increasing in both the number and extent of the danger. Network intrusion techniques often use various methods to bypass the oversight of anomaly detection and surveillance systems. This paper proposes to use behavior analysis techniques, machine learning, and deep learning algorithms for the task of detecting network intrusions. The practical and scientific significance of our paper includes two issues: (1) Regarding the process of selecting and extracting features: instead of using typical abnormal behaviors of attacks, this study will use statistical behaviors that are easy to calculate and extract while still ensuring the effectiveness of the method; (2) Regarding the detection process, this study proposes to use the Random Forest (RF) classification algorithm, the Multilayer Perceptron (MLP) and the Convolutional Neural Network (CNN) deep learning model. The experimental results in Section IV have proven that our proposal in this paper is completely correct and reasonable. Based on the results shown in Section IV, this study has provided network surveillance systems with a number of abnormal behaviors as the basis for detecting network intrusions.**

Keywords—Network intrusion detection; abnormal behaviors; IDS 2018 dataset; deep learning and machine learning

I. Introduction

Unauthorized intrusion techniques are a dangerous attack form, have been growing rapidly in both the number of recorded attacks and the extent of damage that it causes to organizations or enterprises. Therefore, the task of early detecting and warning signs of cyber-attack campaigns is essential nowadays. Currently, there are two main methods to detect network intrusions: signature-based method through rulesets and anomaly-based method based on analyzing data and statistics to seek abnormal characteristics in the network [1], [2], [3]. The signature-based method has the ability to detect network intrusions quickly and accurately, but it is not possible to detect new attack techniques [1]. The anomaly-based method not only has the ability to detect attacks but also has the ability to detect abnormal behaviors, but it requires complex computation and processing processes and its accuracy is not high. The anomaly-based method is often based on two main techniques to classify abnormal and normal behavior, machine learning and deep learning [1], [2]. So clearly, regarding the network intrusion detection method using machine learning or deep learning, the most important factor is how to identify normal behavior and abnormal behavior. The studies [4, 5] focused on extracting abnormal characteristics and behaviors based on specific attack techniques. However, we noticed that such an approach could quickly and accurately detect attacks based on specific datasets, but when using other datasets, it is difficult to detect cyber-attack techniques. Therefore, this paper proposes a new network intrusion detection method using deep learning and machine learning algorithms including RF, MLP, CNN based on analyzing behaviors of network traffic. Accordingly, in this paper, we will not find ways to analyze abnormal behavior in network data, we only try to statistic the behavior of network traffic and then use machine learning and deep learning algorithms for analysis and evaluation. With this approach, this study will reduce many steps in finding and extracting abnormal behavior of network intrusion techniques. For the experimental dataset, PCAP files in the IDS 2018 dataset [6] will be selected and used. The study [7] listed and analyzed a number of datasets typically used for detecting cyber-attacks such as DARPA/KDD Cup99, CAIDA, NSL-KDD, ISCX 2012, UNSW-NB15, IDS 2018, etc. In which, the IDS 2018 dataset is built and developed in accordance with real network systems. Therefore, this study will use the IDS 2018 dataset to conduct experiments of cyber-attack detection methods.

II. Related Works

In the study [8], Vikash Kumar et al. proposed a cyber-attack classification method using UNSW-NB15 and rulesets. Nour Moustafa et al. [9] proposed Geometric Area Analysis Technique for cyber-attack detection using Trapezoidal Area Estimation. This study used UNSW-NB15 and NSL-KDD datasets to conduct experiments in order to evaluate the effectiveness of the proposed method. The experimental results in this study showed the superiority of the UNSW-NB15 dataset compared to the NSL-KDD dataset.

In addition, the study [10] presented a scalable framework for building an effective and lightweight anomaly detection system based on two well-known datasets, the NSL-KDD and UNSW-NB15.

Sikha Bagui et al. proposed in their study [11] a method to detect cyber-attacks based on the Naïve Bayes and Decision Tree (J48) machine learning algorithms. The team [11] used these two algorithms in turn for classifying components of cyber-attacks in the UNSW-NB15 dataset.

The study [12] proposed a cyber-attack detection model using the stacking technique. In their model, the training process uses some machine learning algorithms including K-nearest Neighbor (KNN), Decision Tree (DT) and Logistic

Regression (LR) to build the model based on the UNSW-NB15 and UGR'16 datasets.

The study [13] performed an evaluation of the efficiency of 8 machine learning algorithms (2-layers and 3-layers) for network intrusion detection.

The study [14] presented a DDOS attack detection method using a comprehensive simulation technique of DDOS attacks.

In the study [15], Cho et al. proposed two tasks: detecting cyber-attacks using machine learning algorithms and optimizing features using algorithms such as IG, PCA. Experimental results showed that the team's proposals were relatively good. However, because feature optimization algorithms have large computational times and high complexity, a large calculation system is required. In addition, Cho et al. [16, 17, 24, 25] proposed a method to detect cyber-attacks based on network traffic using machine learning and deep learning algorithms.

In the study [18], Zhao et al. proposed a botnet detection method based on analyzing abnormal behaviors of traffic and flow. Besides, the approach to detect botnet and cyber-attack using the CTU 13 dataset was proposed by Chowdhury et al. [19]. In addition, Ahmed [20] proposed using the ANN deep learning algorithm to classify abnormal connections.

III. Network Intrusion Detection Method using Behavior Analysis Techniques

The facts show that with the approach of detecting unauthorized network intrusion using behavior analysis techniques, systems need to perform two main tasks: i) defining abnormal behavior. This definition process is the task of selecting and extracting features, ii) a method of classifying behaviors. This process uses machine learning or deep learning algorithms to classify the behaviors that have just been built in the task (i). We will delve into analyzing and clarifying this content in the next section of the paper.

A. Selecting and Extracting Features

This paper uses the CICFlowMenter tool [21] to handle network traffic. This tool has a function analyzing network traffic into 76 features [16, 17]. These features were presented in detail in the studies [17, 24].

B. Detection Method

As mentioned above, in order to classify intrusion behavior in network traffic, this paper uses a combination of machine learning and deep learning algorithms including Random Forest, CNN, and MLP. These algorithms are being studied and applied in many different problems of the recognition field.

In this, the Random Forest algorithm is a supervised machine learning algorithm researched and developed by [22]. The studies [1, 16] have shown that this algorithm is currently the best classification algorithm because it has a simple operation principle, is easy to calculate and install, especially has low calculation and classification time. The study [22] presented the operating principle and the mathematical model of this algorithm in detail. This paper will use the Random Forest algorithm with standard parameters. We only change the number of random trees in the algorithm to find and conclude the best model of the algorithm with this experimental dataset.

Regarding the MLP network, the study [23] presented in detail the architecture of an MLP network that is built by simulating the way neurons work in the human brain. MLP networks usually have 3 or more layers including 1 input layer, 1 output layer, and more than 1 hidden layer. Besides, the efficiency of the MLP network depends on the activation function. In this paper, we will tune activation functions to evaluate the effectiveness and suitability of activation functions for the network intrusion detection task.

Finally, the CNN network is defined as a set of basic layers including convolution layer + nonlinear layer, fully connected layer. The detailed structure of CNN as well as the terms: stride, padding, MaxPooling are presented in detail in the paper [23]. In which, the ReLU activation function is used.

IV. Experiments And Evaluation

A. Experimental Dataset and Scenarios

The experimental dataset is extracted from IDS 2018 Dataset with three types of attacks: Bot (Botnet), Dos, and HTTP-attacks. This dataset is divided into 2 sub-datasets with a total of 762,000 records. In which: the first sub-dataset has two labels: 0 (Benign - clean) and 1 (Bot - malicious); the second sub-dataset has three labels: 0 (Benign - clean), 1 (Dos - malicious), 2 (HTTP-attacks - malicious). We use 70% of this dataset for training and the remaining 30% for testing. Besides, in this paper, to see the effectiveness of the proposed method, we will proceed to refine the parameters of each algorithm to find the most optimal model and architecture.

B. Measures to Evaluate the Results of the Algorithm

The following measures will be used in this paper to evaluate the accuracy of models:

- Accuracy: The ratio between the number of samples classified correctly and total number of samples. Accuracy is calculated by the following formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where: TP - True positive: The number of malicious samples classified correctly; FN - False negative: The number of malicious samples classified as normal; TN - True negative: The number of normal samples classified correctly; FP - False positive: The number of normal samples classified as malicious.

- Recall: is the ratio of true positive points to the total number of real positive points (TP + FN). A high recall means that the TP is high and the rate of missing really positive points is low.

$$Recall = \frac{TP}{TP + FN}$$

- Precision: is the ratio of true positive points to the total number of points classified as positive (TP + FP).

$$Precision = \frac{TP}{TP + FP}$$

- F1-score: is harmonic mean of precision and recall. The higher the F1, the better the classifier.

$$F1 = \frac{2 \times precision \times \mathrm{Re}call}{precision + \mathrm{Re}call}$$

C. Experimental Results

1) Experimental results with random forest

a) 2-classes dataset: Table I lists the experimental results of network intrusion detection applying the Random Forest algorithm with 10, 50, 100 trees using the 2-labels dataset.

TABLE I. EXPERIMENTAL RESULTS OF NETWORK INTRUSION DETECTION USING RANDOM FOREST ALGORITHM WITH THE 2-LABELS DATASET

The number of trees	**Accuracy (%)**	**F1 (%)**	**Precision (%)**	**Recall (%)**
10	99.967654	99.967657	99.967679	99.967654
50	**99.995733**	**99.995733**	**99.995734**	**99.995733**
100	99.988050	99.988050	99.988050	99.988050

From Table I, could see that the algorithm has the highest Accuracy and Precision (99.996%) when the number of decision trees is 50. Besides, when the number of decision trees is changed from 10 to 100, the accuracy of the algorithm does not change much. This shows that with the dataset balanced on the ratio of normal and abnormal records, the Random Forest algorithm brings good and stable detection results. Fig. 1 below presents the evaluation results of the confusion matrix when the number of decision trees is 50. From Fig. 1, seeing that the normal and abnormal prediction models all have high accuracy.

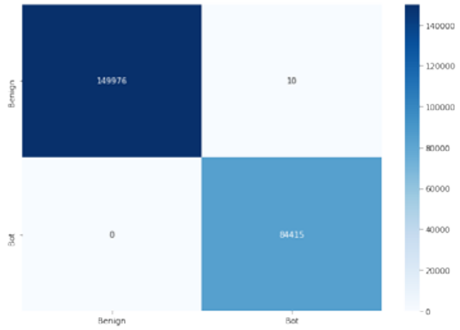

Fig. 1. Confusion Matrix of Random Forest with 50 Trees.

b) 3-classes dataset: Table II lists the experimental results with the 3-labels dataset.

TABLE II. EXPERIMENTAL RESULTS OF NETWORK INTRUSION DETECTION USING RANDOM FOREST ALGORITHM WITH THE 3-LABELS DATASET

The number of trees	**Accuracy (%)**	**F1 (%)**	**Precision (%)**	**Recall (%)**
10	99.864486	99.837309	99.854480	99.864486
50	99.878638	99.866030	99.868550	99.878638
100	**99.886005**	**99.873035**	**99.875924**	**99.886005**

Based on the experimental results in Table II, we found that: similar to the 2-labels, the scores obtained with the 3-labels dataset had high results (all over 99%). The Random Forest algorithm gave the best classification results with the number of trees of 100. Comparing the results in Table I and Table II shows that the Random Forest algorithm gave higher efficiency on all measures when using the 2-labels dataset. Confusion Matrix with 100 trees is shown in Fig. 2.

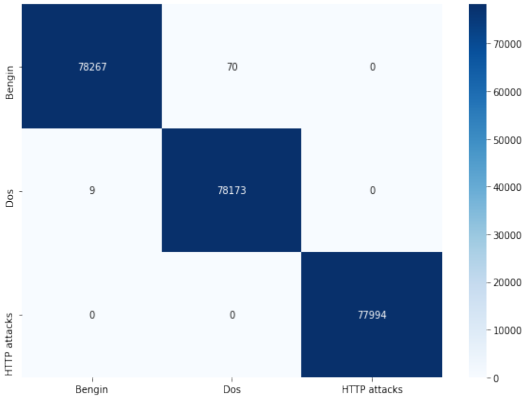

Fig. 2. Confusion Matrix of Random Forest with 100 Trees.

2) Experimental results with MLP

a) 2-classes dataset: From the results shown in Table III, seeing that the MLP model gave very different results when using different activation functions and the number of layers. In particular, with 2 layers, the MLP model gave the best result with ReLU activation. However, when increasing the number of layers to 4, the MLP model had the best results with Logistic activation. But considering accurately detecting the intrusion techniques, the MLP model with ReLU activation still gave a completely better result (reaching 100%). Fig. 3 below is the result of Confusion Matrix when using the ReLU activation function.

From Fig. 3, it can be seen that the MLP model gave prediction results with very high accuracy, with only 32 incorrectly classified records. With this result, it is clear that the MLP model is completely consistent with the purposes and requirements.

TABLE III. EXPERIMENTAL RESULTS OF NETWORK INTRUSION DETECTION USING MLP ALGORITHM WITH THE 2-LABELS DATASET

Parameters		**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
Hidden Units	***Activation function***				
2	identity	94.34	94.51	89.53	91.95
	logistic	97.37	93.65	99.47	96.47
	tanh	81.33	98.33	49.17	65.56
	ReLU	**99.90**	**99.76**	**99.96**	**99.86**
4	identity	99.06	98.12	99.29	98.71
	logistic	**99.62**	**99.64**	**99.64**	**99.48**
	tanh	98.18	95.69	99.45	97.53
	ReLU	63.86	63.86	100	77.94

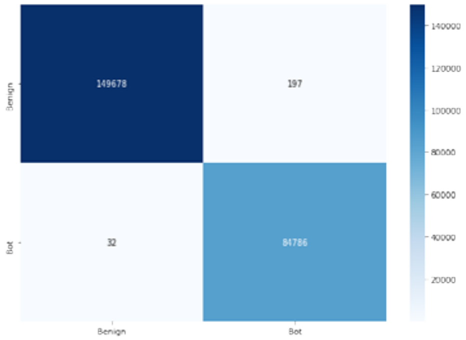

Fig. 3. Confusion Matrix MLP with ReLU Activation Function.

b) 3-classes dataset: The results shown in Table IV show that when increasing the number of classes of the dataset that need to be classified to 3, the F1-score decreased greatly. The average F1-score when using 4 hidden units is higher than when using 2 hidden units. However, the highest F1 was achieved in the case of using 2 hidden units with the Identity activation function. The result of using 4 hidden units and Relu activation function was exceptionally low at 16.67%. Fig. 4 depicts the results of the Confusion Matrix.

TABLE IV. EXPERIMENTAL RESULTS OF NETWORK INTRUSION DETECTION USING MLP ALGORITHM WITH THE 3-LABELS DATASET

Parameters		**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
Hidden Units	***Activation function***				
2	**identity**	**96.10**	**96.14**	**96.10**	**96.10**
	logistic	76.77	85.45	76.69	73.31
	tanh	75.35	80.96	75.27	72.14
	ReLU	33.35	11.11	33.33	16.67
4	identity	90.01	90.43	90.01	89.85
	logistic	90.28	90.23	90.26	90.17
	tanh	88.20	89.57	88.17	88.02
	ReLU	33.35	11.11	33.33	16.67

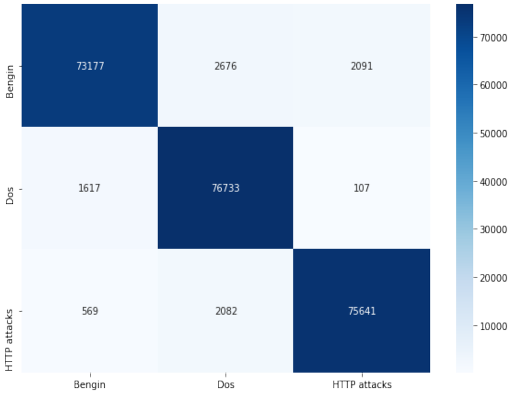

Fig. 4. Confusion Matrix of MLP with Activation Function as Identity.

3) Experimental results with CNN: The CNN network consists of an input layer, hidden layers, and an output layer with corresponding parameters. After many experiments, we found that processing data with Convolution Layers with parameters {filter = 32, 39, 64; filter size = 5; batch size = 32} is optimal. Learning rate parameters of 0.01, 0.001, and 0.0001 were also run to select the most optimal parameter. Based on these results, seeing that a learning rate of 0.0001 gave the best results. Table V describes information about the network models that were selected and tested.

Based on the parameters in Table V, this paper performed with 50 epochs and all Convolution layers used the ReLU activation function.

a) 2-classes dataset: Through results in Table VI, seeing that the CNN model with 1D-CNN achieved very good performance in terms of accuracy, precision, recall, and F1-score. The 1D-CNN 2-layers had the highest performance in 3 models and did not need enough 50 epochs to produce high results. Besides, Fig. 5 presents the accuracy of the training and test process of 1D-CNN 2-layers. Based on it, seeing that this model had an accuracy of approximately 100% after only 23 epochs and maintained that state until the end of the training process. This model detected most attacks (only 8 attack records were not detected). For normal network traffic, the number of false positive record is just 1.

TABLE V. CONFIGURE PARAMETERS OF THE CNN MODEL

Model	**Architecture detail**
1D-CNN 1 layer	Conv1D(32,5)-MaxPool(2)-FC()-FC()
1D-CNN 2 layers	Conv1D(32,5)-Conv1D(64,5)-MaxPool(2)-FC()-FC()
1D-CNN 3 layers	Conv1D(32,5)-Conv1D(64,5)-MaxPool(2)-Conv1D(39,5) -MaxPool(2)-FC()-FC()

TABLE VI. EXPERIMENTAL RESULTS OF NETWORK INTRUSION DETECTION USING CNN ALGORITHM WITH THE 2-LABELS DATASET

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
1D-CNN 1 layer	99.98	99.98	99.98	99.98
1D-CNN 2 layers	**99.994**	**99.994**	**99.994**	**99.994**
1D-CNN 3 layers	99.9936	99.9936	99.9936	99.9936

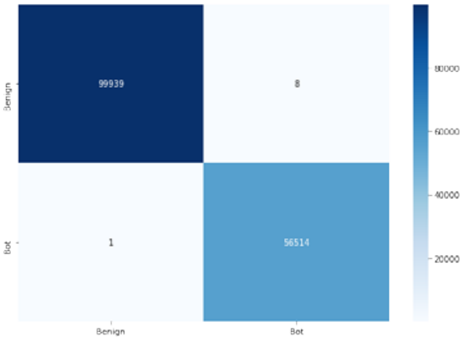

Fig. 5. Confusion Matrix of CNN with 1D-CNN 2 Layers.

b) 3-classes dataset: From Table VII, seeing that the CNN model with 1D-CNN yielded outstanding results on all metrics including accuracy, precision, recall, and F1-score. Besides, for the 3-labels attacker dataset, the 1D-CNN 3-layers had the best performance in the 3 models. The accuracy of the training and testing process of 1D-CNN 3-layers shows in the figure below. It can be seen that this model had an accuracy of approximately 100% after 50 epochs. Fig. 6 below depicts the results of the CNN model with 1D-CNN 3-layers.

TABLE VII. EXPERIMENTAL RESULTS OF NETWORK INTRUSION DETECTION USING CNN ALGORITHM WITH THE 3-LABELS DATASET

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
1D-CNN 1 layer	99.937	99.937	99.937	99.937
1D-CNN 2 layers	99.984	99.984	99.984	99.984
1D-CNN 3 layers	**99.986**	**99.986**	**99.986**	**99.986**

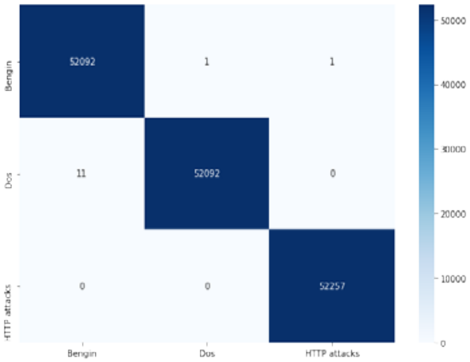

Fig. 6. Confusion Matrix of CNN with 1D-CNN 3 Layers.

D. General Evaluation

Table VIII below shows the overall comparison results of the RF, MLP, CNN classification algorithms with 2-classes and 3-classes dataset.

TABLE VIII. COMPARISON RESULTS OF RANDOM FOREST (RF), MLP, CNN CLASSIFICATION ALGORITHMS

Number of classes	**Algorithm**	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
2	RF (50 trees)	**99.995**	**99.995**	**99.995**	**99.995**
	MLP (relu, 2 units)	99.90	99.76	99.96	99.86
	CNN (1D-CNN 2 layers)	99.994	99.994	99.994	99.994
3	RF (100 trees)	99.967	99.967	99.967	99.967
	MLP (identity, 2 units)	96.10	96.14	96.10	96.10
	CNN (1D-CNN 3 layers)	**99.986**	**99.986**	**99.986**	**99.986**

With the results shown in Table VIII, with 2-labels and 3-labels dataset, algorithms CNN, Random Forest, and MLP all gave classification results with not too large differences on evaluation metrics. However, in the case of 2 classes, the Random Forest algorithm with 50 trees gave a score about 0.01% higher than CNN (1D-CNN 2-layers). And in the case of 3 classes, the CNN (1D-CNN 3-layers) algorithm gave better classification results than the Random Forest with 100 trees (0.0189% higher). This is not a large number. However, with the actual dataset, it is a quite far distance and has a great impact on the prediction. Therefore, depending on the model of the problem, we will build according to the Random Forest or CNN algorithms. From the data, seeing that with a large amount of data, the number of incorrectly predicted records of the two algorithms is quite much different. Therefore we recommend using CNN rather than Random Forest or MLP algorithms although we must define the network's architecture including the number of layers, decision function, etc.

V. CONCLUSION

Unauthorized network intrusion techniques will transform increasingly to bypass the surveillance of attack detection systems. This requires intrusion detection systems to be constantly updated on the abnormal signs and behavior of network attacks. In this paper, based on analyzing behaviors of network intrusion in network traffic, we have succeeded in determining attack behaviors and normal behaviors of the network data. The scientific and practical significance of the paper is shown in the classification and feature extraction. Accordingly, in our research, we did not extract typical features of cyber-attacks. Instead, we tried to enumerate fully their components and characteristics in the network and then use machine learning and deep learning algorithms to classify. With this approach, we have greatly reduced the time cost of finding and extracting features of network attacks. In addition, based on the experimental results, we have proven that our approach and proposal in this paper are correct and reasonable. This result shows that the proposal using behavior analysis techniques of network traffic using machine learning and deep learning techniques not only helps to accurately detect network intrusion techniques but also contributes to improving the time of seeking and extracting features. Besides, based on the experimental results of Random Forest, CNN, and MLP algorithms with different parameters, seeing that the 2-label dataset gave better results than the 3-label dataset. This shows that: the more optimal the standardization of models and data is, the more accurate the classification is; should not clearly distinguish the labels of network intrusion techniques in the dataset. In the future, we will research and use other analysis methods to improve the efficiency of the detection method based on this dataset. In particular, because our behavior analysis technique has extracted statistical features of network traffic, these features express the correlation not only in terms of data but also in terms of time. Therefore, it is necessary to have algorithms and analysis methods to highlight the time factor in behavior.

REFERENCES

[1] Gilberto Fernandes Jr., Joel J. P. C. Rodrigues, Luiz Fernando Carvalho, Jalal F. Al-Muhtadi & Mario Lemes Proença Jr., “A comprehensive

survey on network anomaly detection," Telecommunication Systems, vol. 70, pp. 447–489, 2019.

[2] Mohiuddin Ahmed, Abdun Naser Mahmood, Jiankun Hu, "A survey of network anomaly detection techniques," Journal of Network and Computer Applications, vol. 60, pp 19-31, 2016.

[3] Kh. Ansam. et al., "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 20, pp. 2-20, 2019.

[4] Sebastián García, Alejandro Zunino, Marcelo Campo, "Survey on network-based botnet detection methods," Security Comm. Networks, 2013. https://doi.org/10.1002/sec.800.

[5] Monowar H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, "Network Anomaly Detection: Methods, Systems and Tools," IEEE Communications Surveys & Tutorials, vol. 16 (1), pp. 303–336, 2014.

[6] CSE-CIC-IDS2018 on AWS. https://www.unb.ca/cic/datasets/ids-2018.html.

[7] R. Markus., et al., "A survey of network-based intrusion detection data sets," Computers & security, vol. 8, no. 6, pp. 147–167, 2019.

[8] K. Vikash., et al., "An integrated rule based intrusion detection system: analysis on UNSW-NB15 data set and the real time online dataset," Cluster Computing, vol. 22, doi: 10.1007/s10586-019-03008-x, 2019.

[9] N. Moustafa., et al., "Novel Geometric Area Analysis Technique for Anomaly Detection using Trapezoidal Area Estimation on Large-scale Networks," IEEE Transactions on Big Data, vol. 5, no. 4, pp. 2332-7790, 2017.

[10] N. Moustafa et al., "Big Data Analytics for Intrusion Detection System: Statistical Decision-Making Using Finite Dirichlet Mixture Models," doi: 10.1007/978-3-319-59439-2_5, 2017.

[11] S. Bagui, et al., "Using machine learning techniques to identify rare cyber-attacks on the UNSW-NB15 dataset," Security and Privacy, doi: 10.1002/spy2.91, 2019.

[12] S. Rajagopal., et al., "A predictive model for network intrusion detection using stacking approach," International Journal of Electrical and Computer Engineering (IJECE), vol. 10, no. 3, pp. 2734-2741, June 2020.

[13] S. Rajagopal, et al., "Performance analysis of binary and multiclass models using azure machine learning," International Journal of Electrical and Computer Engineering (IJECE), vol. 10, no. 1, pp. 978-986. February 2020.

[14] H. H. Ibrahim, et al., "A comprehensive study of distributed Denial-of-Service attack with the detection techniques," International Journal of Electrical and Computer Engineering (IJECE), vol. 10, no. 4, pp. 3685-3694, August 2020.

[15] Cho Do Xuan, Hoang Thanh, Nguyen Tung Lam, "Optimization of network traffic anomaly detection using machine learning," International Journal of Electrical and Computer Engineering, vol. 11, no. 3, pp. 2360-2370, 2021.

[16] Cho Do Xuan, Lai Van Duong, Tisenko Victor Nikolaevich, "Detecting C&C Server in the APT Attack based on Network Traffic using Machine Learning," International Journal of Advanced Computer Science and Applications(IJACSA), vol. 11(5), 2020. http://dx.doi.org/10.14569/IJACSA.2020.0110504.

[17] Cho Do Xuan, Hoang Mai Dao, Hoa Dinh Nguyen, "APT attack detection based on flow network analysis techniques using deep learning," Journal of Intelligent & Fuzzy Systems, vol. 39, no. 3, pp. 4785-4801, 2019.

[18] David Zhao, Issa Traore, Bassam Sayed, Wei Lu, Sherif Saad, Ali Ghorbani, Dan Garant, "Botnet detection based on traffic behavior analysis and flow intervals," Computers & Security, vol. 39, pp. 2-16, 2013.

[19] Sudipta Chowdhury, Mojtaba Khanzadeh, Ravi Akula, Fangyan Zhang, Song Zhang, Hugh Medal, Mohammad Marufuzzaman & Linkan Bian, "Botnet detection using graph-based feature clustering," Journal of Big Data, vol. 4, no. 14, 2017.

[20] Abdulghani Ali Ahmed, Waheb A. Jabbar, Ali Safaa Sadiq, Hiran Patel, Journal of Ambient Intelligence and Humanized Computing, 2020. https://doi.org/10.1007/s12652-020-01848-9.

[21] CICFlowMeter. http://www.netflowmeter.ca/netflowmeter.html. Accessed 1 November 2020.

[22] L. Breiman., "Understanding Random Forests: From Theory to Practice," Machine Learning, vol. 80, no. 1, pp. 5-32, 2017.

[23] Samaneh Mahdavifar, Ali A. Ghorbani, "Application of deep learning to cybersecurity: A survey," Neurocomputing, vol. 347, pp. 149–176, 2019.

[24] Cho Do Xuan, Dao M.H, "A novel approach for APT attack detection based on combined deep learning model," Neural Comput & Applic, 2021. https://doi.org/10.1007/s00521-021-05952-5.

[25] Cho Do Xuan, "Detecting APT Attacks Based on Network Traffic Using Machine Learning," Journal of Web Engineering, vol. 20(1), pp. 171-190, 2021.