Intrusion Detection Attacks Classification using Machine Learning Techniques

[1]**Majdi Alqdah**
[1]*Department of Computer Science, Faculty of Science and Information Technology, Zarqa University, Jordan*

[**]***Corresponding Author: malqdah@zu.edu.jo***

Received: *02.01.2021,*
Revised: *05.04.2021,*
Accepted: *15.07.2021,*
Published: *16.08.2021*

DOI:
10.53409/mnaa/jcsit/2201

Abstract: Distributing numerous services over the internet is called Cloud Computing. Applications and tools like networking, data storage, databases, servers, software are examples of the resources. The service provider is required to provide the resource always and from any location. However, the network is the most important factor in gaining access to data in the cloud. When leveraging the cloud network, the cloud threats take advantage. An intrusion Detection System (IDS) observes the network and detects and reports threats. The anomaly method is significant in Intrusion Detection Systems. IDS monitors known and unknown data whenever a virtual machine is developed. If any anonymous data is detected, the Intrusion Detection System identifies it using an anomaly classification algorithm and sends a report to the administrator. Naive Bayes, Decision tree (CART), Support Vector Machine, and random forest techniques are utilized in this work to classify unknown data. These algorithms are assisting in reducing the percentage of false alarms. This proposed work was carried out utilizing the WEKA tool for generating the report, yielding a best result in less computing time.

Keywords: Anomaly Detection, Decision Tree, Naive Bayes, SVM, Random-forest, NSL-KDD dataset.

I. INTRODUCTION

Cloud computing exists at a remote place and delivers service through the networks. The applications like data storage, infrastructures, server, and database can be developed, accessed, and manipulated by the user [1]. The users could access everything as a service like infrastructures, platforms, and software from the cloud wherever in the global via the internet and the cloud-based connection with two ends. The front end should connect with the users and the user requires resources such as software/hardware to implement for application development, database maintenance, and service delivery through the network. The other end of the chain should communicate with a third party and cloud. On the physical layer, the virtual machine monitors (ex., IDS) and runs on several virtual machines. The development tools, database server, web and application servers are completely maintained by the third party [2].

Cloud computing is rapidly being used by the government, corporate sector, institutions, medical, and other organizations. However, they must provide a high level of security because many network attacks target the cloud. Conventional attacks include DDoS, IP spoofing, Port Scanning, User to Port, and so on. An IDS is a novel efficient technique for protecting packets in a regular network. The function of an intrusion detection system (IDS) is monitoring the networks and forecast harmful behaviour before reporting it to the cloud administrator. If an intrusion is identified, the IDS generates an alert signal to keep a constant observation on the event, whether it is a false alarm or true positive. The cloud network IDS has been installed on the cloud servers and is maintained by the service providers. The IDS manage large scale computer systems, scalability, automation, and synchronization [3-5].

The network IDS should choose and limit the count of features that may be easily derived from high data speed. Because the local area network's ability to forward packets at one gigabit per second is dependent on the speed of hard disk. Though, the speed of the hard disk is slower. The framework's minimum size is 64 bytes. As a result, 1 to 14.8 million frames per second may be transmitted. The network is monitoring the data during this transaction, which is a key problem in cloud computing. The most essential problem is data detection in practical [6].

The primary objective of this work is to predict data utilizing four algorithms on anomaly-based approach. These techniques are used in cloud computing to create an effective system for detecting intrusion and selecting features from dataset properties using various tools. The anomaly-based approaches are covered in detail in this work. The remaining part of the work is organized as follows: The anomaly-based approaches are briefly described in related works. The proposed technique then discusses several algorithms used for classification, followed by a discussion of datasets in the next section. Section four summarizes the experimental data, while the final section concludes and discusses future work.

II. RELATED WORK

A hybrid network intrusion detection system was intended to be installed in each host layer of a virtual network. The network-IDS has observed the network traffic and reported onto the upper layers using signature and anomaly-based methods [7]. Only distinguished attacks from the signature database were detected by the misuse approach. To identify the attacks, the snort rule fast multi-pattern matching method was used. However, the anomaly-based method supported in the detection of unknown threats. Statistical modelling, data mining, and machine learning approaches were employed in anomaly-based systems. Different machine learning classification methods were employed. These approaches were used in anomaly-based intrusion detection systems, and they provided improved accuracy and secrecy, as well as lower computational time and false alarms [8-9].

Mohammed et al. developed two methods for classification in [10]. For classifying attacks, one technique was the Support Vector Machines, while other was the Random Forests. 90% of the data set was utilized to train the models and 10% was used to test the models, which was inadequate to validate the attack detection rate. Although the computational time was short, the detection rate was not as high as predicted. As a result, more test case results were required.

In [11], Nabila F et al. implemented a random forest technique in IDS. The NSL-KDD data set was utilized to compare the performances of a random forest model in detecting attacks such as DoS, U2R, Probing, and R2L against other conventional attacks. However, they were increasing the classifier's accuracy in features selection measures. The data mining idea combined with IDS was proposed in [12] and finds related data, concealed data, and associated data with reduced execution time. They utilized several classification methods on the KDD dataset. This method performed well in terms of false alarm and accuracy rates. However, two difficulties like inadequacy of user data and the approaches have prevented the development of an autonomous IDS.

In [13], Xueyan J et al. implemented the Fuzzy C-Mean technique for clustering to train from data set and the K-Nearest Neighbor approach to classify attacks unknown. However, additional testing and training data were required for better outcomes. Rahimeh R et al. presented an anomaly detection approach in [14]. This approach was evaluated utilizing the KDDCup-99 data set. The output showed that this model, when used in feature selection from the KDDCup-99 dataset, efficiently recognized attacks. This study employed the feed-forward neural networks model that has been trained to detect the attack/normal packets in the data set. However, more training and testing datasets were required for this study.

Opeyemi O et al. presented the decision tree method for classification in [15] to identify DDoS attack. This work used features selection techniques; however, the confusion matrix values were not employed. The detection rate and accuracy were not specified. There was no explanation on how to use the decision tree categories. In [16], Ozge cephelli et al. presented a detection method that used traffic packets and sensitivity settings from a network. To identify the DDoS attack, the presented Hybrid-IDS was deployed. As a result, when training performance was necessary, the model's performance was reduced. For assessment, limited samples of the DARPA-2000 dataset were used. The proposed model was unclear; therefore, the detailed commercial bank dataset was obtained via a penetration testing tool.

III. PROPOSED METHODOLOGY

The IDS are divided into two types: signature-based approaches and anomaly-based techniques. The snort rule was used to detect known attacks in signature-based and anomaly-based detection. In anomaly detection, several classification approaches such as Nave Bayesian, Decision tree, SVM, and random forest algorithms are utilized to identify unknown threats.

It is difficult to identify infiltration during severe load by observing real-time traffic. It provides a network intrusion detection solution. Snort is a very adaptable rule, and it is simple to change, unlike commercial NIDS. Snort supports four methods (packet logger, sniffer, intrusion detection system, and prevention system) [16]. In Snort rule, the users can write their own rules for outgoing and incoming network packets, and it consists of two segments: "The Options" and "Header." If the packets should fulfil the threshold conditions, the snort rule is all that is required [18].

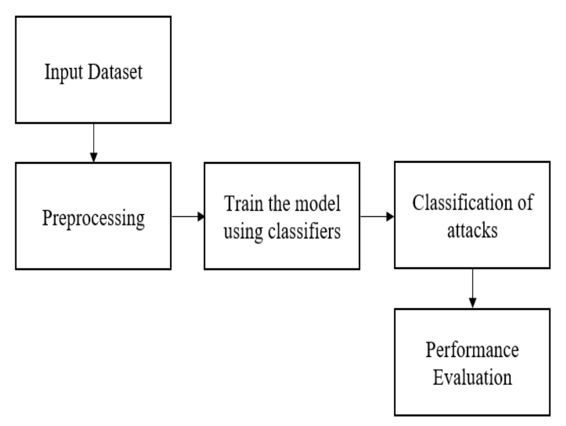

Figure 1. Workflow of Proposed Model

3.1. Naïve Bayes Classification

Naïve bayes is a supervised learning classifier and a statistical approach for classification. The learning process generates a functionality that predicts the values of output. Following training data, the model produces target for new values of input. Given that this approach represents the class variables and that the collection of characteristics is $h_1, h_2 \dots, h_n$.

$$p(g/h_1, h_2 \dots, h_n) = \frac{p(h_1,h_2 \dots,h_n/g)\, p(g)}{p(h_1,h_2 \dots,h_n)} \quad (1)$$

For all $i = 1, 2, \dots, n$ it becomes $p\left(\frac{h_i}{g}\right)$

$P(h/g)$ denotes the g given h probability. P(g) denotes the hypothesis g's previous probability. $P(h)$ represents the training data h's previous probability. $P(g/h)$ was the g's probability given h in the following equation classification technique to aid in the improvement of IDS speed and accuracy [19].

3.2. Decision Tree

Decision tree was the part of algorithms for supervised learning. The principles of DT are simple to comprehend and use in learning systems like the Weka tool. In this work, the CART (Classification and Regression Tree) method is used. The main goal of this DT rule was to build the training and prediction of class value model. The information gain ratio was a value utilized to choose the splitting feature in this case. The decision tree is characterized as a tree structure, with a decision and leaf nodes. The decision node was the root node, with every internal nodes representing a feature and each leaf node representing a class value. The windows are made up of numerous classifiers such as bayes, meta, functions, and trees. The following equation is used to calculate the entropy of a feature *E*.

$$Entropy\ (E) = -\sum_{i=1}^{n} p(E,i) \log\big(p(E,i)\big) \ (2)$$

Let A be the total number of intrusion classes in the provided dataset, and *p* (*E*, *i*) represents the percentage of instances in *E* allocated to the *i*th class. The dataset *G*'s information gain is computed as in following equation:

$$Gain(D,T) = I(D) - I(D,T) \quad (3)$$

Gain (*D*, *T*) of a feature *T* is impacted by the domain size of *T* and is greatest when each subset D_i contains just one record.

A feature's split information *Split* (*D*, *T*) has a larger domain size, typically increases. Each feature's split information is calculated as follows.

$$Split(D,T) = -\sum_{i=1}^{k} \frac{|D_i|}{|D|} \times log_2 \frac{|D_i|}{|D|} \quad (4)$$

Where the domain size of *T*, $|T| = k$.

To choose the feature with the highest gain ratio, the optimal split node was chosen in equation 4. This decreased the complexity of computation [20].

3.3. SVM Classifier

For classification and regression, mostly the SVM learning method is used. However, it is mostly employed in classification issues. SVM operates in two classes using a hyperplane. The classification is completed by utilizing a hyperplane, which was developed by the greatest margin in the training data [10][21].

The network intrusion detection system detects network intrusion. At times, the Network-IDS was unsure if the network packets were abnormal or normal. Machine learning approaches are utilized to classify normal and abnormal packets in that important scenario.

Step 1: Input the data set that includes 41 characteristics and features.

Step 2: Preprocess the data to remove unnecessary and redundant information.

Step 3: Use machine learning classification algorithms (Nave Bayes, CART, SVM and Random Forest) to classify data.

Step 4: The classification algorithms that were utilized to construct the models (trained model)

Step 5: Determine if the data was abnormal or normal.

Step 6: Finally, the performance of the classifier approaches was compared.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The NSL-KDD dataset, an enhanced version of the KDD dataset, is employed for assessment in this case. The features of the NSL-KDD dataset classified in this work can detect attacks such as DOS, Probe, R2L, U2R, and so on. It is also mostly utilized for detecting abnormal attacks. The benefits of the NSL-KDD dataset were described individually. First, because the training set does not include unnecessary entries, the classification cannot provide a partial output. The following step was to process the unnecessary entries with the testing dataset. NSL-KDD has resulted in higher reduction rates [22]. Each record includes 42 features that contain data as well as the network's five different classifications. One is the original class, and the other four are assault classes, which were Probe, DOS, U2R, and R2L. Table 1 displays the most common forms of attacks in the testing and training datasets [23].

Table 1. Dataset Description

Class	Training	Testing
Normal	67343	9711
DoS	45927	7458
Probe	11656	2754
R2L	995	2421
U2R	52	200
Total	125973	22544

To undertake classification testing, the experimental setup utilizes the NSL-KDD data set and the automated data analysis WEKA tool. Weka was a data mining technique that includes clustering, preprocessing, regressions, features selection, and classification model. It is compatible with the Windows operating system. To accomplish the classification, just 20% of the NSL KDD data set was needed. The presentation of the classifier is assessed using modified metrics like true positive, false positive, accuracy, and computational time.

The dataset was initially classified before preprocessing, and the classification range was 0 to 1. (i.e., can chose this range similar to 0.01, 0.05, or 0.10). The dataset has almost 41 features available for selection. This classification effort employs the Naive Bayes, CART, SVM, and Random Forest methods. The experiment was carried out with the help of WEKA (Waikato Environment for Knowledge Analysis) tool. The initial stage was to preprocess only samples of data before classifying them with the algorithms.

The data from the NSL-KDD data set has been accessed. In the Weka tool, the machine learning algorithms were used to detect diverse outcome results, and just 20% of the data set was used to evaluate the training set. The Weka tool produces superior results, and Table 2 shows the percentage performance of the classification methods produced: TPR, FPR, Accuracy, and computational time.

$$TPR = \frac{TP}{(TP+FN)} \qquad (5)$$

$$FPR = \frac{FP}{(FP+TN)} \qquad (6)$$

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \qquad (7)$$

The accuracy value is the percentage of correctly classifies cases in relation to the total number of occurrences.

Table 2. Performance Analysis of Proposed Models

Classifiers	TPR	FPR	Accuracy	Computation Time (ms)
Naïve Bayes	88.66	0.8	92.51	520
CART	97.92	1.5	97.18	186
SVM	98.35	0.6	98.27	120
Random Forest	98.87	0.5	99.02	135

Many intruders attacked the virtual machine while packets out of the source IP addresses to the destination IP transit the networks. When compared to Naïve Bayesian classification, SVM, CART, and Random Forest algorithms produce superior results. Below is a graphical depiction of the performance result. Figures 2 and 3 show the TP and FP rates based on the machine learning method and the NSL-KDD dataset, respectively.

Table 2 compares the TPR utilizing 41 features for training to four methods, with the random forest achieving 98.87 percent. It also displays the FPR findings utilizing 41 features for data training, where the random forest classifier achieved the lowest false positive rate.

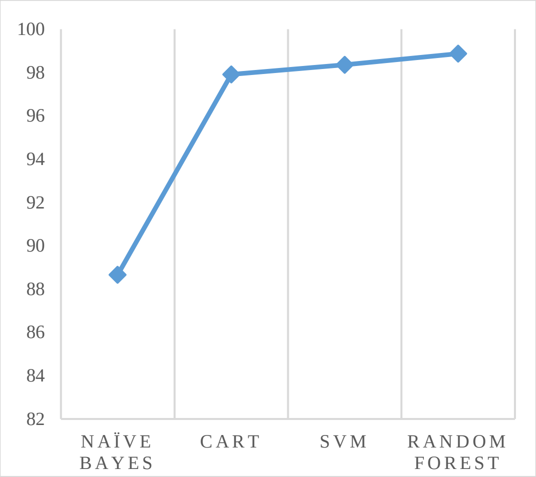

Figure 2. Graphical Plot Comparison of TPR

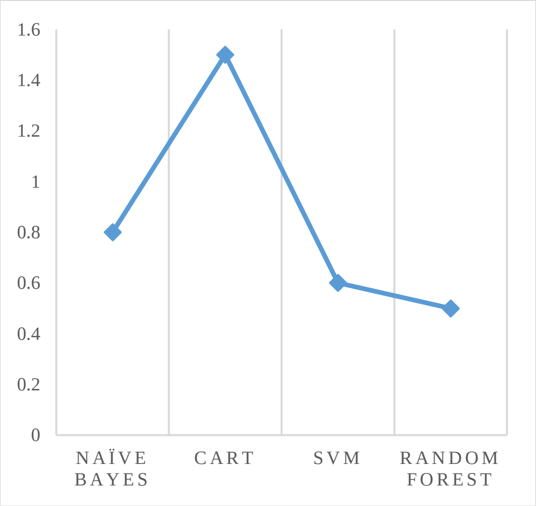

Figure 3. Graphical Plot Comparison of FPR

The accuracy and computational time are compared in Figures 5 and 6. Random-forest has a greater accuracy value than SVM, CART, and Nave Bayesian classifiers. And when the computational time was compared to others, the SVM comes out on top.

The graphical representations illustrated the performance of the four machine learning algorithms with the results. The maximum TPR for any algorithm is 90%, however the random forest has 98.87 percent TPR and an extremely low FPR. When compared to the other classifiers, the random forest outperformed them all. Its accuracy is 99.02 percent, and the random forest computing time is minimal.

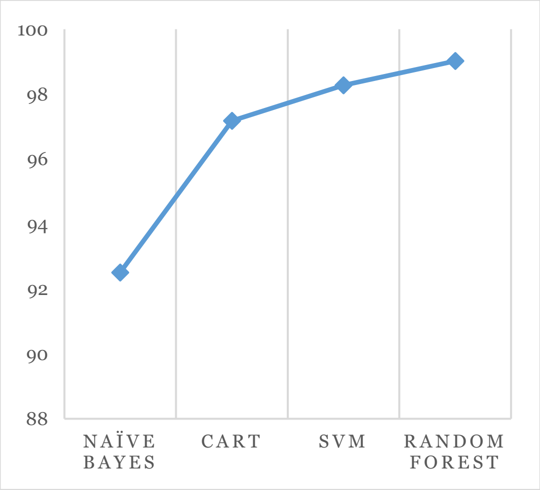

Figure 4. Graphical Plot Comparison of Accuracy

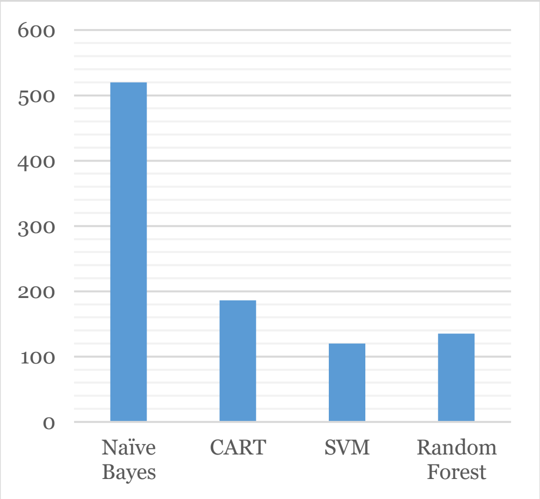

Figure 5. Graphical Plot Comparison of Computational Time

V. CONCLUSION AND FUTURE WORK

In this research, the performance of an intrusion detection system based on various machine learning algorithms was analyzed. The NSL-KDD dataset was used in this work to test Naive Bayes, SVM, CART, and Random Forest algorithms. There are 41 features accessible in this dataset. It mimics the training data by replicating the pre-processed dataset. Conventional attacks include DDoS, IP spoofing, Port Scanning, User to Port, and so on. The IDS has a novel effective approach for safeguarding packets in a conventional network. To increase the accuracy and lower false alarm, dataset samples for anomaly approaches were employed. The simulation results revealed that random forest outperforms in terms of TP rate by almost 0.5 percent to 10.2 percent. The random forest has a higher FP rate than others, although it is 1% lower than CART. The random forest classifier outperforms other classifiers in terms of accuracy, with an increase ranging from 0.7 to 6.5 percent. SVM also has a faster execution time than others. The key finding is that random forest outperforms alternative classifiers. According to the proposed research, an effective network IDS technique in cloud computing was established. In the future, the optimum feature selection method may be used to minimize the attributes and construct the training model.

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

Not applicable.

HUMAN AND ANIMAL RIGHTS

No animals/humans were used for studies that are basis of this research.

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The authors confirm that the data supporting the findings of this research are available within the article.

FUNDING

None.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

The authors would like to thank their present employer for providing support while carrying out this research work.

REFERENCES

[1]. Uttam K, Bhavesh N.G. A survey on intrusions detections system for cloud computing environments. Int. J. Comput. Appl., 2015, 109(1), 6–16.

[2]. Arjunan K, Chirag N.M. An enhanced intrusions detections frameworks for securing networks layers of clouds computing. ISEASP, 2017, pp. 1-10.

[3]. Mahalaksmi B, Susendran G. Effectuations of Secured Authorized Deduplications in Hybrid Clouds. Indian J Sci Technol., 2016, 9(25), 1-7.

[4]. Nathiya T. Reducing DDOS Attacks Technique in Cloud Computing Networks Technology. Int J Innov Res Appl Sci Eng., 2017, 1(1), 23–29.

[5]. Bathlaa RK, Susendran G, Shalu. Research analyses of bigdata and cloud computing with the emerging impacts of testing. Int J Eng Technol., 2018, 7(3), 239–243.

[6]. Ralf C S, Christian W O. Extracting salient feature for networks intrusions detections using machine learning method. S. Afr. Comput. J., 2014, 52(7), 82–96.

[7]. Snehal GK, Deepti PT. A Review on Intrusions Detections Technique for cloud computing and Security Challenge. International Conferences on Electronics and Communications System, 2015, pp. 227–232.

[8]. Kamatchi, A, Chriag N M. An Efficient Security Frameworks to Detects Intrusion at Virtual Networks Layers of Clouds Computing. International ICIN Conferences - Innovation in Cloud, Internet and Network, 2016, pp. 133–140.

[9]. Natiya T, Susendran G. An Effective Hybrid Intrusions Detections Systems for Uses in Security Monitor in the Virtual Networks Layers of Clouds Computing Technology, Data Managements, Analytic and Innovations, Advance in Intelligent System and Computing, 2019, 839, 483-496.

[10]. Mohammed AH, Nasser M, Pal B, Ahmad S. Support Vector Machines and Random Forests Modeling for Intrusions Detections Systems (IDS). J Intell Learning Syst Appl., 2014, 6(2), 45–53.

[11]. Nabila F, Jabar MA. Random Forests Modelling for Networks Intrusions Detections Systems. Procedia Comput. Sci, 2016, 89, pp. 213–218.

[12]. Nadiamai GV, Hemalatha M. Effective approaches towards Intrusions Detections Systems using data mining technique. Egypt. Inform. J., 2014, 15(1), 37–50.

[13]. Xueyan J, Yingtao B, Hai D. An innovative two-stages fuzzy kNN-DST classifiers for unknown intrusions detections. Int. Arab J. Inf. Technol., 2016, 13(4), 359–366.

[14]. Rahimeh R, Farshid K, Mehran A. Improving the Intrusions Detections System's Performances by Correlations as a Samples Selections Method. J. Comput. Sci. Appl., 2013, 1(3), pp. 33–38.

[15]. Opeyemi O, Cai H, Choo KK, Dehghantanha A, Xu Z, Dlodlo M. Ensemble-based multi-filters features selections methods for DDoS detections in cloud computing. EURASIP J Wirel Commun Netw., 2016, 130, 1-10.

[16]. Özge C, Saliha B, Güneş KK. Hybrid Intrusions Detections Systems for DDoS Attack. Int. J. Electr. Comput. Eng., 2016, 2016.

[17]. Nidal M T, Anas AT, Masadeh RS. Cloud Computing Challenge and Solution. Int. J. Comput. Netw. Commun., 2013, 5(5), 209–216.

[18]. Neminath H, Vinoth S. False alarms minimizations technique in signatures-based intrusions detections system: A survey. Comput Commun., 2014, 49, 1–17.

[19]. Kian M A C, Hai T H, Hwee T N. Bayesian Online Classifier for Texts Classifications and Filtering. Proceeding of the 25th annual internationals ACM SIGIR conferences on Research and Developments in Information Retrievals, 2002, pp. 97–104.

[20]. Harvinder C, Anu C. Implementations of decision tree algorithms c4.5. Int. J. Sci. Res., 2013, 3(10), 4–6.

[21]. Ozgur C F, Balabhan M.E. Cloud-SVM: Training an SVMs classifiers in cloud computing system. Joint International Conferences on Pervasive Computing and the Networked World, 2013, pp. 57–68.

[22]. Revathi S, Malathi A. A Detailed Analyses on NSL-KDD Datasets Using Various Machines Learning Technique for Intrusion Detections. Int. J. Eng. Res. Technol., 2013, 2(12), 1848–1853.

[23]. Dhanabal L, Shantharajah SP. A Study on NSL-KDD Data set for Intrusions Detections Systems Based on Classifications Algorithm. Int. J. Adv. Res. Comput. Commun. Eng., 2015, 4(6), 446–452.

Cite this article as: Majdi A. Intrusion Detection Attacks Classification using Machine Learning Techniques. J. Comput. Sci. Intell. Technol. 2021; 2(2): 01-06. ©JCSIT, MNAA PUB WORLD, 2021.