UTM UNIVERSITI TEKNOLOGI MALAYSIA
INTERNATIONAL JOURNAL OF INNOVATIVE COMPUTING
ISSN 2180-4370
Journal Homepage : https://ijic.utm.my/

Comparing Malware Attack Detection using Machine Learning Techniques in IoT Network Traffic

Yee Zi Wei[1], Marina Md-Arshad[2*], Adlina Abdul Samad[3], Norafida Ithnin[4]
Faculty of Computing,
Universiti Teknologi Malaysia,
81310 UTM Johor Bahru, Malaysia
Email: ziweiyee@yahoo.com[1], marinama@utm.my[2*], adlina6@graduate.utm.my3, afida@utm.my[4]

Submitted: 13/9/2022. Revised edition: 28/2/2023. Accepted: 27/2/2023. Published online: 30/5/2023
DOI: https://doi.org/10.11113/ijic.v13n1.384

***Abstract*—Most IoT devices are designed and built for cheap and basic functions, therefore, the security aspects of these devices are not taken seriously. Yet, IoT devices tend to play an important role in this era, where the amount of IoT devices is predicted to exceed the number of traditional computing devices such as desktops and laptops. This causes more and more cybersecurity attacks to target IoT devices and malware attack is known to be the most common attack in IoT networks. However, most research only focuses on malware detection in traditional computing devices. The purpose of this research is to compare the performance of Random Forest and Naïve Bayes algorithm in terms of accuracy, precision, recall and F1-score in classifying the malware attack and benign traffic in IoT network traffic. Research is conducted with the Aposemat IoT-23 dataset, a labelled dataset that contains IoT malware infection traffic and IoT benign traffic. To determine the data in IoT network traffic packets that are useful for threat detection, a study is conducted and the threat data is cleaned up and prepared using RStudio and RapidMiner Studio. Random Forest and Naïve Bayes algorithm is used to train and classify the cleaned dataset. Random Forest can prevent the model from overfitting while Naïve Bayes requires less training time. Lastly, the accuracy, precision, recall and F1-score of the machine learning algorithms are compared and discussed. The research result displays the Random Forest as the best machine learning algorithm in classifying the malware attack traffic.**

***Keywords*—Machine Learning, IoT, Malware, Attack Detection, Naïve Bayes, Random Forest**

I. INTRODUCTION

Internet of Things (IoT) is a network of physical devices that are equipped with sensors to gather information, identifiers to distinguish the source of data, software to examine the data and can connect to the Internet [1]. In recent years, the adoption of the IoT is increasing steadily. Nevertheless, IoT attacks have also skyrocketed by 900% in 2019 [2]. The hackers are aiming at IoT devices because they are more vulnerable compared to regular computers. Unlike a regular computer, the IoT devices contain neither a firewall nor a virus scanner and up to 98% of all IoT device network traffic is not encrypted [3]. Besides, most of the IoT devices in the market are built with the same low security mechanisms. This is mainly because most IoT device producers are not equipped with IT security expertise.

Among all the attacks detected, malware attack is the most common attack in IoT networks, such as the Mirai malware, Stuxnet computer worm and Silex/Brickerbot [4]. Malware will distribute across the network and thus permit the attackers to execute malicious code on the network and run multiple new attacks. For the Mirai malware, an infected device will propagate the malware to other devices on the network. All the devices are then under the attacker's control. The attacker will then launch a DDoS attack with those devices to the target destination and take down the victim's server. A survey is conducted on the IoT Commercial Adoption in Canada and 40% of the respondents say that their organization is using IoT solutions, while 22% show that their company is planning to integrate IoT solutions [5]. Even the US military is utilizing IoT technology and data to transform their warfare. All these situations have clearly depicted the increasing malware attacks on IoT devices in today's world and the urge to enhance the security of IoT devices.

Machine learning is an area of computer science and artificial intelligence (AI) that utilises data and algorithms to replicate how humans learn and gradually improve their

accuracy. It is classified into four main types: supervised, unsupervised, reinforcement, and semi-supervised. The model evaluation demonstrates how the model functions in practice. The predictions are compared to the actual (real) classes using either classification or continuous techniques. For classification, it can be done by assessing the categorization model based on the number of correct/incorrect predictions. Confusion metrics, accuracy, precision, recall, and F1-Score are some of the evaluation methodologies.

Therefore, this research proposed to compare the accuracy, precision and recall of Random Forest and Naïve Bayes algorithm in classifying between malware attacks and benign traffic in IoT network traffic. The dataset used in this research is the IoT-23 dataset [6], published in January 2020. This dataset comprises 3 benign and 20 malware network traffic samples collected on IoT devices from 2018 and 2019. Mirai malware mentioned above is also one of the malwares captured in this dataset. The full IoT-23 dataset is too big for most of the machines to handle. Therefore, a few Zeek log files from the dataset are selected and a subset is generated from the original dataset. After that, dataset clean-up is performed to remove unnecessary data. Two different machine learning techniques Random Forest and Naïve Bayes are used to train the sample data. Lastly, the performance of both the machine learning algorithm is compared and the better algorithm for detecting and classifying malicious traffic is identified.

This paper is structured as follows: Section II represents the literature review and dataset in Section III. Framework for IoT Malware Detection are illustrated in Section IV. Besides, the result analysis was made in Section V. Limitations of the research also shown in Section VI, whereas the suggestion for the limitation also made in Section VII. Finally, the conclusion was made in the last section.

II. Literature Review

To build the basis for this research, a literature review is performed on current IoT devices and vulnerabilities, high profile cyberattacks in IoT and supervised machine learning algorithms to understand the vulnerabilities of the Internet of Things (IoT) and supervised machine learning algorithms.

A. Internet of Things (IoT) Background

According to the Cambridge dictionary, the IoT is defined as computing devices that are interconnected and able to interchange information [7]. Generally, IoT devices are made up of actuators, sensors and other programmable components, enabling them to converse with humans and other IoT devices over the Internet [8]. IoT devices can be divided into three main categories, that are for industrial, enterprise and consumer [9]. In industry, IoT devices are utilized in monitoring manufacturing processes to make sure it is running smoothly and optimally. When a fault arises during the process, IoT devices will detect it and alert the technician to the cause of the problem. Besides that, IoT devices used in the enterprise can assist in meetings. In a large company where they have a lot of meeting rooms, sensors placed in those rooms could allow the employee to detect the availability of rooms and the suitability of the room for scheduled meetings. For consumers, IoT devices could be present in smart homes where they can adjust the room's temperature, lighting and others based on the data received from sensors.

Most IoT devices use a generic architectural design [10]. The perception layer, network layer, middleware layer, and application layer are the four layers of IoT architecture. The perception layer contains two components which are the sensors and short distance conveyed network. Sensors are also known as the controller where it is used to gather data and implement control. The data gathered is then transmitted out by the sensor to the gateway. The communication network and the Internet are both under the network layer. The middleware layer is responsible for decision making as it contains extra functions like processing and computation. While the application layer is the bridge to connect IoT and users where it displays the reaction and the response when certain interaction is made with the IoT devices. Fig. 1 illustrate the four layers of the architectural design of IoT devices.

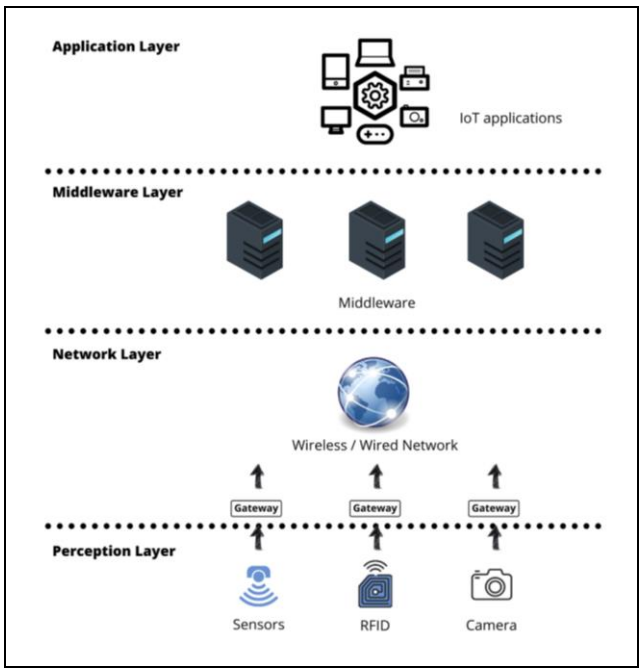

Fig. 1. The Four-Layer IoT Architecture

B. Vulnerabilities in IoT Devices

There are numerous vulnerabilities in IoT for each of the IoT verticals. Industrial IoT is vulnerable to device hijacking where attackers gain control of the industrial IoT endpoint without notice by the organization [11]. In the energy sector, Russian hackers take advantage of the vulnerabilities that exist due to the lack of knowledge or resources in securing IoT and have launched attacks on power grids in a few countries [12]. Due to the extensive use of mobile devices that rely on cloud storage in healthcare, IoT in the healthcare industry is more prone to data breaches [13]. The vulnerabilities in government IoT systems arise because of large amounts of devices located in public sector networks [14]. The IoT vulnerabilities in IoT

verticals are mostly caused by the vulnerabilities underlies in the IoT devices and the users who interact with the devices. Therefore, this literature review will briefly discuss the 5 most common vulnerabilities in IoT devices.

Based on a report by an online community which is Open Web Application Security Project (OWASP), one of the top vulnerabilities lies in the passwords used by IoT device users. The weaker and guessable the password, the more vulnerable it is to the IoT network. Especially for the user who left their device's default password unchanged, it will be easier for those hackers to brute-force and get their password, thus gaining control over their devices [15]. Therefore, IoT device users should change their password to a strong password that is long and contains a combination of numbers, characters, uppercase and lowercase letters. The second vulnerability is unsecured network services. While attackers try to take over an IoT endpoint, the primary and easiest breakthrough surface is the IoT device network model and services.

Unsecured ecosystem interfaces also contribute to the vulnerability of IoT devices. Secure encryption algorithms, strong authentication and authorization mechanism, and implementation of input and output filtering in ecosystem interfaces are crucial in reducing the vulnerabilities of IoT devices. Most of the IoT devices also do not contain any secure update mechanism, which leads to one of the major IoT vulnerabilities. They should include security features such as secure delivery, anti-rollback mechanisms and firmware validation. Finally, some of the IoT devices still utilize outdated or unsecured components. This outdated software or devices can degrade the overall security of the devices. Third-party software and hardware components that are non-authentic could be the entry point of attackers to start or continue an attack.

C. High Profile Cyberattacks in IoT

Among the cyberattacks in IoT, the most well-known cyber attack is a huge, distributed denial of service (DDoS) attack caused by Mirai malware. On 21st October 2016, a DDoS attack was launched, targeted on Dyn's servers. Dyn is a company that manages a large share of the Internet's domain name system (DNS) infrastructure [16]. Most of the popular websites are down due to the attack, this includes Twitter, Netflix, Amazon, GitHub and Reddit, which are the websites in the US and Europe.

While most of the previous DDoS attacks are done by computer, this DDoS attack is performed by a network of IoT devices infected by Mirai malware, which forms the "Mirai botnet"[17]. The Mirai malware propagated in IoT networks by scanning for devices in the same network which runs on ARC processors [18]. The malware will then attempt to login into those devices with default usernames and passwords. The devices that are infected by the Mirai malware are those that keep their default credential unchanged. These compromised IoT devices included video recorders, DVR players and home routers. 100,000 devices are estimated to be involved in the attack and they are instructed to bombard Dyn's servers with network traffic until the servers' collapse.

Dale Drew one of the cybersecurity experts who serves as the Chief Security Officer (CSO) at Level 3 Communications, which was acquired by CenturyLink states that the main goal of the attackers is to interrupt Dyn's services [19]. He draws this conclusion because the attacks targeted multiple ranges of domains authorized to Dyn. The attackers could just attack one domain if they intend to damage the owner of the particular domain. As a result, the attack causes Dyn servers to fail to process the user's request. Since the domain name could not be resolved, Internet users face difficulties in accessing websites that utilized Dyn's services, causing potential economic losses to the website's owner [20].

D. Supervised Learning Classification Algorithms

In this research, machine learning is utilized in malware attack classification where the connections captured are classified into malware attacks and benign traffic. The performance of machine learning is measured by how good it generalizes abstract data and how accurate it predicts output. Therefore, supervised classification algorithms are chosen because multi-class malware classification that requires discrete output.

1) Naïve Bayes

This collection of algorithms is based on Bayes' theorem which gives an assumption of independence among predictors [21]. The algorithms refer to the same principle where it assumes that a feature in a class is unrelated to the presence of any other feature. The Naïve Bayes equation is displayed below.

$$P(x_1, x_2, \dots x_n) = \frac{P(C_i)}{P(x_1, x_2, \dots x_n)} for\ 1 < i < k \tag{1}$$

2) Random Forest

Random Forest is a machine learning algorithm invented by Leo Breiman and Adele Cutler that mixes the output of several decision trees to produce a single conclusion [22]. There are three major hyperparameters of random forest algorithms that must be specified before training. The size of the nodes, the number of trees and the number of characteristics samples are all factors to consider. The benefits of using random forest algorithms are it decreases the tendency of overfitting, issue versatility and ease in ascertaining the feature importance.

E. Related Studies

Network anomaly detection begins by classifying network traffic. Therefore, there are numerous studies on malware traffic classification. Most studies mainly emphasise on ways to improve the performance of the classifier.

Arivudainambi *et al*. [23] proposed a solution using Artificial Neural Network (ANN) and Principal Component Analysis (PCA). This model is highly efficient in malware attack traffic classification and has 99% accuracy compared to most recent models. Yu *et al*. [24] proposed a malicious traffic

classification model through stacking Dilated Convolutional Autoencoders (DCAEs). This model is able to learn important features from the unlabelled dataset automatically and produce a low false alarm rate. Gao *et al.* [25] suggested a model based on deep belief network. Javaid *et al*. [26] proposed a model utilizing sparse auto encoder. Both designs function in network-based intrusion detection systems. Although representation learning can learn features straight away from raw input, the two studies above choose to construct flow feature datasets.

To classify the malware traffic in android, Alam *et al*. [27] applied Random Forest algorithms and 5-fold cross-validation. This study focuses on testing how different features and the number of trees affect the results of the experiment. Other than the classification of normal malware traffic, research is presented on encrypted traffic. Six most popular machine learning algorithms are involved in malware traffic detection and Random Forest is found to be the best suit for this problem domain [28]. Shafiq *et al.* [29] proposed a machine learning based hybrid feature selection algorithm to deal with an imbalanced network traffic dataset. Weighted mutual information and area under Receiver Operating Characteristic (ROC) curve metrics are used to choose significant features in network traffic.

III. DATASET

The dataset used is under the Aposemat IoT-23 dataset [6]. This dataset is chosen for the research because it comprises of network traffic captures in malicious and benign scenarios on real IoT devices It includes 3 captures of traffic from benign IoT devices and 20 captures of traffic from IoT devices infected with malware. Over 760 million packets and 325 million labelled flows were generated by more than 500 hours of traffic capture. The capture is performed between 2018 and 2019 at Stratosphere Laboratory in the Czech Republic. The malware dataset contains three major files, which are README.md, .pcap and conn.log.labeled for every IoT malware captured. The conn.log.labeled files originate from the conn.log log files generated by Zeek, which contains TCP/UDP/ICMP connections. This research involves 6 network traffic scenarios out of the 23 scenarios, a total of 482,378 rows. The description of each attribute in the conn.log.label files are depicted in Table I.

TABLE I. FEATURE IN IOT-23 DATASET

Feature	Description
ts	Timestamp of first packet in Unix Epoch format
uid	Unique connection ID
id.orig_h	Source IP address (ORIG)
id.orig_p	Source TCP/UDP port (/ ICMP code)
id.resp_h	Destination IP address (RESP)
id.resp_p	Destination TCP/UDP port
proto	Transport layer protocol
service	Automatically detect application protocol, if available
duration	Period of connection
orig_bytes	Source payload bytes; from sequence number if TCP
resp_bytes	Destination payload bytes; from sequence number if TCP
conn_state	Connection state
local_orig	If connection source local ‘T’, if remote ‘F’. This field is empty if Site::local_nets is undefined
local_resp	If connection respond to local ‘T’, if remote ‘F’. This field is empty if Site::local_nets is undefined
missed_bytes	Number of missing bytes in content gaps
history	Connection state history
orig_pkts	Number of packets from source
orig_ip_bytes	Number of source IP bytes
resp_packets	Number of packets from destination
resp_ip_bytes	Number of destination IP bytes
tunnel_parents	UID if encapsulating parent(s) is tunneled

IV. FRAMEWORK FOR IOT MALWARE DETECTION

There are three phases in this research. In the first phase, a review and study are done on the techniques and characteristics. In addition, data pre-processing will be carried out. The second phase focusses on implementing the chosen machine learning algorithms. The last phase analyses and discusses the result of the research.

A. Phase 1: Identification of Useful Attribute in Dataset to Train Classifier

The attributes in Table I are examined during data pre-processing.

The local origin (local_orig) and local response (local_resp) columns are removed because all the entries recorded are a “-”, indicating that no value is captured for both columns. Besides, the unique connection ID (uid) is dropped because it is not useful to train the classifier. The source (source_ip) and destination IP (dest_ip) addresses are excluded from the machine learning features because IP can be spoofed by the attackers [30]. Therefore, IP is infeasible to be used as an attack detection and classification feature.

The correlation matrix in Fig. 2 depicts the degree of association of all the attributes in the dataset. The orig_ip_bytes and resp_ip_bytes attributes that had a high correlation (> 0.95) are removed to save space and time during model training and testing. High correlation attributes are unnecessary since they generally have a similarity influence in prediction calculations.

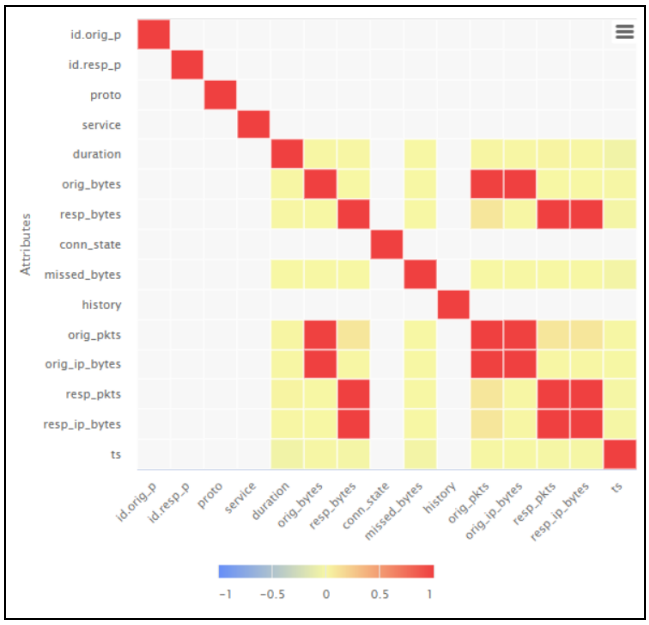

Fig. 2. Correlation Matrix

The feature selection is done by weighting the information gain, the weight for each attribute is shown in Fig. 3. The higher the value of attribute weight, the more relevant the attribute is to classification. Therefore, attributes with information gain weight below 0.5 are omitted. The attributes that have an information gain weighted below 0.5 are missed_bytes, service, proto and resp_pkts.

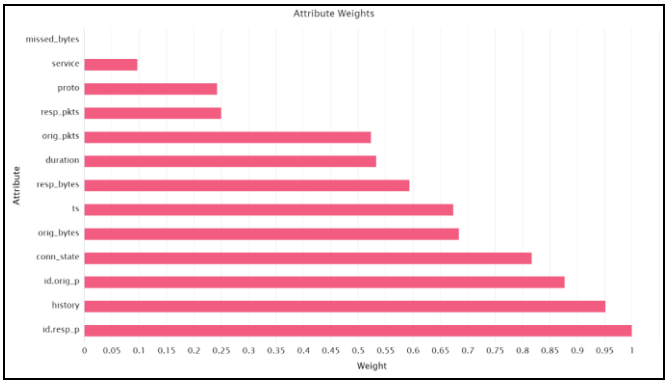

Fig. 3. Attribute Weights by Information Gain

The 9 remaining attributes are ts, id.orig_p, id.resp_p, duration, orig_bytes, resp_bytes, conn_state history and orig_pkts. These features are used in phase 2 to train the Naïve Bayes and Random Forest Models to classify the anomalies network flows.

B. *Phase 2: Development of Malware Detection and Classification Model*

10-fold cross validation is used in this experiment. This method divides the set into 10 parts, 9 parts are used for training and 1 part is used for testing. The process is repeated 10 times with a different part for testing each time. Since the dataset is highly imbalanced, the sampling type selected is stratified sampling. Stratified sampling creates random subsets with nearly equal proportions of each class as the whole dataset. Stratified Cross-Validation divides the data into k folds, ensuring that each fold with nearly equal proportions (class distribution, mean, variance, and etc.) of the original data. By using this method, all the class is present in each fold, ensuring that every class are included in model training and testing. The same settings applied for both Naive Bayes and Random Forest model. In Random Forest, the gain ratio is selected as the criterion because this information gain option modified the information gain for each attribute to control the breadth and uniformity. The maximal depth parameter controls the depth of the random forest's tree, in the experiment, it is set as 10. The voting strategy is assigned to "confidence vote", in which the prediction strategy is done by selecting the class with the highest accumulated confidence.

C. *Phase 3: Comparison of the Performance of Machine Learning Models*

The performance of Random Forest and Naïve Bayes algorithms is evaluated based on the confusion matrix. Confusion matrix applicable in our research as the classification output contains equal or more than two types of classes. To find out the machine algorithm that is the best in classifying malware attack, accuracy is defined as the number of correct predictions among all the predictions made on malware type [31]. The formula is displayed below.

$$Accuracy = \frac{TN+TP}{TN+FP+TP+FN} \quad (2)$$

Precision provides information on the percentage of points that are positive among all the points in the model that are predicted positive. The equation for precision is shown below.

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

While recall shows the true positive rate where among all the points that belong to positive, how many are predicted as positive.

$$Recall = \frac{TP}{TP + FN.} \quad (4)$$

The F1-Score combines the effect of both precision and recall, it displays the harmonic mean for precision and recall.

$$F1 - score = 2 \times \frac{Precision * Recall}{Precision + Recall.} \quad (5)$$

V. Result Analysis

There are seven classes involved in this experiment, one is the benign traffic and six are of malicious traffic. Table I shows the summary table that compare the accuracy, precision, recall

and F1-Score between Naïve Bayes and Random Forest. The Random Forest is the best model in all aspects, achieving micro F1-score of 98.55%.

TABLE II. COMPARISON OF NAIVE BAYES AND RANDOM FOREST PERFORMANCE METRIC

Metrics		**Algorithms**	
		Naïve Bayes (%)	***Random Forest (%)***
	Accuracy	99.42	99.92
Precision	Weighted	83.47	98.27
	Micro	81.46	98.26
Recall	Weighted	84.32	98.85
	Micro	84.24	98.85
F1 Score	Weighted	83.89	98.56
	Micro	82.83	98.55

The number of estimators (100 trees per forest) and the overall stability they provide are most likely contributing factor for this strong predictive ability. Furthermore, the depth of 10 appears to capture the most important features sufficiently to split the data without requiring additional nodes. As seen in the table, both models exceeded 80% in accuracy, recall, precision and F1-Scores. This shows that the ability of models to predict anomalies flow is moderate to strong. Naïve Bayes model has the worse micro precision rate (81.46%) and micro recall (84.24%), indicates that it has a high number of false positives and false negatives. This occurs because the dataset is imbalanced, and the Naïve Bayes model is unable to identify the overlap between classes and the noise in the data. The Random Forest also outperformed Naïve Bayes when predicting a single class that has the least data.

VI. LIMITATIONS

In this research, there are some constraints that prohibit the research in generating a more robust and accurate machine learning malware detection models. The major limitations the size of dataset and the hardware used for research. The IoT-23 dataset is considered a big data because the combination of all scenarios (malicious and benign) produces more than 1,000,000,000 Zeek flows (rows) for machine learning. If the original datasets are used, it will cause computational problem because the device used only has 8GB RAM and 500GB SSD storage. Therefore, the dataset is subsampled, and the malicious categories are greatly reduced. The models built become less robust because not all the malicious categories are included.

VII. SUGGESTION FOR IMPROVEMENT AND FUTURE WORKS

In this research, the main purpose is to compare and evaluate two supervised learning machine learning algorithms on already available datasets. This project can be expanded in the future by collecting own dataset, which will aid in overcoming many of the challenges that come with using other datasets. More IoT devices, more datasets, and the use of unsupervised learning algorithms may all be added to the data collecting process. IoT devices of the same brands may also be used to perform anomaly detection, giving more information about the network and application layers. Last but not least, the current study solely considers training and testing a labelled dataset; future research may investigate utilizing unsupervised learning algorithms like K-means and ISODATA clustering on unlabelled data to identify hostile activities.

VIII. CONCLUSION

This research compares the performance of Random Forest and Naïve Bayes algorithms in the detection of IoT malware attacks. The algorithm results in higher accuracy, precision, recall and F1-score can be used to implement in malware attack detection and classification.

Throughout the research, Random Forest and Naïve Bayes models are built and the result is discussed. In Section IV, the implementation of Random Forest and Naïve Bayes to detect IoT malware is explained, from phase 1 to phase 3. In phase 1, the dataset is subsampled from 3,419,676 rows to 482,378 rows. The useful attributes for model training and testing are identified by analyzing the dataset, removing correlated attributes and weighting the information gain for all attributes. Next, Naïve Bayes and Random Forest malware detection and classification models are built in phase 2. To achieve the objective in this phase, which is generating models that have high classification power, parameters are tuned numerous times to obtain satisfying results. In phase 3, the performance of both models is evaluated and compared in terms of accuracy, precision, recall and F1-score.

Both machine learning models obtain good results in all aspects. Nonetheless, Random Forest model outperformed the Naïve Bayes model as it can achieve a 98.55% of Micro Average F1-score. By completing this research, cybersecurity personnel can use this as a reference to identify the machine learning algorithms that can perform better in detecting malicious network flows, especially between Random Forest and Naïve Bayes.

ACKNOWLEDGMENT

The authors wish to thank all the experts involved for evaluation contribution of this research and the Ministry of Higher Education and University Teknologi Malaysia for funding this work under vot number (Q.J130000.3851.19J91).

REFERENCES

[1] Rayes, A. and S. Salam. (2019). Internet of things (IoT) overview. Internet of Things from hype to reality. Springer. 1-35.
[2] WatchGuard Threat Lab. (2020). Internet Security Report: Q4 202.
[3] Palo Alto Networks. (2020). Unit 42 IoT Threat Report.
[4] EMnify. (2020). IoT Attacks, Hacker Motivations, and Recommended Countermeasures. 2020 August 12; Available from: https://www.iotforall.com/iot-attacks-hacker-motivation.

[5] Eclipse Foundation. (2020). The Eclipse Foundation Releases IoT Commercial Adoption Survey Results. 2020 March 10; Available from: https://www.eclipse.org/org/press-release/20200310-iot-commercial-adoption-survey-2019.php.

[6] Garcia, S., A. Parmisano, and M. J. Erquiaga. (2020). IoT-23: A labeled dataset with malicious and benign IoT network traffic (Version 1.0.0). 2020: Zenodo.

[7] The Internet of Things. (2022). Cambridge Dictionary.

[8] IEEE. (2015). Internet of Things (IoT) Ecosystem Study.

[9] Posey, B. (2022). IoT devices (internet of things devices).

[10] Muhammad, F., W. Anjum, and K. S. Mazhar. (20150. A critical analysis on the security concerns of internet of things (IoT). *International Journal of Computer Applications, 111*(7), 1-6.

[11] Archon Secure Team. (2021). What are the Risks Associated with Industrial IoT (Industrial Internet of Things)? 2021 January 11; Available from: https://www.archonsecure.com/blog/what-are-the-risks-associated-with-industrial-iot.

[12] Joy, K. (2019). As IoT Enhances the Energy Sector, Security Issues Grow. 2019 April 30; Available from: https://biztechmagazine.com/article/2019/04/iot-enhances-energy-sector-security-issues-grow.

[13] Incognito Forensic Foundation. (2018). The technological revolution called the Internet of Things (IoT). 2018 June 1; Available from: https://ifflab.org/how-healthcare-iot-is-vulnerable-to-cyber-security-threats/.

[14] Alcatel Lucent Enterprise. (2019). The Internet of Things in the Enterprise.

[15] Mukherjee, L. (2020). The OWASP IoT Top 10 List of Vulnerabilities. 2020 April 30; Available from: https://sectigostore.com/blog/owasp-iot-top-10-iot-vulnerabilities/.

[16] Antonakakis, M., *et al*. (2017). Understanding the Mirai Botnet, in 26th USENIX Security Symposium (USENIX Security 17). 2017, USENIX Association. 1093-1110.

[17] Wang. (2018). The 2016 Dyn Attack and its Lessons for IoT Security. 2018; Available from: https://mse238blog.stanford.edu/2018/07/clairemw/the-2016-dyn-attack-and-its-lessons-for-iot-security/.

[18] Cloudflare. What is the Mirai Botnet? Available from: https://www.cloudflare.com/learning/ddos/glossary/mirai-botnet/.

[19] Greene, T. (2016). How the Dyn DDoS attack unfolded. 2016 October 22; Available from: https://www.networkworld.com/article/3134057/how-the-dyn-ddos-attack-unfolded.html.

[20] Greenstein, S. (2019). The aftermath of the dyn DDOS attack. *IEEE Micro*, *39*(4), 66-68.

[21] Pedamkar, P. (2021). Supervised Machine Learning Algorithms. 2021 November 12; Available from: https://www.educba.com/supervised-machine-learning-algorithms/.

[22] IBM Cloud Education. (2020). Random Forest. 2020 December 7; Available from: https://www.ibm.com/cloud/learn/random-forest.

[23] Arivudainambi, D., V. K. K. A, and P. Visu. 2019. Malware traffic classification using principal component analysis and artificial neural network for extreme surveillance. *Computer Communications, 147*, 50-57.

[24] Yu, Y., J. Long, and Z. Cai. (2017). Network intrusion detection through stacking dilated convolutional autoencoders. Security and Communication Networks.

[25] Gao, N., *et al*. (2014). An intrusion detection model based on deep belief networks. 2014 *Second International Conference on Advanced Cloud and Big Data*. IEEE.

[26] Javaid, A., *et al*. (2016). A deep learning approach for network intrusion detection system. *Eai Endorsed Transactions on Security and Safety, 3*(9), p. e2.

[27] Alam, M. S. and S. T. Vuong. (2013). Random forest classification for detecting android malware. *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE cyber, physical and social computing.* IEEE.

[28] Anderson, B. and D. McGrew. (2017). Machine learning for encrypted malware traffic classification: accounting for noisy labels and non-stationarity. *Proceedings of the 23rd ACM SIGKDD International Conference on knowledge discovery and data mining.*

[29] Shafiq, M., *et al*. (2018). A machine learning approach for feature selection traffic classification using security analysis. *The Journal of Supercomputing*, *74*(10), 4867-4892.

[30] Seo, J. W. and S. J. Lee. (2016). A study on efficient detection of network-based IP spoofing DDoS and malware-infected Systems. SpringerPlus, *5*(1): 1878.

[31] Visa, S., *et al*. (2011). Confusion matrix-based feature selection. *MAICS*, *710*(1), 120-127.