Bulletin of Electrical Engineering and Informatics
Vol. 12, No. 1, February 2023, pp. 418~426
ISSN: 2302-9285, DOI: 10.11591/eei.v12i1.4555

Man-in-the-middle and denial of service attacks detection using machine learning algorithms

Sura Abdulmunem Mohammed Al-Juboori[1], Firas Hazzaa[1], Zinah Sattar Jabbar[2], Sinan Salih[2], Hassan Muwafaq Gheni[3]
[1]Ministry of Higher Education and Scientific Research, Baghdad, Iraq
[2]Department of Communication Technology Engineering, College of Information Technology, Imam Ja'afar Al-Sadiq University, Baghdad, Iraq
[3]Department Computer Techniques Engineering, Al-Mustaqbal University College, Hillah, Iraq

Article Info

Article history:

Received Aug 19, 2022
Revised Oct 1, 2022
Accepted Oct 20, 2022

Keywords:

Attacks detection
Classification metrics
Computer networks and communications
DoS attack
Machine learning
MTM attack

ABSTRACT

Network attacks (i.e., man-in-the-middle (MTM) and denial of service (DoS) attacks) allow several attackers to obtain and steal important data from physical connected devices in any network. This research used several machine learning algorithms to prevent these attacks and protect the devices by obtaining related datasets from the Kaggle website for MTM and DoS attacks. After obtaining the dataset, this research applied preprocessing techniques like fill the missing values, because this dataset contains a lot of null values. Then we used four machine learning algorithms to detect these attacks: random forest (RF), eXtreme gradient boosting (XGBoost), gradient boosting (GB), and decision tree (DT). To assess the performance of the algorithms, there are many classification metrics are used: precision, accuracy, recall, and f1-score. The research achieved the following results in both datasets: i) all algorithms can detect the MTM attack with the same performance, which is greater than 99% in all metrics; and ii) all algorithms can detect the DoS attack with the same performance, which is greater than 97% in all metrics. Results showed that these algorithms can detect MTM and DoS attacks very well, which is prompting us to use their effectiveness in protecting devices from these attacks.

Corresponding Author:

Sura Abdulmunem Mohammed Al-Juboori
Ministry of Higher Education and Scientific Research
Baghdad, Iraq
Email: sura.sultan.ss@gmail.com

1. INTRODUCTION

The internet of factors (IoT) is a idea of connecting thousands and thousands of devices over the net to exchange and percentage facts between those devices, like sensors, mobile phones, laptops, or actuators [1], [2]. These gadgets can interact with each other the use of many one-of-a-kindwireless verbal exchange strategies like Bluetooth, c084d04ddacadd4b971ae3d98fecfb2a, and ZigBee [1], [2]. The IoT has evolved because many a couples of technologies are converging, which includes commodity sensors, machine gaining knowledge of, embedded structures, and ubiquitous computing [3], [4]. Its miles stricken by several sorts of attacks to obtain and thieve statistics, like man-in-the-middle (MTM), adware, sq. injection, denial of provider, social engineering, and ransomware [3]. The man-in-the-center MTM is a 9aaf3f374c58e8c9dcdd1ebf10256fa5 assault, and its miles a cyber-attack in which the attacker discreetly transmits and may alternate the communications among two sufferers who expect they're interacting without delay with each different because the attacker has positioned himself among sufferers [5]–[8]. Simplest whilst the attacker mimics every sufferer nicely wi-fi to satisfy their

expectancies can MTM defeat mutual authentication in any community [5]–[8]. So, this assault could be very risky if it attacks the community that has critical information on its linked devices [5]–[8]. Every other 9aaf3f374c58e8c9dcdd1ebf10256fa5 assault that attacks the community known as denial of service (DoS). DoS is a cyber-assault that the attacker attempts to render a device or community supply inaccessible to stop-customers by using disrupting the host offerings, which are linked to the net whether forever or momentarily [9], [10]. Denial of provider is commonly executed by inundating the targeted laptop or aid with needless requests in try to overload structures and prevent some or all the multiple requests from being fulwirelesslled. As a result, if this attack goals a network with sensitive records on its related devices, it's miles extremely dangerous [9], [10].

The contribution to this paper is to build 4 machine learning algorithms which can be intense: i) gradient boosting; ii) random forest (RF); iii) decision tree (DT); and iv) gradient boosting, to detect two assaults received from the datasets on the Kaggle website. Those building algorithms may be used to lessen and defend the linked gadgets in any community. After obtaining the dataset, we follow preprocessing steps like wirelesslling in the lacking fee and changing some columns to numerical statistics kinds due to the fact these algorithms can cope with numeric data. Then, we use four classification metrics to evaluate the algorithms' performance: precision, accuracy, consider, and f1-rating. The remainder of this paper is as follows: segment 2 offers a few related paintings about detection of assaults. Section 3 describes the proposed method used on this paper. segment four presents the experimental results and discusses them. Section 5 wireless presents the realization of the paper and some future work. Plenty of researchers studied the detection of several IoT assaults from numerous sources the usage of gadget learning algorithms like DoS and MTM attacks. This segment describes the preceding associated work to stumble on DoS and MTM attacks the use of one of a kind device gaining knowledge of algorithms.

Rathee and Mann [11] used several gadget studying algorithms to be a malicious interest detector for DoS attacks. They accrued a dataset for 2 weeks, that is referred to as Canadian institute for cybersecurity (CIC) DoS dataset in-store customer experience (ISCX). This CIC dataset was gathered by using the University of New Brunswick in Canada and incorporates a variety of attributes like: i) the quantity of sent push acknowledgment (ACK) packets in a time-window, the ratio of reset packets in a time-window; ii) the quantity of despatched push ACK packets in a time-window; iii) the wide variety of despatched reset packets in a time-window; iv) the range of packets in a time-window; and v) the relationship duration. These algorithms are random wooded area, DT, Gaussian Naïve Bayes, logistic regression, k-nearest neighbour, guide vector system, and linear discriminant evaluation. To evaluate these algorithms, they used three evaluation metrics: accuracy (ACC), region below the relative operating characteristic (ROC) area under curve (AUC), and root imply rectangular blunders root mean squared error (RMSE). They've shown that the random woodland gave the first-class overall performance in detecting the DoS attack as follows: ACC=0.985, AUC=zero 0.972, and RMSE=0.030.

The have a look at [12] used a hard and fast of category algorithms to detect DoS assaults at the SNMP-MIB dataset. This dataset incorporates several statistics as follows: i) hypertext transfer protocol (HTTP) flood attack; ii) regular; iii) brute force attack; iv) the internet control message protocol (ICMP)-echo assault; v) user datagram protocol (UDP) flood attack; vi) slowpost attack; vii) transmission control protocol-synchronize (TCP-SYN) assault; and viii) slowloris assault. Then, they used twelve device gaining knowledge of algorithms: Naïve Bayes, J48, logistic model tree (LMT), random tree, logistic, Bayes net, sequential minimal optimization (SMO), multilayer perception, RF, instance based learning (IBK), simple logistic, Naïve Bayes, Naïve Bayes updatable, and multiclass classifier. They've proven that each one the algorithms except Naïve Bayes and Naïve Bayes updatable gave an excessive accuracy of 99.7.

The observe [13] cautioned a detection gadget to lessen and mitigate distributed (DDoS) assaults in the cloud computing surroundings. This gadget is based totally on a gadget mastering algorithm, which is referred to as the C.4.5 set of rules, and it makes use of different algorithms to validate its system, along with Naive Bayesian and ok method. They amassed a dataset associated with DoS assaults that includes the following attributes: land, provider, protocol, flag, initial time to live (TTL), and class (normal or DoS attack). they have got proven that the C.4.5 gave a better accuracy of 98.8% within the detection of DoS techniques.

Scire *et al*. [14] proposed a framework detection based totally on gadget getting to know algorithms to locate a DoS attack. They used two datasets: four-elegance containing 1,012,052 samples, and 7-elegance datasets containing 1,042,500 samples. The 4-elegance consists of 4 instructions: message queuing telemetry transport (MQTT-DoS), regular, MQTT-FUZZ, and transmission control protocol (TCP-DoS). While the seven-magnificence dataset which includes: MQTT-DOS-BF1, MQTT-FUZZ, ordinary, MQTT-DOS-BF3, MQTT-DoS-IAUTHS, MQTT-DoS-BF2, and TCP-DoS. The device gaining knowledge of algorithms that the machine is based on are: C.4.5 choice timber, average one-dependence estimator (AODE), and multi-layer perceptron (MLP). they have got shown that the AODE gave the higher accuracy in two datasets: 96.968% and 99.85%, respectively.

The have a look at [15] used neural networks and system studying methods to detect the DoS assault. They trusted many packages layer protocols including HTTP, file transfer protocol (FTP), hypertext transfer protocol secure (HTTPS), and secure shell (SSH), in addition to the CIC intrusion detection system (IDS) 2017

dataset relating to DoS attacks, which turned into accumulated by way of 25 users. FlowID, BwdPackets/s, SourceIP, MinPacketLength, SourcePort, MaxPacketLength, DestinationIP, PacketLengthMean, estinationPort, PacketLengthStd, Protocol, and PacketLengthVariance are only a few of the features in this collection. MLP and random woodland are the algorithms hired MLP. In evaluation to MLP, which had a 99.9563% accuracy, the RF had a better accuracy.

Wu *et al.* [16] proposed a new technique to locate and stumble on the MTM attack that came about in a wireless community amongst two nodes. They used a residual sum of squares (RSS) dataset that became acquired from a constructing, that is known as densely populated metropolitan. Then they used many device studying algorithms to do the detection technique: help vector device, Gaussian Nave Bayes, and k-nearest neighbor. The consequences showed that the Gaussian Nave Bayes and k-nearest neighbour gave the better prediction accuracy. Jones and Kumar [17] used a deep getting to know set of rules with network simulator 2 (NS2) simulation platform to hit upon the MTM attack, that's called synthetic artificial neural networks (ANN). They used a dataset with mobility styles and network-numerous site visitors conditions for a couple of attacks. They hired 4 assessment metrics to evaluate the ANN model: precision, accuracy, f1-score, and recall. They determined that the ANN had an accuracy fee of 88.235%. The study [18] proposed a detection version based totally-system mastering techniques to detect MTM from business manage structures. They accrued real-time data related to MTM that consists of many functions like temp max, cntt avg, cntt stdev, temp stdev, temp min, and temp avg. The version that the device is based on is ok-nearest neighbor. They have proven that the usage of the model primarily based on k-nearest neighbor gave the quality performance for detecting the MTM assault. The internet protocol (IP) spoofing guy-in-the-center category and identification detection system became evolved by the study [19] to discover the MTM version. They used the MTM dataset from Kaggle, which incorporates the following features: protocol type, duration, service, Dst bytes, land, incorrect fragment, pressing, Src bytes, flag, and hot. Then they applied a deep mastering technique referred to as the multilayer perceptron neural community, and that they evaluated it the use of a ramification of measures, which include accuracy, precision, and F1-score. They confirmed that this set of rules may want to stumble on the MTM with an accuracy of 83%. Banerjee and Chakraborty [20] proposed a version based on a supervised gadget getting to know technique to discover the MTM from an encrypted network. They accrued information regarding the MTM from three resources: Skype (63,782 packets), YouTube (113,146 packets), and WhatsApp (19,935 packets). Then, inside the identification section, they implemented 3 machines getting to know algorithms: first-rate Tree, 3-okay-nearest neighbor, and linear discriminant. The first-rate tree has the highest accuracy in 3 sources, with 96.7%, 99.3%, and 97.2%, respectively, as shown in Table 1.

Table.1 Previous research paper in detection DoS and MTM attacks

Ref	Year	Algorithm	Result
[16]	2015	Gaussian Naïve Bayes, support vector machine, and K-nearest neighbour	Gaussian Naïve Bayes, and K-nearest neighbour gave the best prediction accuracy.
[18]	2016	K- Nearest neighbour	Gave best results when detect the MTM attack.
[13]	2017	C.4.5, Naive Bayesian and K-means	Accuracy of C.4.5 is 98.8%
[15]	2018	RF and MLP	RF accuracy is 99.9563%.
[17]	2019	ANN	Accuracy: 88.235%
[11]	2019	Linear discriminant analysis, support vector machine, Gaussian Naïve Bayes, DT, RF, K- nearest neighbour, and logistic regression.	RF results: ACC: 0.985 AUC: 0.972 RMSE: 0.030
[12]	2020	Naïve Bayes, J48, RF, LMT, Random Tree, Naïve Bayes updatable, logistic, Bayes NET, SMO, multilayer perception, IBK, simple logistic, and multiclass classifier	All algorithms gave a higher accuracy rate with 99.7% except Naïve Bayes and Naïve Bayes Updatable
[14]	2020	AODE, C4.5, and MLP	Accuracy of AODE is: 99.968 %, 99.85%.
[19]	2020	MLP	Accuracy is 83%
[20]	2020	Fine tree, 3-K-nearest neighbour, and linear discriminant	Fine tree gave the higher accuracy in three sources

2. METHOD

Figure 1 shows the flow-chart of the proposed methodology to detect two well-known IoT attacks: DoS and MTM attacks; and which both contain several steps. The first one, we collected two datasets for DoS and MTM. In the second one, we applied several preprocessing steps to the collected dataset to make it more understandable for both humans and machines. The third one, we classified the MTM dataset into samples containing MTM attacks or normal samples, and we classified the DoS dataset into samples containing DoS attacks or normal samples. The fourth one, we used several machine learning algorithms to detect MTM and DoS attacks. In the final step, we used many classification metrics to assess the performance of the algorithms. We will explain the steps in more detail in the following subsections.

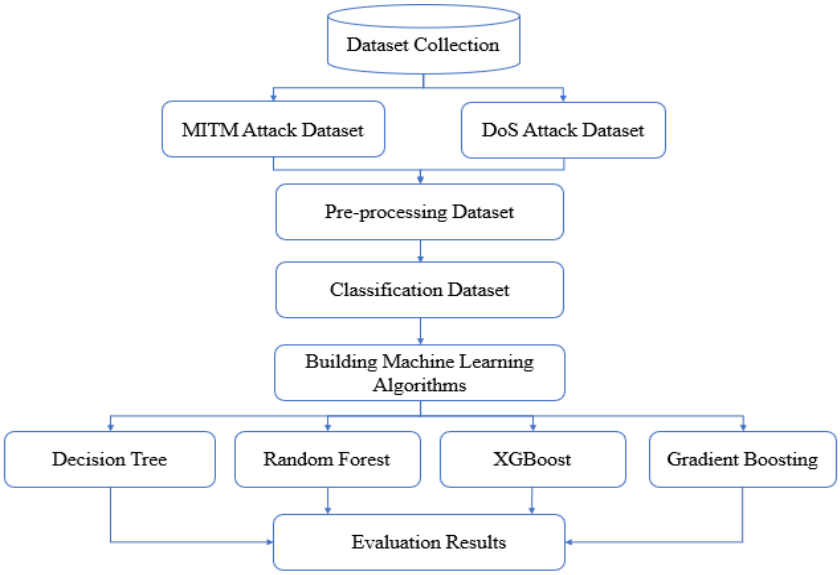

Figure 1. Proposed methodology

2.1. Dataset collection

In this paper, we used the dataset from the Kaggle website that related to two well-known IoT attacks. The MTM attack and the DoS attack [1]. This dataset contains many features, and each attack has different features based on the nature of the attack, as shown in Table 2.

Table 2. Dataset features, description, and the types

Field	Description	Type	MTM	DoS
frame time_delta	Time difference compared to the previous captured packet	timestamp	✔	✔
frame.time_epoch	Full time in the form of timestamp	timestamp	✖	✔
frame.time_relative	Time since the first package was sent	timestamp	✖	✔
ipv6.plen	IPv6 Payload length	numerical, 2 bytes	✔	✔
ipv6.nxt	Next Header	numerical, 1 byte	✔	✔
ipv6.src	Source IPv6 address	nominal-categorical	✖	✔
ipv6.dst	Destination IPv6 address	nominal-categorical	✖	✔
tcp.srcport	Source Port	numerical, 2 bytes	✔	✔
tcp.dstport	Destination Port	numerical, 2 bytes	✔	✔
eth.src	Source MAC address	nominal-categorical	✖	✔
eth.dst	Destination MAC address	nominal-categorical	✖	✔
frame.len	Frame length	numerical, 4 bytes	✔	✔
frame.number	Frame number	numerical, 4 bytes	✔	✔
mqtt.clientid	Client ID	nominal-categorical	✖	✖
mqtt.dupflag	Duplicate message flag	boolean	✖	✔
mqtt.hdrflags	Header flags	numerical, 1 byte	✖	✔
mqtt.kalive	Keep Alive	numerical, 2 bytes	✖	✔
mqtt.len	Message length mqtt	numerical, 8 bytes	✔	✔
mqtt.msg	Topic content	nominal-categorical	✔	✔
mqtt.msgid	Message ID	numerical, 2 bytes	✖	✖
mqtt.msgtype	Message type mqtt	numerical, 1 byte	✔	✔
mqtt.passwd	Password	nominal-categorical	✖	✖
mqtt.qos	QoS level	numerical, 1 byte	✖	✔
mqtt.retain	If the message is retained for a period	boolean	✔	✔
mqtt.topic	Topic name	nominal-categorical	✖	✔
mqtt.topic_len	Topic content size	numerical, 2 bytes	✖	✔
mqtt.username	Username	nominal-categorical	✖	✖
mqtt.willmsg	Retained topic content	nominal-categorical	✖	✖
mqtt.willtopic	Topic name retained	nominal-categorical	✖	✖
label	class	nominal-categorical	✔	✔

So, we have saved each attack in a separate excel file to use it in the next step. The MTM dataset contains 336,623 samples and 11 features with labels that classify them as normal samples or MTM samples, as shown in Figure 2. While the DoS dataset contains 643,722 samples and 23 features with labels that classify them as normal samples or DoS samples, as shown in Figure 3.

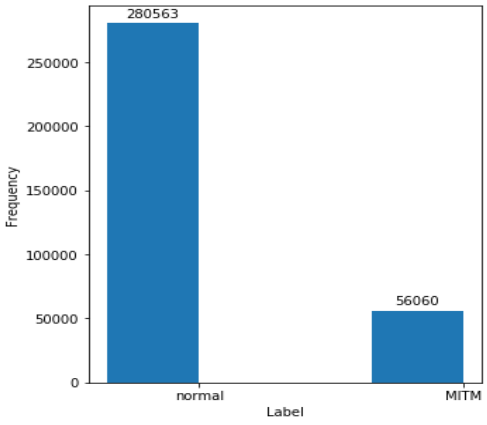

Figure 2. MTM dataset

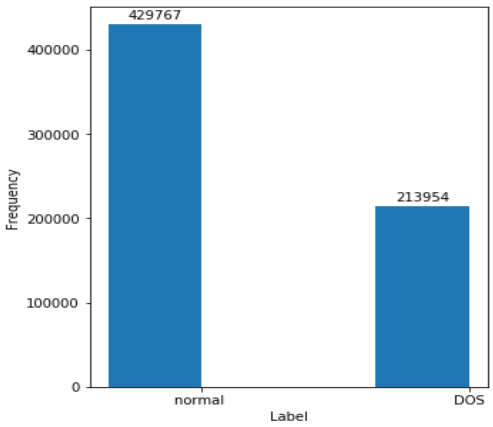

Figure 3. DoS dataset

2.2. Pre-processing dataset

To make the dataset more readable, understandable, and to not contain any null values, we applied two steps. In the first one, we fill missing values in the dataset of the mean value for certain columns. The second is that we use the label encoder method to convert the data types for specific columns to numeric data types because machine learning algorithms can only deal with numeric data types. Now the dataset is ready to be used as an input to the algorithms.

2.3. Machine learning models

After pre-processing the dataset, it is ready to be fitted into machine learning algorithms for prediction and detection purposes. Then, we used the Hold out method to divide the dataset into training dataset (training the model) that is 0.70 of all dataset and testing dataset (assessing the algorithms' performance) that is 0.30 of all datasets. The number of samples in all datasets, training and testing is shown in Table 3. So, we used four famous algorithms used in IoT attack detection: eXtreme gradient boosting (XGBoost), RF, DT, and GB. In the following subsection, we will present an overview of these algorithms and what the parameters are that they use.

Table 3. Number of samples in both datasets

Dataset	All dataset	Training	Testing
MTM	336,623	235,636	100,987
DoS	643,722	450,605	193,117

2.4. Gradient boosting

The GB is an ensemble algorithm that was developed to solve classification and regression tasks [21], [22]. It merges several weak learners into a single strong learner. These are GB-DTs, in which each tree is run separately, producing independent forecasts, which are then combined to make a final model's prediction. The number of weak learners is determined as number of estimators parameter [21], [22]. The model's prediction is integrated in classification problems like detecting the MTM attack or DoS by selecting the class label (MTM, normal in MTM dataset or DoS, normal in DoS dataset) with the most votes from all trees [21], [22]. In our experiment for both datasets, we used the following parameters of GB: i) n_estimators=100; ii) learning_rate=0.1; iii) max_depth=3; and iv) random_state=42.

2.5. eXtreme gradient boosting

The XGBoost is an ensemble model built to solve classification and regression problems [23], [24]. It combines a number of weak learners into a single strong learner. These are GB-DTs, in which each tree is run separately, producing independent forecasts, which are then combined to make a final model's prediction. Unlike GB, it uses the gradient descent technique to reduce the difference between actual and anticipated results, improving speed and performance [23], [24]. In classification tasks such as predicting or detecting an MTM attack or a DoS, the model's predictions are merged by choosing the class label (MTM, normal in the MTM dataset or DoS, normal in the DoS dataset) with the most votes from all trees [23], [24]. We utilized the

following XGBoost parameters in our experiment for both datasets: i) n estimators=100; ii) colsample bytree=1; iii) max depth=10; iv) and subsample=1.

2.6. Random forest

The RF is a supervised ensemble model comprised of many DTs created for regression and classification tasks, each of which is carried out by a single individual and yields a prediction [25], [26]. Then, in classification problems, the class with the most votes become the model's forecast like predict or detect the MTM attack or DoS, while in regression tasks, the model's prediction is computed as the average of all trees' predictions (MTM, normal in MTM dataset or DoS, normal in DoS dataset), because the label in classification is discrete while the label is continuous in regression task. The number of estimators supplied as a parameter in the RF model determines the number of trees [25], [26]. In our experiment for both datasets, we used the following parameters of RF: i) n_estimators=500; ii) max_features=log2; and iii) random_state=42.

2.7. Decision tree

The DT is a member of the supervised learning algorithm family, which the algorithm developed based on a training dataset that has a class label [23], [27]. By generating a tree in order to forecast the value, DT is used to solve numerous classification and regression challenges. This tree contains many parts: root node, splitting criteria (i.e., entropy, information gain, gini index, gain ratio, reduction in variance and chi-square), internal node, and leaf node (act a target value) [23], [27]. The tree is splitting the input dataset or training dataset (MTM dataset and DoS dataset), constituting a root node and children's nodes. This process is still in each child until the tree finishes all the samples in the training dataset [23], [27]. We used the following DT parameters in our experiment for both datasets/criterion: gini, min_samples_split=2, and random_state=42.

3. RESULTS AND DISCUSSION

For the machine learning algorithms based on the evaluation metrics, we present the evaluation metrics utilized and the outcomes in both datasets in this section. The Anaconda tool and the Python programming language are used for all experimental outcomes. Four well-known classification metrics—accuracy, precision, recall, and F1-score-were utilized in our studies to assess the effectiveness of machine learning algorithms [28], [29]. That accuracy is determined by dividing the total number of guesses by the number of right forecasts (1):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

precision is a measure of a positive example's likelihood of being truely positive, as indicated in (2):

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

as indicated in (3), recall estimates the chance of actual positives being accurately classified as positive.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1-score: as indicated in (4), this is the weighted mean of precision and recall, which includes both erroneous positives and false negatives.

$$F1 - score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (4)$$

In our experiments, we applied four machine algorithms on the MTM dataset and DoS dataset with specific parameters as we mentioned in the previous section, and the performance results as shown in. Table 4 shows the results of aforementioned algorithms in terms of four classification metrics. All the algorithms have almost the same results in all metrics, which means that these algorithms can detect the MTM attack.

Table 4. MTM dataset results

Algorithm	Accuracy	Precision	Recall	F1-score
XGB	99.9	99.9	99.9	99.9
RF	99.8	99.6	99.9	99.7
DT	99.9	99.9	99.9	99.9
GB	99.9	99.6	99.9	99.8

Table 5 shows the performance results of four algorithms in terms of four classification metrics. The XGB has slightly better results compared with the other algorithms, which means that the XGB algorithm has a better ability to detect DoS attacks compared with the others. Finally, the aforementioned machine learning techniques are quite good at detecting MTM and DoS assaults, motivating us to deploy them to protect devices from these types of attacks.

Table 5. DoS dataset results

Algorithm	Accuracy	Precision	Recall	F1-score
XGB	97.8	98.0	97.0	97.5
RF	97.2	97.0	97.0	97.0
DT	97.2	96.8	96.8	96.8
GB	97.6	97.7	96.8	97.2

4. CONCLUSION

In this paper, we develop four machine learning algorithms to detect two well-known attacks that attack the connected devices in any network by obtaining related datasets from the Kaggle website. The algorithms that are used are: XGBoost, RF, decision tree, and GB. We then used four classification metrics to assess these algorithms: precision, accuracy, f1-score, and recall. We achieved the following results: i) all algorithms detect the MTM attack with a performance greater than 99% in all metrics and ii) all algorithms can detect a DoS attack with a performance greater than 97% in all metrics. So, these four algorithms can be relied on to detect MTM and DoS attacks very well for both datasets, prompting us to use their effectiveness in protecting devices from these attacks. In future work, we plan to collect datasets related to other attacks and use another machine learning algorithms. In addition, we will also apply deep learning algorithms, pre-trained models, and all state-of-the-art models to future datasets.

ACKNOWLEDGEMENTS

The authors are grateful to the Iraqi Ministry of Higher Education and Scientific Research (MOHESR) for technically supporting the current research.

REFERENCES

[1] I. Al-Barazanchi, Z. A. Jaaz, H. H. Abbas, and H. R. Abdulshaheed, "Practical application of IoT and its implications on the existing software," in *2020 7th International Conference on Electrical Engineering, Computer Sciences and Informatics (EECSI)*, Oct. 2020, pp. 10–14, doi: 10.23919/EECSI50503.2020.9251302.

[2] H. R. Abdulshaheed, Z. T. Yaseen, A. M. Salman, and I. Al_Barazanchi, "A survey on the use of WiMAX and Wi-Fi on vehicular Ad-Hoc networks (VANETs)," *IOP Conference Series: Materials Science and Engineering*, vol. 870, no. 1, p. 012122, Jun. 2020, doi: 10.1088/1757-899X/870/1/012122.

[3] I. Al_Barazanchi, S. A. Shawkat, M. H. Hameed, and K. S. L. Al-badri, "Modified RSA-based algorithm: a double secure approach," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 6, pp. 2818–2825, Dec. 2019, doi: 10.12928/telkomnika.v17i6.13201.

[4] M. Jovic, E. Tijan, S. Aksentijevic, and D. Cisic, "An overview of security challenges of seaport IoT systems," in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2019, pp. 1349–1354, doi: 10.23919/MIPRO.2019.8757206.

[5] S. Q. Salih *et al*., "Integrative stochastic model standardization with genetic algorithm for rainfall pattern forecasting in tropical and semi-arid environments," *Hydrological Sciences Journal*, vol. 65, no. 7, pp. 1145–1157, May 2020, doi: 10.1080/02626667.2020.1734813.

[6] D.-Z. Sun, Y. Mu, and W. Susilo, "Man-in-the-middle attacks on Secure Simple Pairing in Bluetooth standard V5.0 and its countermeasure," *Personal and Ubiquitous Computing*, vol. 22, no. 1, pp. 55–67, Feb. 2018, doi: 10.1007/s00779-017-1081-6.

[7] R. A. Karthika and M. Maheswari, "Detection analysis of malicious cyber attacks using machine learning algorithms," Jun. 2022, doi: 10.1016/j.matpr.2022.05.510.

[8] K. B. Dasari and N. Devarakonda, "Detection of different DDoS attacks using machine learning classification algorithms," *Ingénierie des systèmes d information*, vol. 26, no. 5, pp. 461–468, Oct. 2021, doi: 10.18280/isi.260505.

[9] J. L. Lee and C. S. Hong, "Nonparametric detection methods against DDoS attack," *Korean Journal of Applied Statistics*, vol. 26, no. 2, pp. 291–305, Apr. 2013, doi: 10.5351/KJAS.2013.26.2.291.

[10] M. Mittal, V. Kadyan, D. Kumar, and V. Kukreja, "Detection of DoS attacks using machine learning techniques," *International Journal of Vehicle Autonomous Systems*, vol. 15, no. 3–4, pp. 256–270, 2020, doi: 10.1504/IJVAS.2020.10039658.

[11] D. Rathee and S. Mann, "Detection of E-Mail phishing attacks–using machine learning and deep learning," *International Journal of Computer Applications*, vol. 183, no. 47, pp. 1–7, Jan. 2022, doi: 10.5120/ijca2022921868.

[12] R. J. Alzahrani and A. Alzahrani, "Security analysis of DDOS attacks using machine learning algorithms in networks traffic," *Electronics*, vol. 10, no. 23, pp. 1–15, Nov. 2021, doi: 10.3390/electronics10232919.

[13] M. Agarwal, D. Pasumarthi, S. Biswas, and S. Nandi, "Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization," *International Journal of Machine Learning and Cybernetics*, vol. 7, no. 6, pp. 1035–1051, Dec. 2016, doi: 10.1007/s13042-014-0309-2.

[14] A. Scirè, F. Tropeano, A. Anagnostopoulos, and I. Chatzigiannakis, "Fog-computing-based heartbeat detection and arrhythmia classification using machine learning," *Algorithms*, vol. 12, no. 2, pp. 1–21, Feb. 2019, doi: 10.3390/a12020032.
[15] K. Ganesan, "Machine learning data detection poisoning attacks using resource schemes multi-linear regression," *Neural, Parallel, & Scientific Computations*, vol. 28, no. 2, pp. 73–82, Jun. 2020, doi: 10.46719/npsc20202821.
[16] Z. Wu, M. Yue, D. Li, and K. Xie, "SEDP-based detection of low-rate DoS attacks," *International Journal of Communication Systems*, vol. 28, no. 11, pp. 1772–1788, Jul. 2015, doi: 10.1002/dac.2783.
[17] S. B. R. Jones and N. Kumar, "Unraveling the security pitfalls that stem from core cloud benefits through analyzing various DoS attacks, detection and prevention," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 11, no. 9, pp. 541–553, Sep. 2019, doi: 10.5373/JARDCS/V11/20192603.
[18] M. Benter and P. Kuhlang, "MTM-HWD – integration of ergonomic evaluation into production planning," *ASU Arbeitsmedizin Sozialmedizin Umweltmedizin*, vol. 2021, no. 11, pp. 703–706, Nov. 2021, doi: 10.17147/asu-2111-9792.
[19] M. Yu, "An adaptive method for source-end detection of pulsing DoS attacks," *International Journal of Security and Its Applications*, vol. 7, no. 5, pp. 279–288, Sep. 2013, doi: 10.14257/ijsia.2013.7.5.26.
[20] S. Banerjee and P. S. Chakraborty, "Proposed approach to detect distributed denial of service attacks in software defined network using machine learning algorithms," *International Journal of Engineering & Technology*, vol. 7, no. 2.8, p. 472, Mar. 2018, doi: 10.14419/ijet.v7i2.8.10488.
[21] Y. K. Salih, O. H. See, S. Yussof, A. Iqbal, and S. Q. M. Salih, "A proactive fuzzy-guided link labeling algorithm based on MIH framework in heterogeneous wireless networks," *Wireless Personal Communications*, vol. 75, no. 4, pp. 2495–2511, Apr. 2014, doi: 10.1007/s11277-013-1479-z.
[22] Y. Mirsky, N. Kalbo, Y. Elovici, and A. Shabtai, "Vesper: Using echo analysis to detect man-in-the-middle attacks in LANs," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1638–1653, Jun. 2019, doi: 10.1109/TIFS.2018.2883177.
[23] Z. Li, G. Hu, D. Yang, and X. Yao, "Global abnormal correlation analysis method for DDoS attack detection," *Journal of Computer Applications*, vol. 29, no. 11, pp. 2952–2956, Dec. 2009, doi: 10.3724/SP.J.1087.2009.02952.
[24] X. Gu, H. Wang, T. Ni, and H. Ding, "Detection of application-layer DDoS attack based on time series analysis," *Journal of Computer Applications*, vol. 33, no. 8, pp. 2228–2231, Nov. 2013, doi: 10.3724/SP.J.1087.2013.02228.
[25] S. Balaji and R. Seshadri, "Attack prevention and attack detection strategies by comparing different DDos models," *International Journal of Computer Applications*, vol. 129, no. 14, pp. 24–27, Nov. 2015, doi: 10.5120/ijca2015907094.
[26] H. Rahmani, N. Sahli, and F. Kamoun, "Distributed denial-of-service attack detection scheme-based joint-entropy," *Security and Communication Networks*, vol. 5, no. 9, pp. 1049–1061, Sep. 2012, doi: 10.1002/sec.392.
[27] B. Kashyap and S. K. Jena, "DDoS attack detection and attacker identification," *International Journal of Computer Applications*, vol. 42, no. 1, pp. 27–33, Mar. 2012, doi: 10.5120/5657-7549.
[28] Y. Zhou, C. Jiao, H. Chen, L. Ma, and G. Hu, "Traffic behavior feature based DoS&DDoS attack detection and abnormal flow identification for backbone networks," *Journal of Computer Applications*, vol. 33, no. 10, pp. 2838–2841, Nov. 2013, doi: 10.3724/SP.J.1087.2013.02838.
[29] P. S. Saini, S. Behal, and S. Bhatia, "Detection of DDoS attacks using Machine learning algorithms," in *2020 7th International Conference on Computing for Sustainable Global Development (INDIACom)*, Mar. 2020, pp. 16–21, doi: 10.23919/INDIACom49435.2020.9083716.

BIOGRAPHIES OF AUTHORS

Sura Abdulmunem Mohammed Al-Juboori Received the B.Sc. AL-Mustansirya University–Iraq in 2004, and M.Sc. degree in computer science from Middle East University-Jordan in 2019. Working as Assistant Lecturer in the Ministry of the Higher Education in Iraq. Her research interest in security and cryptography, artificial intelligence, and cloud computing. She can be contacted at email: sura.sultan.ss@gmail.com.

Dr. Firas Hazzaa received his PhD in Cyber security from Anglia Ruskin university in Cambridge/United Kingdom in 2019. Working as lecturer in the ministry of the higher education in Iraq. He is associated fellow of the UK higher education Academy. His research interest in IoT security and cryptography. He has many published papers in this field and collaborate with different researchers and industry in cyber security to develop new ideas and research. He can be contacted at email: fih7600@gmail.com.

Zinah Sattar Jabbar received her master's degree in information technology from Middle East university, Jordan in 2012. CurrentlyJabbar receivedturer at Imam Ja'afar Al-Sadiq university. Her research interest in computer security. Her research interests include data and information quality, social media analytics, information system, data analysis, cloud computing and machine learning and AI. She can be contacted at email: sattarzeina@gmail.com.

Sinan Salih received the B.Sc. degree in information systems from the University of Anbar, Al-Anbar, Iraq, in 2010, the M.Sc. degree in computer sciences from Universiti Tenaga National (UNITEN), Malaysia, in 2012, and the Ph.D. degree in soft modeling and intelligent systems from Universiti Malaysia Pahang (UMP). His current research interests include optimization algorithms, nature-inspired metaheuristics, machine learning, and feature selection problem for real world problems. He can be contacted at email: sinan.salih@duc.edu.iq.

Hassan Muwafaq Gheni received his Bachelor (B.Sc) of Electrical and Electronic Engineering from Department of Electrical Engineering, Babylon university-Iraq-hilla in June 2016. In February 2018, he entered the master's program at the Faculty of Electrical and Electronic Engineer, Universiti Tun Hussein Malaysia. He is a lecturer at Al-Mustaqbal University College/Department of Computer Techniques Engineering. His research interest is optical communication, IoT, wireless sensor network, communications, V2V system, and artificial intelligent. He can be contacted at email: hasan.muwafaq@mustaqbal-college.edu.iq.