Detection of DoS Attack using AdaBoost Algorithm on IoT System

Salman Rachmadi
School of Computing
Telkom University
Bandung, Indonesia
salmanrachmadi@telkomuniversity.ac.id

Satria Mandala
Human Centric(HUMIC) Engineering
School of Computing
Telkom University
Bandung, Indonesia
satriamandala@telkomuniversity.ac.id

Dita Oktaria
Human Centric(HUMIC) Engineering
School of Computing
Telkom University
Bandung, Indonesia
dioktaria@telomuniversity.ac.id

2021 International Conference on Data Science and Its Applications (ICoDSA) | 978-1-6654-4303-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICODSA53588.2021.9617545

Abstract—**Internet of Things (IoT) is a networking concept where an object can transmit data over the internet without any human interaction. The object is usually a sensor with a communication device connected to the Internet. The popularity of IoT is increasing with the advent of 5G technology. However, the threats to the system are also getting more intense. One of the serious threats to IoT systems is known as denial of service (DoS) attack, which usually target broker services on that system. Several researches have been performed to overcome this DoS attack. However, the results appear to be ineffective. It can be seen that the accuracy of the DoS detection systems are still low. This study aims to provide a solution to the above problems by proposing an Intrusion Detection System based on Artificial Intelligence (AI, AdaBoost) for IoT system. The method used in this study is supervised learning which measures the accuracy of predictions in detecting DoS on IoT network data. The experiments have been carried out on 130223 DoS attack data and 130284 normal data. The detection accuracy of the DoS detection is 95.84% and the F1-Score is 95.72%. Recall and precision have achieved 93.28% and 98.29%, respectively.**

Keywords— DoS detection, AdaBoost, IoT

I. Introduction

According Holst A. [1] The number of IoT device will be increase to almost triple from 8.74 billion in 2020 to more 25.4 billion IoT devices 2030. IoT devices are used in all types of industry consumer markets, with the number segment accounting for around 60 percent of all IoT connected devices in 2020. As the number of IoT devices increases, the threats to the security of these systems also increase, such as in the Mirai Botnet attack [2]. The attack exploited the security weaknesses of existing IoT systems by launching a distributed denial of service (DDoS) attack against more than 400,000 IoT devices.

There are several communication protocols in IoT system. Three protocols that are frequent used in IoT systems are: Message Queue Telemetery Transport (MQTT), Constrained Application Protocol (CoAP), Extensible Messaging, and Presence Protocol (XMPP) [3]. MQTT is one of the IoT protocols that uses the concept of publish and subscribe to save device resources. Publishers are normally sensors that publish certain topics on brokers. While subscribers are end users who utilize data according to topics that have been published on the broker. The main challenge in the MQTT system is the security of the portocol which is still easily exploited by attackers [4]. On IoT systems, attackers can sniff, modify packet data toa ttack data privacy, or even shut down brokerage services [4].

Wong H. [4] has conducted a study on the Man in The Middle attack by changing the message on MQTT so that it becomes a malicious message. Andy S. [5] have done an analysis of the security MQTT protocol, they explain the publish and subscribe pattern making it easy for attackers to retrieve data and also carry out DoS attacks. Paudel R [6], shown several machine learning algorithms still have low accuracy to detecting attacks.

This study aims to develop an IDS on an IoT system using the MQTT protocol where DoS attacks are detected using an ensemble learning algorithm. The ensemble learning-based model enables accurate DoS detection results on network traffic data obtained from research conducted by [7].

The main contribution of this paper are as follows:

- Development of an ensemble learning-based model (AdaBoost) for DoS detection on an IoT system
- Optimization of the ensemble learning model developed by tuning the AdaBoost parameter to improve detection accuracy.

The rest of this paper is organized as follows. Part II high lights the facts and theory of DoS attacks and machine learning detection. And the design of the system model dataset, as well as the selection of features used in this paper. Section IV presents the results and analysis of machine learning models for DoS attack detection. Finally, Part V draws conclusions andout lines for future work.

II. Related Work

Intrusion Detection is very popular in research of network analytics of the past few years. Cause IoT system has connected in internet it has risk threat of cyber attack to loss the system. Syarif and Gata (2017) [8] research about intrusion detection system using combine K-NN algorithm with Particle Swarm Optimization (PSO), their use KDDCUP99 and the result is optimize 2% then usual K-NN algorithm. Another reseacher Wang, Cao, and Hong (2020) [9] work about IDS using Convolutional Neural Network (CNN), they using NSL-KDD dataset for classification model. The result of this model IDS-CNN can efficient to detect intrusion for network data stream and also its detection precision is better than the state-of-the-art method. However, not many people do research on IDS in IoT system. Alaiz Moreton, et al. (2019) [10] research about multi classification IDS in IoT system based MQTT-protocols, they using ensemble method and deep learning. The result XGBoost has highest accuracy above deep learning LSTM.

A. Internet of Things

Internet of Things is combination object physics, sensor, aquator, and controller, who connected to internet. Low cost and easy developing mobile device, it would be new style technolgy in new era. Howover as soon as development, security problem increase because device connected to internet, thats mean possibility attacker to attack the system [11]. Security on IoT protocols, especially MQTT still needs

978-1-6654-4303-6/21/$31.00 ©2021 IEEE

to be developed, and integration is needed for every development [4].

B. Message Queue Telemtery Transport (MQTT)

MQTT one of most lightweight protocol communication for IoT. It works publish and subscribe for ensure communication across platform. Low complexity, low power, and low computational is become compatible for IoT network communication. A central server as broker, receive message from publisher and forward the message to the subscribers based on their topics. MQTT requires a TCP/IP connection; however, the implementation in a sensor network is difficult due to the limited resources of the nodes on the network [12].

C. Denial of Service (DoS)

DoS attack are carried out by by malicious attackers to consume resources or bandwidth for legitimate users. Various compromised node when penetrated by several type of attacks is called DDoS. The most common DoS attack flooding of huge amount data of traffic to network resource will consume, bandwidth, CPU time, etc. Several types of DoS attacks are SYN flood, DNS flood, Ping flood, UDP flood, TCP flood, ICMP broadcast etc[13].

D. Ensemble Method

In general, supervised learning algorithms give solution to find model on hypothesis space. Right model maybe exist, it possible be hard to find one. Ensmble method combine multi-learner to find best model for best hypothesis alone. Ensmble method use several weak learner to become strong learner [14]. Boosting is one of ensemble method to train several weak learner algorithm and the result based on weight combination [15]. Ivan Vaccari (2020) [7], work with machine learning techniques that used three ensemble learning to detect multi classification attack, on mqtt dataset. Result of the research is Random Forest has highest accuracy more than Gradient Boosting and Multilayer Preceptron.

III. Methodology

This research is conduct to detecting attack using AdaBoost algorithm, implement the models to simulation acitvity, and compared three ensemble algorithm to the model of AdaBoost. The main stage is carried out in this study can be seen in Fig. 1.

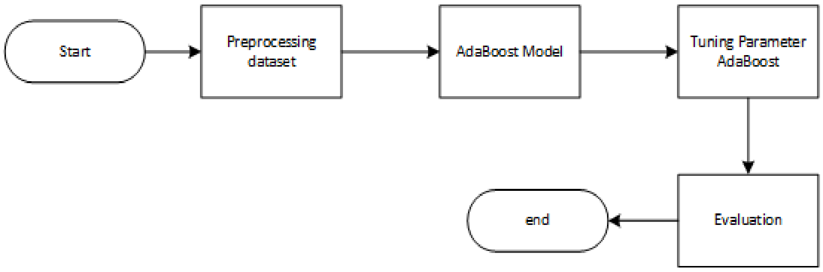

Fig. 1. Research methodology.

A. System Architecture

The architecture of the IoT network and IDS we model in this work is as shown in Fig. 2. The IoT system has sensor node publish message with some topic to the MQTT-broker and some device will be subscribe the topic and get the read the message. Some attacker flood publish message to the broker as a target using mqtt-malaria tools [16].

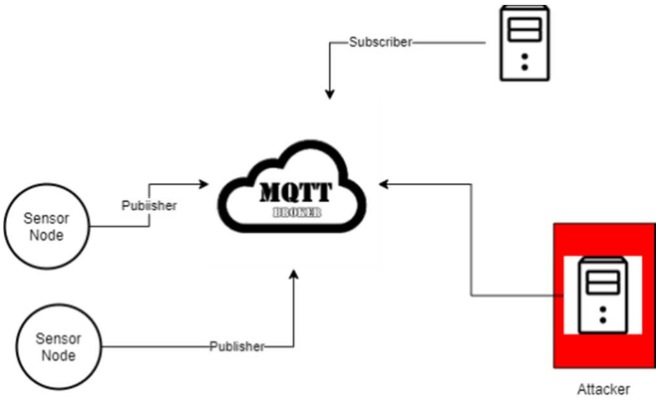

Fig. 2. Network architecture.

B. Data

In this work we used dataset has been provided by previous research using MQTTset [7]. This dataset purpose to conduct a model detection DoS attack on protocol MQTT.

1) *Dataset from MQTTset*[7]*:* This dataset have many attack scenarios, but in this work we use one most populars attack, just DoS attack. Distribution of this dataset is on Table I. As we know from dataset MQTTset, normal traffic is more then 11 million traffic, but we just used 130284 traffic normal for balanced the dataset.

TABLE I. Table Traffic MQTTset[7] Distribution

Traffic	Sample Size
DoS Attack	130223
Normal	130284

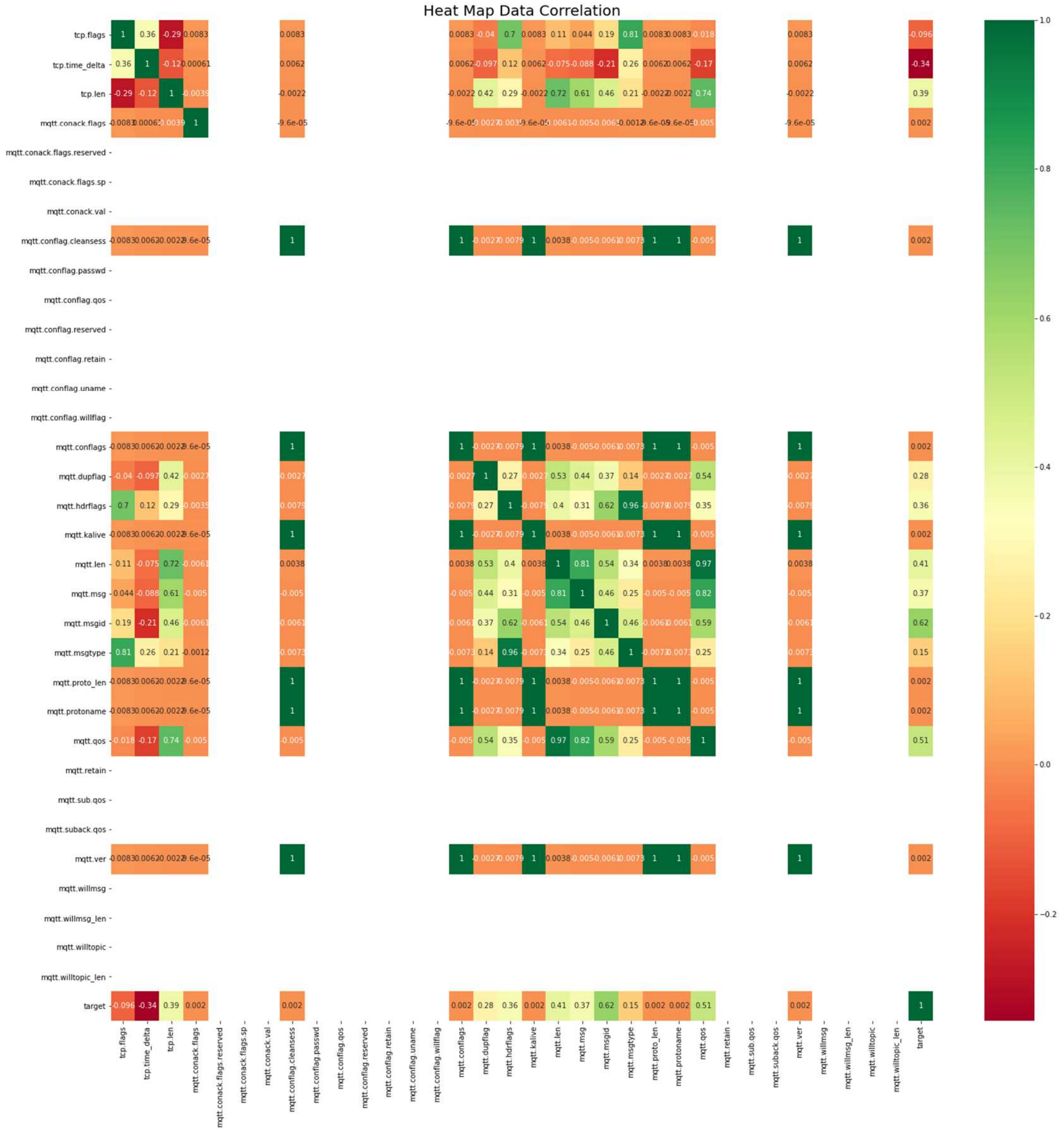

Fig. 3. Dataset[7] correlation.

C. Feature Selection

We exploration the MQTTset [7] dataset, these dataset have 33 feature have been extracted and this feature for MQTT behavior. Further we show the correlation dataset on Fig. 3. to analyze relationship between a pair of variables. After we shown data correlation we try to looking for feature importance from this dataset. AdaBoost Classifier have been created to show feature importance of this model on Fig. 4.

After we analyzed data correaltion and feature importance we used 33 feature by the model from this dataset. And this decision on Table II. based on MQTT behavior have been explaned by Ivan Vaccari(2020) [7].

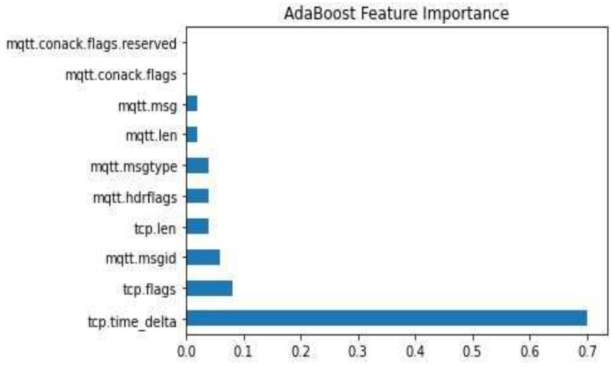

Fig. 4. Feature importance.

TABLE II. FEATURE TABLE

No	Feature Name	Porotocol	Description
1	tcp.flags	TCP	TCP Flags
2	tcp.len	TCP	TCP Segment Length
3	tcp.time_delta	TCP	Time TCP Stream
4	mqtt.conack.flags	MQTT	Acknowladge Flags
5	mqtt.conack.flags.reserved	MQTT	Reserved
6	mqtt.conack.flags.sp	MQTT	Session Present
7	mqtt.conack.val	MQTT	Return Code
8	mqtt.conflag.cleansess	MQTT	Clean Session Flags
9	mqtt.conflag.passwd	MQTT	Password Flags
10	mqtt.conflag.qos	MQTT	QoS Level
11	mqtt.conflag.reserved	MQTT	(Reserved)
12	mqtt.conflag.retain	MQTT	Will Retain
13	mqtt.conflag.uname	MQTT	User Name Flag
14	mqtt.conflag.willflag	MQTT	Will Flag
15	mqtt.conflags	MQTT	Connect Flags
16	mqtt.dupflags	MQTT	DUP Flags
17	mqtt.hdrflags	MQTT	Header Flags
18	mqtt.kalive	MQTT	Keep Alive
19	mqtt.len	MQTT	Message Length
20	mqtt.msg	MQTT	Message
21	mqtt.msgid	MQTT	Message Identifier
22	mqtt.msgtype	MQTT	Message Type
23	mqtt.protolen	MQTT	Proto Name Length
24	mqtt.protoname	MQTT	Proto Name
25	mqtt.qos	MQTT	QoS Level
26	mqtt.retain	MQTT	Retain
27	mqtt.sub.qos	MQTT	Requested QoS
28	mqtt.suback.qos	MQTT	Granted QoS
29	mqtt.ver	MQTT	Version
30	mqtt.willmsg	MQTT	Will Message
31	mqtt.willmsglen	MQTT	Will Msg Length
32	mqtt.willtopic	MQTT	Will Topic
33	mqtt.willtopiclen	MQTT	Will Topic Length

Algorithm 1 AdaBoost Algorithm[17]

```
1) Initialize the observation weights $W_i$ =1/N , i= 1,2,3.. ,N
2) For m = 1 to M:
    a) Fit a classifier $G_m(x)$ to the training data using wieght $w_i$
    b) Compute
        $err_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}$
    c) Compute $\alpha_m = log(\frac{(1-err_m)}{err_m})$
    d) Set $w_i \leftarrow w_i . exp[\alpha_m . I(y_i \neq G_m(x_i))], i = 1,2,3, \ldots, N$.
3) Output $G(x) = [\sum_{m=1}^{M} \alpha_m G_m(x)]$.
```

D. AdaBoost Classification

AdaBoost Algorithm 1 selected weak learner or weak classifer to do several iteration which will be combined with maximum average, and the result will be strong learner for the data.

E. Preprocessing Dataset

Data was preprocessed to clean up and transform it into numerical and categorical type. This process goes through several stages:

(1) Fill NaN data with 0
(2) Transform data to kategorical
(3) Split dataset into 80% data train, and 20% data test

F. Evaluation

Evaluation is used to measure the preformance of system or algorthm. In this research we used confusion matrix to visualizing the preformance of the system. Result of model classification shown in each entry in confusion matrix. Our AdaBoost model have been implement in binary classification, there is positive and negative class. In this research we used accuracy, precision, and recall as performance measure.

TABLE III. TABLE CONFUSION MATRIX

Confusion Matrix		Actual Class	
		1	0
Predicted Class	1	TP	FP
	0	FN	TN

True Positive (TP) is data with positive actual value and positive predictive value. False Positive (FP) is data with positive predictive value and actual negative value. False Negative (FN) is data with the negative predictive value and the actual value is positive. True Negative (TN) is data with negative predictive value and actual negative value.

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

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

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

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

IV. EXPERIMENT AND RESULT

In this research, we test the system on Intel Core i3-8100 @2.50GHz processor with 8GB RAM. We used python programming by as we know machine learning and artificial intelligence libraries and tools such as Sklearn[18]. After preprocessing data, we divided the dataset and build model using AdaBoost Classifier with default parameter. The result of this model is on Table IV. The result we presented based on confusion matrix from the Table V for validation model of AdaBoost model, we use Cross-Validation with K=5. Results of this validation have mean score achieve 95.77%.

TABLE IV. NORMAL ADABOOST ALGORITHM

AdaBoost	Accuracy	Recall	Precision	F1-Score
Result	95.13%	90.97%	99.14%	94.88%

TABLE V. CONFUSION MATRIX ADABOOST DETECTION

Confusion Matrix		Actual Class	
		1	0
Predicted Class	1	24243 (TP)	421 (FP)
	0	1745 (FN)	25693 (TN)

TABLE VI. MEAN SCORE GRIDSEARCHCV BY LEARNING RATE AND NUMBER OF TREE

Score (Mean)	Learning Rate	Number of Tree
90.91%	0.1	20
90.90%	0.1	40
95.03%	0.1	60
95.03%	0.1	80
95.03%	0.1	100
95.01%	0.3	20
95.03%	0.3	40
95.05%	0.3	60
95.10%	0.3	80
95.10%	0.3	100
95.01%	0.5	20
95.07%	0.5	40
95.75%	0.5	60
95.76%	0.5	80
95.76%	0.5	100
94.98%	0.7	20
95.60%	0.7	40
95.75%	0.7	60
95.75%	0.7	80
95.75%	0.7	100
94.97%	0.9	20
95.76%	0.9	40
95.76%	0.9	60
95.77%	**0.9**	**80**
95.77%	0.9	100
95.33%	1.0	20
95.76%	1.0	40
95.77%	1.0	60
95.76%	1.0	80
95.76%	1.0	100

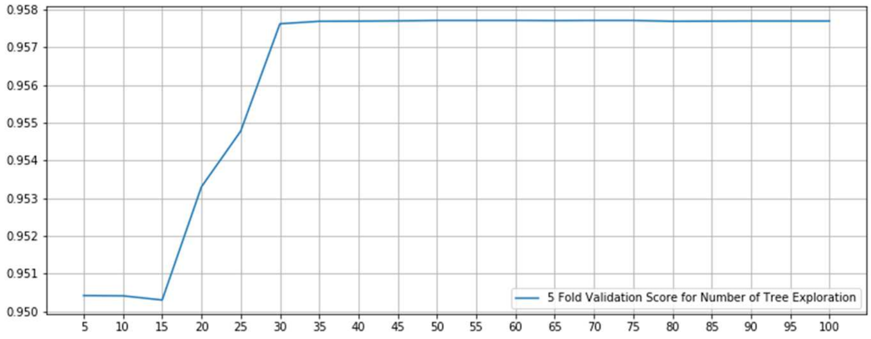

Fig. 5. 5-Fold cross validation score AdaBoost for tree exploration.

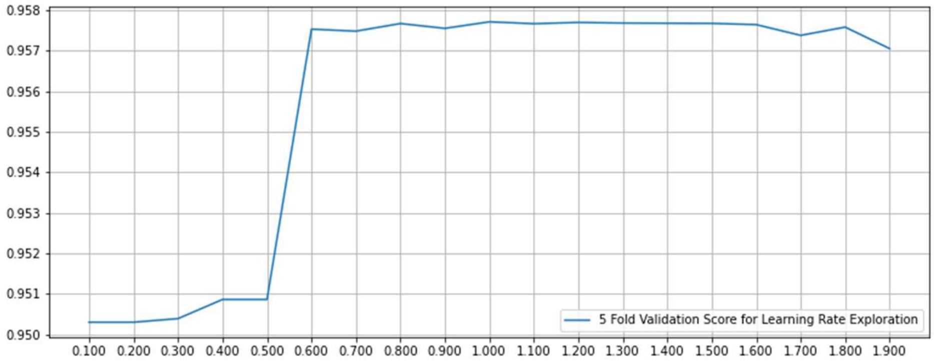

Fig. 6. 5-Fold cross validation score AdaBoost for learning rate exploration.

Further we explore learning rate and number of tree of AdaBoost model for validation score, the result has been shown in Fig. 5 and Fig. 6. This exploration puposed to analysis paramater of learning rate and number of tree from AdaBoost algorithm, to make sure get the best model of detection DoS using AdaBoost algorithm.

From Fig. 5, it can see by the increase number of tree will be increase validation score of detection, and from Fig. 6 by increase learning rate some times increase or decrease validation score. After exploration tree and learning rate, we looking for best model by combine learning rate and number of tree using GridSearch Cross Validation with 5-Fold. From on Table VI. show the score comparison of the combination hyperparameter tuning based learning rate and number of tree. Which is useful to determining the best validation classification model. Hyperparameter tuning has been proven to increase score validation detection model. Result for best model has shown on Table VII.

TABLE VII. TABLE RESULT TUNING ADABOOST ALGORITHM

AdaBoost	**Accuracy**	**Recall**	**Precision**	**F1-Score**
Result	95.84%	93.28%	98.29%	95.72%

V. CONCLUSION

In this paper, we describe design and implementation of AdaBoost algorithm to detect DoS attack for IoT System, especially IoT system using MQTT protocol. We generated model from MQTTset[7] dataset and tuning this model to increase score for detection DoS in IoT sytem. We have considered tuning parameter for the best model detection of AdaBoost. It has been proven conducted tuning parameter can be increase score for detection. From MQTTset[7] the best model have parameter 0.9 learning rate and 80 number of tree. The result of this model have accuracy is 95.84%, F1 Score is 95.72%, recall is 93.28%, precision 98.29%. And for the validation score reached an average of 95.77% The experiment show ensemble learning based AdaBoost model with high performance for detection. In the future, we can improve another model of ensemble learning for IDS.

REFERENCES

[1] A. Holst, “IoT connected devices worldwide 2019-2030,” *Statista*. https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/.

[2] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, “DDoS in the IoT: Mirai and Other Botnets,” *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[3] B. H. Çorak, F. Y. Okay, M. Güzel, Ş. Murt, and S. Ozdemir, “Comparative Analysis of IoT Communication Protocols,” in *2018 International Symposium on Networks, Computers and Communications (ISNCC)*, Jun. 2018, pp. 1–6.

[4] H. T. Wong and T. Luo, “Man-in-the-Middle Attacks on MQTT-based IoT Using BERT Based Adversarial Message Generation,” 2020. https://www.semanticscholar.org/paper/Man-in-the-Middle-Attacks-on-MQTT-based-IoT-Using-Wong-Luo/113038b77bc5f2638cb7ca941e0c0186736f5e48.

[5] S. Andy, B. Rahardjo, and B. Hanindhito, “Attack scenarios and security analysis of MQTT communication protocol in IoT system,” presented at the 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), 2017. doi: 10.1109/EECSI.2017.8239179.

[6] R. Paudel, T. Muncy, and W. Eberle, “Detecting DoS Attack in Smart Home IoT Devices Using a Graph-Based Approach,” in *2019 IEEE International Conference on Big Data (Big Data)*, Dec. 2019, pp. 5249–5258.

[7] I. Vaccari, G. Chiola, M. Aiello, M. Mongelli, and E. Cambiaso, “MQTTset, a New Dataset for Machine Learning Techniques on MQTT,” *Sensors*, vol. 20, no. 22, p. 6578, Jan. 2020.

[8] A. R. Syarif and W. Gata, “Intrusion detection system using hybrid binary PSO and K-nearest neighborhood algorithm,” in *2017 11th International Conference on Information Communication Technology and System (ICTS)*, Oct. 2017, pp. 181–186.

[9] H. Wang, Z. Cao, and B. Hong, “A network intrusion detection system based on convolutional neural network,” *J. Intell. Fuzzy Syst.*, vol. 38, no. 6, pp. 7623–7637, Jan. 2020.

[10] H. Alaiz-Moreton, J. Aveleira-Mata, J. Ondicol-Garcia, A. L. Muñoz-Castañeda, I. García, and C. Benavides, “Multiclass Classification Procedure for Detecting Attacks on MQTT-IoT Protocol,” *Complexity*, vol. 2019, p. e6516253, Apr. 2019.

[11] S. Siboni *et al.*, “Security Testbed for Internet-of-Things Devices,” *IEEE Trans. Reliab.*, vol. 68, no. 1, pp. 23–44, Mar. 2019.

[12] O. Sadio, I. Ngom, and C. Lishou, “Lightweight Security Scheme for MQTT/MQTT-SN Protocol,” in *2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, Oct. 2019, pp. 119–123.

[13] K. Sonar and H. Upadhyay, “A Survey: DDOS Attack on Internet of Things,” vol. 10, no. 11, pp. 58–63, 2014.

[14] R. Polikar, “Ensemble based systems in decision making,” *IEEE Circuits Syst. Mag.*, vol. 6, no. 3, pp. 21–45, 2006.

[15] A. L. Buczak and E. Guven, “A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection,” *IEEE Commun. Surv. Tutor.*, vol. 18, no. 2, pp. 1153–1176, 2016.

[16] K. Palsson, “Attacking MQTT systems with Mosquittos (scalability and load testing utilities for MQTT environments),” 2019. https://github.com/etactica/mqtt-malaria.

[17] Y. Freund and R. E. Schapire, “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting,” *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[18] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *J. Mach. Learn. Res.*, vol. 12, no. 85, pp. 2825–2830, 2011.