Indonesian Journal of Electrical Engineering and Computer Science
Vol. 22, No. 1, April 2021, pp. 450~459
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v22.i1.pp450-459

The trend malware source of IoT network

Susanto[1], M. Agus Syamsul Arifin[2], Deris Stiawan[3], Mohd. Yazid Idris[4], Rahmat Budiarto[5]
[1,2]Faculty of Computer, Universitas Bina Insan, Indonesia
[1,2]Faculty of Engeenering, Universitas Sriwijaya, Indonesia
[3]Faculty of Computer Science, Universitas Sriwijaya, Indonesia
[4]Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, Malaysia
[5]College of Computer Science & Information Technology, Albaha University, Al Baha, Saudi Arabia

Article Info

Article history:

Received Mar 19, 2020
Revised Dec 5, 2020
Accepted Jan 11, 2021

Keywords:

Data repository malware
Feature extraction
IoT
Malware
Malware detection

ABSTRACT

Malware may disrupt the internet of thing (IoT) system/network when it resides in the network, or even harm the network operation. Therefore, malware detection in the IoT system/network becomes an important issue. Research works related to the development of IoT malware detection have been carried out with various methods and algorithms to increase detection accuracy. The majority of papers on malware literature studies discuss mobile networks, and very few consider malware on IoT networks. This paper attempts to identify problems and issues in IoT malware detection presents an analysis of each step in the malware detection as well as provides alternative taxonomy of literature related to IoT malware detection. The focuses of the discussions include malware repository dataset, feature extraction methods, the detection method itself, and the output of each conducted research. Furthermore, a comparison of malware classification approaches accuracy used by researchers in detecting malware in IoT is presented.

Corresponding Author:

Deris Stiawan
Faculty of Computer Science
Universitas Sriwijaya
Palembang, Indonesia
Email: deris@unsri.ac.id

1. INTRODUCTION

Internet of things (IoT) has different characteristics from other technologies that provide research opportunities in the study of malware in IoT. These characteristics are: 1) having an uncontrolled access environment where various devices connected to the IoT network are highly mobile. 2) heterogeneity where the diversity of devices interacting between devices that have high computing and those that have low-end computing such as servers with sensors and actuator devices. 3) scalability where the network on IoT devices is globally distributed but can be scaled in an application. 4) resource constraints where low energy requirements make the IoT design minimalist, so sensors and actuators limit security [1].

Malware or malicious software is a threat to information security and affects a computer system, a computer network, as well as cellular devices through the exploitation of system vulnerabilities [2]. Malware detection is a massive challenge at any time [3]. Malware detection is an action that must be prepared in the fight against attacks on IoT data security devices that were not designed during the initial stages of network development [4]. Malware may disrupt the IoT system/network when it resides in the network, or even harm the network operation. Therefore, malware detection in the IoT system/network becomes an important issue. Research works related to the development of IoT malware detection have been carried out with various methods and algorithms to increase detection accuracy. A malware detection system in IoT is a system that can recognize, even to find malware in a computer system, network traffic, node sensor packet data, in files, and inside the software, inside hardware, or an executable file installed on a computer system.

This paper attempts to identify problems and issues in IoT malware detection presents an analysis of each step in the malware detection as well as provides alternative taxonomy of literature related to IoT malware detection. The focuses of the discussions include malware repository dataset, feature extraction methods, the detection method itself, and the output of each conducted research.

The author of the paper provides an understanding of the evaluation methods of malware detection in IoT in addition to knowledge of data repositories, feature extraction, and detection methods. In particular, the study of malware literature on IoT is different from the study of malware literature on existing IoT, as listed in Table 1.

Table 1. Comparison of malware literature studies in IoT

Discussion Topics	Researchers			This paper
	Karanja *et al.*, 2017 [1]	Costin and Zaddah., 2018 [5]	Tahaei *et al.*, 2020 [6]	Susanto *et al.*, 2020
Data repository	-	√	√	√
Feature extraction	-	-	-	√
Detection Method	-	-	√	√
Output	-	-	-	√

2. REVIEW OF LITERATURE

2.1. Data repository malware

Malware detection is a part of the intrusion detection system (IDS). Research works on IoT malware detection use various datasets and traffic. Table 2 depicts a comparison of malware data sources versus evaluation methods used by researchers.

Authors of this paper observe from the results of a literature study that there are three types of malware source data used in IoT malware detection research. First, the use of malware captured directly from executable files, processors, or networks. The second one is the use of malware dataset. The third one is the use of malware captured from a testbed network.

Table 2. Comparison of data repository used by researchers

Author(s)	Category Name			Evaluation Method	Notes
	Testbed	Captured	Dataset		
Takase *et al.*, 2019 [7]		√		Experiment	Use information from processor
Wu *et al.*, 2019 [8]		√		Experiment	Data from network traffic packet
Dinakarrao *et al.*, 2019 [9]		√		Experiment & Real-time	Data from 20 temperature sensors
Kumar and Lim., 2019 [10]	√			Experiment	Data from network traffic
Wei and Qiu., 2018[11]	√			Simulation & Real	Use weather station for sensor data collection
Han *et al.*, 2019 [12]			√	Experiment	Malware dataset from virus share
Xiao *et al.*, 2019 [13]			√	Experiment	Malware dataset from VX Heaven
Liu *et al.*, 2019 [14]			√	Experiment	Malware dataset from DREBIN
Naeem *et al.*, 2019 [15]			√	Experiment	Malware dataset from the research lab of University California and IKM Lab National Cheng Kung University, Taiwan
Cui *et al.*, 2018 [16]			√	Experiment	Malware dataset from Vision Research
Kumar *et al.*, 2019 [17]			√	Experiment	Malware dataset from the Chinese App Store and Google Play Store
Alhanahnah *et al.*, 2018 [18]			√	Experiment	Malware dataset from IoTPOT team
Ullah *et al.*, 2019 [19]			√	Experiment	Malware dataset from Google Code Jam
Haddadpajouh *et al.*, 2018 [20]			√	Experiment	Malware dataset from VirusTotal
Alasmary *et al.*, 2019 [21]			√	Experiment	Malware dataset from CyberIOCs
Dovom *et al.*, 2019 [22]			√	Experiment & Simulation	Malware dataset from Vx-Heaven, and Kaggle
Le *et al.*, 2019 [23]			√	Experiment & Real-time	Malware dataset from VirusShare and IoTPOT team
Su *et al.*, 2018 [24]			√	Experiment	Malware dataset from IoTPOT team
Liu *et al.*, 2019 [25]			√	Experiment	Malware dataset from UCI Repository
Karbab *et al.*, 2018 [26]			√	Experiment	Malware dataset from virus share, Malgenome, and Drebin

Table 2. Comparison of data repository used by researchers (continue)

Author(s)	Category Name			Evaluation Method	Notes
	Testbed	Captured	Dataset		
Nguyen *et al.*, 2018 [27]			√	Experiment	Malware dataset from IoTPOT team
Azmoodeh *et al*., 2018 [28]			√	Experiment	Malware dataset from VirusTotal
Tzagkarakis *et al.*, 2019 [29]			√	Experiment	Malware dataset from UCI Repository
Dietz *et al*., 2018 [30]	√			Experiment & Real-time	Data from the access router
Meidan *et at*., 2018 [31]			√	Experiment	Malware dataset from UCI Repository
McDermott *et al*., 2018 [32]	√			Experiment	Data from network traffic
Bahsi *et al.*, 2018 [33]			√	Experiment	Malware dataset from UCI Repository
Abusnaina *et al*, 2019 [34]			√	Experiment	Malware dataset from CyberIOCs
Manzanares *et al.*, 2019 [35]			√	Experiment	Malware dataset from UCI Repository and Cyber Range Lab of UNSW Canberra
Namanya *et al.*, 2019 [36]			√	Experiment	Malware dataset from the repository of Nettitude Ltd, UK
Ham *et al*., 2014 [37]			√	Experiment	Malware dataset from Ham *et al*
Ren *et al.*, 2020 [38]			√	Experiment	Malware dataset from VirusShare and Google Play Store
Nguyen *et al.*, 2020 [39]			√	Experiment	Malware dataset from VirusShare and IOTPOT team
Jung *et al*., 2020 [40]		√		Experiment	Data from power consumption

2.2. Feature Extraction

The first phase of malware detection is feature extraction. The extracted feature is initial information contains in an input file or resulted from an information processing [41]. The extraction process can be carried out using static analysis, dynamic analysis, and a combination of both [42-44]. A survey by researchers in [43] reports that static analysis consists of API calls, control flow graph (CFG), Opcode, and N-gram; Dynamic analysis consists of function calls, function parameters, instruction traces, and instruction flow. S. Talukder [45] mention that static analysis consists of Opcode, N-gram, syntactic library, CFG, string signature, and others; dynamic analysis is a controlled environment such as virtual machines, simulators, emulators, sandboxes, and others. K. Diaz-Chito *et al.* [46] shows that the extraction process can also incremental. Furthermore, research work in [47] shows that the extraction process can also use deep learning. The feature extraction technique used in malware detection researches varies, some of them, as summarized in Table 3.

Table 3. Various feature extraction used in related researches

Author(s)	Feature			Notes	Pros and contras
	Static Analysis	Dynamic Analysis	Other		
Takase *et al.*, 2019 [7]		Qemu		Extracting malware data from CPU information	Using an open-source emulator; The information obtained is incomplete if the source code is not changed
Kumar and Lim, 2019 [10]			Feature vector	Extract malware from a data traffic packet	Extraction results can be stored in an online database
Xiao *et al.*, 2019 [13]	API calls	Cuckoo Sandbox	Stacked AutoEncoders	Extracting Portable executable files	Can study malware behavior
Naeem, 2019 [15]			Deep Convolutional Neural Network	Extracts executable malware files into color images	Can automatically extract malware; The time needed for the extraction process is faster
Cui *et al.*, 2018 [16],			Convolutional Neural Network	Extracts executable malware files into grayscale images	Can extract malware automatically
Kumar *et al.*, 2019 [17]		Dex2Jar	Blockchain	Extracting executable .apk files	Faster and more accurate in malware extraction
Alhanahnah *et al*., 2018 [18]	N-gram	Yara		Extracting string feature	Can execute word sequences on unique IP addresses;
Ullah *et al.*, 2019 [19]			Convolutional Neural Network	Extracts executable malware files into color images	Get a better visualization of malware
Haddapajouh *et al*., 2018 [20]	Opcode and Object-dump			Extracting malware from Debian package files	Object-dump is only compatible with Raspberry Pi II processors

Table 3. Various feature extraction used in related researches (continue)

Author(s)	Feature Static Analysis	Feature Dynamic Analysis	Other	Notes	Pros and contras
Alasmary *et al*., 2019 [21]	Control flow graph (CFG)	Radare2		Executable and Linkable Format (ELF) files convert into binary files	Has algorithmic and structural properties so that it can be used in understanding the level of complexity of codes and avoidance analysis techniques;
Dovom *et al*., 2019 [22]	Opcode feature			Extracting Executable file into Binary information	
Le *et al*., 2019 [23]		Sandbox; Strace	Deep feature	Extract malware from ELF files	Can create a structured calling system
Su *et al*., 2018 [24]			Convolutional Neural Network	Extracting DDOS malware into grayscale images	Can extract malware automatically, can learn features that are difficult to find and understand by humans
Liu *et al*., 2019 [25]			incremental	Extract malware from data traffic	Can extract dynamic network traffic at high speed
Karbab *et al*., 2018 [26]			Embedding method	Extracting Android DEX files	Can extract malware automatically
Nguyen *et al*., 2018 [27]			Convolutional Neural Network	Extract scala, Extract binary code into color images and Extracts the Executable and Linkable Format files convert into binary files	Simple and easy to use; Extracts into fixed-size color images; Extraction results into variable-sized vectors
Azmoodeh *et al*., 2018 [28]	Opcode and Object-dump			Extract malware from PE Files	Can avoid and eliminate less instructive Opcode
Tzagkarakis *et al*., 2019 [29]			Incremental	Extracting malware from packet transmissions	Fast in extracting malware
Meidan *et al*., 2018 [31]			Incremental	Extracting malware from packet transmissions	Fast in extracting malware
Abusnaina *et al*., 2019 [34]	Control flow graph	Radare2		Extract malware from executable files	Can extract a variety of different algorithmic
Namanya *et al*., 2019 [36]	API calls	Hashdeep		Extract malware from PE Files	
Ren *et al*., 2020 [38]			Ghost and Spydealer	Extract the APK malware file into a grayscale image	Can convert images into 2-dimensional arrays
Nguyen *et al*., 2020 [39]	Rooted subgraph			Extract the ELF file into a PSI graph	Effective in used in detecting malware with machine learning
Jung *et al*., 2020 [40]			Threshold-based segmentation	Extracting malware Mirai	

2.3. Malware detection methods

Various methods are used in malware detection research. A survey study by [48, 49] reveals that malware detection in IoT can use machine learning and deep learning methods. Another survey study by [50] says that malware detection in the CPU can use an emulator. Each method has advantages as well as disadvantages. A comprehensive study comparison of the use of malware detection methods was done by the author of this paper and summarized in Table 4.

Table 4. Comparison of the malware detection methods

Author	Category	Methods/ Algorithm	Pros and cons	Accuracy
Takase *et al*., 2019 [7]	Emulator	*Qemu*	High accuracy in malware detection.	100%
Wu *et al*., 2019 [8]	Machine learning	Bayesian Model Update Method	Detecting malware based on traffic data. Having high accuracy, ability to filter unuseful data or data having negative impacts. The attribute must be independent	96%
Dinakarrao *et al*., 2019 [9]	Machine learning	OneR	Detecting malware without creating overhead. If the performance degrades under a threshold, then the regulation process is stopped. Needing data in bulk	92%
Kumar and Lim., 2019 [10]	Machine learning	Random Forest, k-NN, Gaussian Naïve Bayes	High accuracy in malware detection.	RF = 88.8%; k-NN= 94.44%; GNB= 77.78%
Wei and Qiu., 2018 [11]	Emulator	Augmented Dickey-Fuller test and Mann-Kendall Test	Ability to know IoT devices that quickly infected	

Table 4. Comparison of the malware detection methods (continue)

Author	Category	Methods/ Algorithm	Pros and cons	Accuracy
Han *et al.*, 2019 [12]	Machine learning	Systematic profiling	Detection and classification of malware with high accuracy	99.76%
Xiao *et al.*, 2019 [13]	Hybrid	Stacked Auto Encoders with Decision Tree	Malware Detection with high accuracy.	98.6%
Liu *et al.*, 2019 [14]	Machine learning	Neural Network, Logistic Regression, Decision Tree, Random Forest, Extreme Tree	Detecting malware with high accuracy	NN=99.83%; LR=99.45%; DT=99.86%; RF=99.92%; ET=99.96%
Naeem *et al.*, 2019 [15]	Deep learning	Deep Convolutional Neural Network	Malware Detection with high accuracy. High computing time and resources are needed.	98.18%
Cui *et al.*, 2018 [16]	Deep learning	Convolutional Neural Network	The speed of detection is significantly faster than other methods. Detecting malware with high accuracy. Requiring to modify the size of all inputted figures	94.5%
Kumar *et al.*, 2019 [17]	Hybrid	Blockchain with naïve bayes	Increasing the run-time malware detection with higher accuracy for detecting malware	98%
Alhanahnah *et al.*, 2018 [18]	Machine learning	K-Means	The same IP address matching can classify malware. Vulnerability against string confusion and encryption	85.2%
Ullah *et al.*, 2019 [19]	Deep learning	Deep Neural Network	Classification malware with high accuracy.	97.6%
Haddadpajouh *et al.*, 2018 [20]	Deep learning	Recurrent Neural Network	High accuracy in malware detection, additional computation is required for renewing neuron's weights. Use a small dataset compared to the real cyber-attack.	94%
Alasmary *et al.*, 2019 [21]	Deep learning	Convolutional Neural Network	It is detecting malware and classification malware with high accuracy.	99.66%
Dovom *et al.*, 2019 [22]	Machine learning	Fuzzy Pattern Tree	Malware detection with high accuracy.	99.834%
Le *et al.*, 2019 [23]	Deep learning	Convolutional Neural Network	Detecting malware with high accuracy, Only working on IoT bot files, not yet being scaled up to other dangerous lines of IoT devices	97.22%
Su *et al.*, 2018 [24]	Deep learning	Convolutional Neural Network	Requires a good graphics card to speed up the training process. High accuracy of malware classification	94.67%
Liu *et al.*, 2019 [25]	Deep learning	Convolutional Neural Network	High accuracy of classification malware.	99.57%
Karbab *et al.*, 2018 [26]	Deep learning	Neural Network	Accurate in detecting malware, Efficiency on some architecture, and needing manual categorization.	99.84%
Nguyen *et al.*, 2018 [27]	Deep learning	Convolutional Neural Network	Malware entropy is higher than non-malware files. Needing much more time	100%
Azmoodeh *et al.*, 2018 [28]	Deep learning	Convolutional Neural Network	Reducing junk codes injection attack. Detecting malware with high accuracy	98.37
Tzagkarakis *et al.*, 2019 [29]	Machine learning	Orthogonal matching pursuit	With limited computation, resources can detect botnet attacks accurately	99%
Dietz *et al.*, 2018 [30]		Scanning and Isolation	The isolation approach systematically protects IoT networks that are vulnerable to Mirai infection	
Meidan *et al.*, 2018 [31]	Deep learning	Deep Autoencoders	Very fast at detecting malware attacks	
McDemott *et al.*, 2018 [32]	Deep learning	Recurrent Neural Network	High accuracy and prediction for botnet malware	99%
Bahsi *et al.*, 2018 [33]	Machine learning	decision tree and k-NN	The classification process requires lower computing power so that it can be used to work in real-time easily in cyber-security analysis	DT=98.9%; k-NN=94.9%
Abusnaina *et al.*, 2019 [34]	Deep learning	Convolutional Neural Network	Requires a slight change in graph topology in modifying features. High misclassification rate	97.13%
Manzanares *et at.*, 2019 [35]	Machine learning	Random Forest, and k-NN	Increasing accuracy	RF=99.94%; k-NN=99.94%
Namanya *et al.*, 2019 [36]	Machine learning	Fuzzy logic and Command Factor	They are creating malware classification mechanism and detecting malware with high accuracy. Need hash database.	FL=91,6%; CF=91.6%
Ham *et al.*, 2014 [37]	Machine learning	Support Vector Machine	Detecting malware with high accuracy	99.5%
Ren *et al.*, 2020 [38]	Deep learning	Dex CNN and Dex CRNN	There are no file size limits, resulting in more false positives. Requires a longer time for the detection process	Dex CNN=93.4%; Dex CRNN=95.8%

Table 4. Comparison of the malware detection methods (continue)

Author	Category	Methods/ Algorithm	Pros and cons	Accuracy
Nguyen *et al.*, 2020 [39]	Machine learning	Support Vector Machine, Decision Tree, k-NN, Random Forest, and Bagging	The use of rooted subgraph extraction features with machine learning results in better detection accuracy. Allow for errors in the storage of all subgraphs	SVM= 99.6%; Bagging=976%; DT= 98.7%; RF=991%; k-NN=99.2%
Jung *et al.*, 2020 [40]	Deep learning	Convolutional Neural Network	It is detecting malware with high accuracy. It cannot be integrated on IoT devices.	98.6%

2.4. Output

Overall, the output of the existing IoT malware detection researches is in the form of scores and labels. The score is the output of every trial in the experiment in the form of detection accuracy rankings. Research in [22] produces classification accuracy in terms of the highest rank. The label is the output from every experimental trial in the form of label 'malware' or 'benign.' Research in [14] produces output from detecting malware in the form of malware label and benign label. Research in [25] considers the output is in the form of benign traffic label and attack traffic label. Research in [7] produces output in the form of a normal label and attack label.

3. DISCUSSION AND ANALYSIS

Literature shows that in IoT malware detection researches, the malware data repository (dataset) is taken from testbed, self capturing, and various public dataset sources. Table 2 presents data repositories that have been used in researches that show 76.47% using malware dataset, 11.76% using malware captured directly from processor and network, and 11.76% using the testbed network. The most used public dataset is sourced from IoTPOT of Yokohama National University. The dataset is labeled by two types: malware and benign. From the data repositories used by researchers, the majority of IoT malware detection research is mostly only done as an experiment in a laboratory. It is not done in a real-time fashion so that it becomes a challenge on how to implement IoT malware detection in real-time. IoT technology has different characteristics, so that it has a more significant problem in detecting malware in real-time. The first challenge is developing a fast and lightweight detection system without using huge costs [9]. Second, developing energy-efficient detection systems with limited resources [18], and the third one is identifying known malware and new malware in real cyberattacks using a small dataset at the time of the experiment [20].

In extracting the information from the dataset and then in the classification, data in Table 3 presents feature extraction consisting of static analysis, dynamic analysis, and also a combination of static and dynamic analysis. Also, there is a feature extraction using incremental, deep learning, and blockchain. Attributes in the static analysis that have been used by researchers include API calls, N-grams, Opcodes, Control flow graph, rooted subgraphs. There are also those using open-source Object-dump tools, while in dynamic analysis, the tools that have been used by researchers in the form of open-source tools include Cuckoo Sandbox, Dex2Jar, Yara, Qemu, Radare2, Object-dump, Strace, hashdeep. Each malware analysis tool can be used to extract different malware files. From the results of the literature studies, extraction feature is used to extract malware from network traffic, executable files, and processors. The feature extraction method that is most widely used by researchers in deep learning. By using deep learning, the features can automatically be extracted [15, 16, 24], and be able to learn on its own from the malware [13].

Data in Table 4 presents the malware detection methods on IoT. The information on the detection methods from literature is divided into three categories, namely machine learning, deep learning, and emulator. Machine learning methods that have been used by researchers include logistic regression, Decision trees, random forests, extreme trees, k-means, fuzzy pattern trees, fuzzy logic, orthogonal matching pursuit, support vector machines, k-nearest neighbors, and Bagging. In contrast, in deep learning, the methods that have been used by researchers include neural networks, convolutional neural networks, deep neural networks, deep convolutional neural networks, recurrent neural networks, deep autoencoders, Dex CNN and Dex CRNN. Besides, researchers also used the Qemu emulator and the augmented Dickey-Fuller test and the Mann-Kendall test. Then there are also researchers with hybrid methods, including neural network stacked auto encoders with decision tree and blockchain with naive Bayes. Machine learning and deep learning are used to perform binary classification, i.e., to classify whether the application file is a malware or not. From the results of literature studies, the most widely used malware detection method is deep learning with the convolutional neural network algorithm. The convolutional neural network algorithm requires a good graphics card to speed up the training process [24]. Decision tree, Orthogonal matching pursuit, and k-NN in the classification process require lower computing power so that it can be used to work in real-time efficiently in the analysis of malware attacks on IoT [33]. The output is a final result of malware detection with the majority in the form of labels (malware and benign).

There are several indicators used in measuring the performance of classification accuracy, from the use of malware repository data, feature extraction to malware classification methods. The indicators used in each study differ from each other, and some papers do not address the issue of detection accuracy. In this paper, the authors present the results of a literature review paper on malware detection on IoT by comparing the accuracy of each approach used by researchers, as shown in Table 4. The results of the study presented in Table 4 have an average high level of detection accuracy.

Furthermore, we analyze literature that contributes to IoT malware detection researches. The IoT networks have different characteristics so that it becomes a challenge in malware detection. Data acquisition from sensors, Android devices, and network protocols should be extracted using the appropriate method with the primary aim that the information of the data can be read. The information yielded from the extraction process will then be analyzed to determine whether the data packet is malware or benign. In some cases, there are traffic data that are not recognized, so they need an algorithm that can identify those data using a smart/intelligent system automatically. Therefore, the feature extraction and method in IoT malware detection become the primary key to the success of malware detection.

4. CONCLUSION AND FUTURE WORK

An alternative taxonomy of literature related to IoT malware detection has been discussed. The focuses of the discussions include malware repository dataset, feature extraction methods, the detection method itself, and the output of each conducted research. In conducting malware detection experiments on IoT, input data may use self captured data, testbed as well as public datasets. Several datasets for malware detection on IoT has been provided by researchers and are ready to be used for research according to the selected scenario. Feature extraction is one of the crucial processes in malware detection. Extracting malware features may use static or dynamic methods or a combination of both, even combining with the use of deep learning features. The dynamic methods can be implemented using open source tools. Each feature extraction has advantages and disadvantages of each. The classification method is used to determine the output of malware detection, whether the data is malware or not. From the classified output, the level of accuracy of the detection can be measured. Besides, this paper has analyzed each step of IoT malware detection. The alternative taxonomy complements existing literature studies, strips issues of malware detection in IoT network/system, and helps researchers in designing reliable malware detection system for IoT network/system. Real-time IoT malware detection system development is considered one of the future works in this research area.

REFERENCES

[1] E. M. Karanja, S. Masupe, and J. Mandu, “Internet of Things Malware : A Survey,” *Int. J. Comput. Sci. Eng. Surv.*, vol. 8, no. 3, pp. 1-20, 2017, doi: 10.5121/ijcses.2017.8301.

[2] A. O. Eze and C. C. E, “Malware Analysis and Mitigation in Information Preservation,” *IOSR J. Comput. Eng.*, vol. 20, no. 4, pp. 53-62, 2018, doi: 10.9790/0661-2004015362.

[3] M. Akour, I. Alsmadi, and M. Alazab, “The malware detection challenge of accuracy,” *2016 2nd Int. Conf. Open Source Softw. Comput. OSSCOM 2016*, 2017, doi: 10.1109/OSSCOM.2016.7863750.

[4] C. Vorakulpipat, E. Rattanalerdnusorn, P. Thaenkaew, and H. Dang Hai, “Recent challenges, trends, and concerns related to IoT security: An evolutionary study,” *Int. Conf. Adv. Commun. Technol. ICACT*, vol. 2018-Febru, pp. 405-410, 2018, doi: 10.23919/ICACT.2018.8323774.

[5] A. Costin and J. Zaddach, “IoT Malware: Comprehensive Survey, Analysis Framework and Case Studies,” *BlackHat 2018 USA*, 2018.

[6] H. Tahaei, F. Afifi, A. Asemi, F. Zaki, and N. B. Anuar, “The rise of traffic classification in IoT networks: A survey,” *J. Netw. Comput. Appl.*, vol. 154, p. 102538, 2020, doi: 10.1016/j.jnca.2020.102538.

[7] H. Takase, R. Kobayashi, M. Kato, and R. Ohmura, “A prototype implementation and evaluation of the malware detection mechanism for IoT devices using the processor information,” *Int. J. Inf. Secur.*, 2019, doi: 10.1007/s10207-019-00437-y.

[8] F. Wu, L. Xiao, and J. Zhu, “Bayesian Model Updating Method Based Android Malware Detection for IoT Services,” *2019 15th Int. Wirel. Commun. Mob. Comput. Conf.*, pp. 61-66, 2019, doi: 10.1109/iwcmc.2019.8766754.

[9] S. M. Pudukotai Dinakarrao, H. Sayadi, H. M. Makrani, C. Nowzari, S. Rafatirad, and H. Homayoun, “Lightweight Node-level Malware Detection and Network-level Malware Confinement in IoT Networks,” *Proc. 2019 Des. Autom. Test Eur. Conf. Exhib. DATE 2019*, pp. 776-781, 2019, doi: 10.23919/DATE.2019.8715057.

[10] A. Kumar and T. J. Lim, “EDIMA: Early Detection of IoT Malware Network Activity Using Machine Learning Techniques,” *2019 IEEE 5th World Forum Internet Things*, pp. 289-294, 2019, doi: 10.1109/wf-iot.2019.8767194.

[11] D. Wei and X. Qiu, "Status-based Detection of malicious code in Internet of Things (IoT) devices," *2018 IEEE Conf. Commun. Netw. Secur. CNS 2018*, pp. 1-7, 2018, doi: 10.1109/CNS.2018.8433183.
[12] W. Han, J. Xue, Y. Wang, Z. Liu, and Z. Kong, "MalInsight: A systematic profiling based malware detection framework," *J. Netw. Comput. Appl.*, vol. 125, pp. 236-250, 2019, doi: 10.1016/j.jnca.2018.10.022.
[13] F. Xiao, Z. Lin, Y. Sun, and Y. Ma, "Malware Detection Based on Deep Learning of Behavior Graphs," *Math. Probl. Eng.*, vol. 2019, 2019, doi: 10.1155/2019/8195395.
[14] X. Liu, X. Du, X. Zhang, Q. Zhu, H. Wang, and M. Guizani, "Adversarial samples on android malware detection systems for IoT systems," *Sensors (Switzerland)*, vol. 19, no. 4, pp. 1-16, 2019, doi: 10.3390/s19040974.
[15] H. Naeem, "Detection of Malicious Activities in Internet of Things Environment Based on Binary Visualization and Machine Intelligence," *Wirel. Pers. Commun.*, vol. 108, no. 4, pp. 2609-2629, 2019, doi: 10.1007/s11277-019-06540-6.
[16] Z. Cui, F. Xue, X. Cai, Y. Cao, G. G. Wang, and J. Chen, "Detection of Malicious Code Variants Based on Deep Learning," *IEEE Trans. Ind. Informatics*, vol. 14, no. 7, pp. 3187-3196, 2018, doi: 10.1109/TII.2018.2822680.
[17] R. Kumar, X. Zhang, W. Wang, R. U. Khan, J. Kumar, and A. Sharif, "A Multimodal Malware Detection Technique for Android IoT Devices Using Various Features," *IEEE Access*, vol. 7, no. May, pp. 64411–64430, 2019, doi: 10.1109/ACCESS.2019.2916886.
[18] M. Alhanahnah, Q. Lin, Q. Yan, N. Zhang, and Z. Chen, "Efficient signature generation for classifying cross-architecture IoT malware," *2018 IEEE Conf. Commun. Netw. Secur. CNS 2018*, pp. 1-9, 2018, doi: 10.1109/CNS.2018.8433203.
[19] F. Ullah *et al.*, "Cyber Security Threats Detection in Internet of Things Using Deep Learning Approach," *IEEE Access*, vol. 7, pp. 124379-124389, 2019, doi: 10.1109/access.2019.2937347.
[20] H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K. K. R. Choo, "A deep Recurrent Neural Network based approach for Internet of Things malware threat hunting," *Futur. Gener. Comput. Syst.*, vol. 85, pp. 88-96, 2018, doi: 10.1016/j.future.2018.03.007.
[21] H. Alasmary *et al.*, "Analyzing and Detecting Emerging Internet of Things Malware: A Graph-based Approach," *IEEE Internet Things J.*, vol. 4662, no. c, pp. 1-1, 2019, doi: 10.1109/jiot.2019.2925929.
[22] E. M. Dovom, A. Azmoodeh, A. Dehghantanha, D. E. Newton, R. M. Parizi, and H. Karimipour, "Fuzzy pattern tree for edge malware detection and categorization in IoT," *J. Syst. Archit.*, vol. 97, pp. 1-7, 2019, doi: 10.1016/j.sysarc.2019.01.017.
[23] H. Le, Q. Ngo, and V. Le, "Iot Botnet Detection Using System Call Graphs and One-Class CNN Classification," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 10, pp. 937-942, 2019, doi: 10.35940/ijitee.j9091.0881019.
[24] J. Su, V. Danilo Vasconcellos, S. Prasad, S. Daniele, Y. Feng, and K. Sakurai, "Lightweight Classification of IoT Malware Based on Image Recognition," *Proc. - Int. Comput. Softw. Appl. Conf.*, vol. 2, pp. 664-669, 2018, doi: 10.1109/COMPSAC.2018.10315.
[25] J. Liu, S. Liu, and S. Zhang, "Detection of IoT botnet based on deep learning," *Chinese Control Conf. CCC*, vol. 2019–July, no. 1, pp. 8381-8385, 2019, doi: 10.23919/ChiCC.2019.8866088.
[26] E. M. B. Karbab, M. Debbabi, A. Derhab, and D. Mouheb, "MalDozer: Automatic framework for android malware detection using deep learning," *DFRWS 2018 EU - Proc. 5th Annu. DFRWS Eur.*, vol. 24, pp. S48-S59, 2018, doi: 10.1016/j.diin.2018.01.007.
[27] K. D. T. Nguyen, T. M. Tuan, S. H. Le, A. P. Viet, M. Ogawa, and N. Le Minh, "Comparison of Three Deep Learning-based Approaches for IoT Malware Detection," *Proc. 2018 10th Int. Conf. Knowl. Syst. Eng. KSE 2018*, pp. 382-387, 2018, doi: 10.1109/KSE.2018.8573374.
[28] A. Azmoodeh, A. Dehghantanha, and K.-K. R. Choo, "Robust Malware Detection for Internet of (Battlefield) Things Devices Using Deep Eigenspace Learning," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 1, pp. 88-95, 2018, doi: 10.1109/tsusc.2018.2809665.
[29] C. Tzagkarakis, N. Petroulakis, and S. Ioannidis, "Botnet attack detection at the IoT edge based on sparse representation," *Glob. IoT Summit, GIoTS 2019 - Proc.*, pp. 1-6, 2019, doi: 10.1109/GIOTS.2019.8766388.
[30] C. Dietz *et al.*, "IoT-Botnet Detection and Isolation by Access Routers," *Proc. 2018 9th Int. Conf. Netw. Futur. NOF 2018*, pp. 88-95, 2018, doi: 10.1109/NOF.2018.8598138.
[31] Y. Meidan *et al.*, "N-BaIoT-Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 12-22, 2018, doi: 10.1109/MPRV.2018.03367731.
[32] C. D. McDermott, F. Majdani, and A. V. Petrovski, "Botnet Detection in the Internet of Things using Deep Learning Approaches," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2018-July, pp. 1-8, 2018, doi: 10.1109/IJCNN.2018.8489489.
[33] H. Bahsi, S. Nomm, and F. B. La Torre, "Dimensionality Reduction for Machine Learning Based IoT Botnet Detection," *2018 15th Int. Conf. Control. Autom. Robot. Vision, ICARCV 2018*, pp. 1857-1862, 2018, doi: 10.1109/ICARCV.2018.8581205.
[34] A. Abusnaina *et al.*, "Examining Adversarial Learning against Graph-based IoT Malware Detection Systems," 2019.
[35] A. Guerra-Manzanares, H. Bahsi, and S. Nomm, "Hybrid feature selection models for machine learning based botnet detection in IoT networks," *Proc. - 2019 Int. Conf. Cyberworlds, CW 2019*, pp. 324-327, 2019, doi: 10.1109/CW.2019.00059.
[36] A. P. Namanya, I. U. Awan, J. P. Disso, and M. Younas, "Similarity hash based scoring of portable executable files for efficient malware detection in IoT," *Futur. Gener. Comput. Syst.*, 2019, doi: 10.1016/j.future.2019.04.044.
[37] H. S. Ham, H. H. Kim, M. S. Kim, and M. J. Choi, "Linear SVM-based Android malware detection for reliable IoT services," *J. Appl. Math.*, vol. 2014, 2014, doi: 10.1155/2014/594501.

[38] Z. Ren, H. Wu, Q. Ning, I. Hussain, and B. Chen, "End-to-end malware detection for android IoT devices using deep learning," *Ad Hoc Networks*, vol. 101, p. 102098, 2020, doi: 10.1016/j.adhoc.2020.102098.
[39] H. T. Nguyen, Q. D. Ngo, D. H. Nguyen, and V. H. Le, "PSI-rooted subgraph: A novel feature for IoT botnet detection using classifier algorithms," *ICT Express*, no. xxxx, 2020, doi: 10.1016/j.icte.2019.12.001.
[40] W. Jung, H. Zhao, M. Sun, and G. Zhou, "IoT botnet detection via power consumption modeling," *Smart Heal.*, vol. 15, p. 100103, 2020, doi: 10.1016/j.smhl.2019.100103.
[41] H. El Merabet and A. Hajraoui, "A survey of malware detection techniques based on machine learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 1, pp. 366-373, 2019, doi: 10.14569/IJACSA.2019.0100148.
[42] D. Ucci, L. Aniello, and R. Baldoni, "Survey of machine learning techniques for malware analysis," *Comput. Secur.*, vol. 81, pp. 123-147, 2019, doi: 10.1016/j.cose.2018.11.001.
[43] R. Sihwail, K. Omar, and K. A. Z. Ariffin, "A survey on malware analysis techniques: Static, dynamic, hybrid and memory analysis," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 8, no. 4-2, pp. 1662-1671, 2018, doi: 10.18517/ijaseit.8.4-2.6827.
[44] J. Landage and M. Wankhade, "Malware and Malware Detection Techniques: A Survey," *Int. J. Eng. Res. Technol.*, vol. 2, no. 12, pp. 61-68, 2013.
[45] S. Talukder, "Tools and Techniques for Malware Detection and Analysis," *ArXiv*, 2020.
[46] K. Diaz-Chito, F. J. Ferri, and A. Hernández-Sabaté, "An overview of incremental feature extraction methods based on linear subspaces," *Knowledge-Based Syst.*, vol. 145, pp. 1-14, 2018, doi: 10.1016/j.knosys.2018.01.020.
[47] M. F. Rafique, M. Ali, J. Y. K. Qureshi, Aqsa Saeed, Asifullah Khan, and and A. M. Mirza, "Malware Classification using Deep Learning based Feature Extraction and Wrapper based Feature Selection Technique," *ArXiv*, pp. 1-20, 2019.
[48] M. A. Al-Garadi, A. Mohamed, A. Al-Ali, X. Du, and M. Guizani, "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security," *ACM Comput. Surv.*, vol. 5, pp. 1-42, 2018.
[49] S. M. Tahsien, H. Karimipour, and P. Spachos, "Machine learning based solutions for security of Internet of Things (IoT): A survey," *J. Netw. Comput. Appl.*, vol. 161, no. April, 2020, doi: 10.1016/j.jnca.2020.102630.
[50] M. Egele, T. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamic malware-analysis techniques and tools," *ACM Comput. Surv.*, vol. 44, no. 2, pp. 1-49, 2012, doi: 10.1145/2089125.2089126.

BIOGRAPHIES OF AUTHORS

Susanto received his master degree in Computer Science from Universitas Bina Darma Palembang, South Sumatera, Indonesia. Currently he is a PhD candidate at Faculty of Engineering, Universitas Sriwijaya. He is currently a senior lecturer at Faculty of Computer, Universitas Bina Insan, Indonesia. His research interests include cryptography, information technology, information security, and network security.

M. Agus Syamsul Arifin received his master degree in Computer Science from Universitas Bina Darma Palembang, South Sumatera, Indonesia. Currently he is a PhD candidate at Faculty of Engineering, Universitas Sriwijaya. He is currently a senior lecturer at Faculty of Computer, Universitas Bina Insan, Indonesia. His research interests include information technology, information security, and network security.

Deris Stiawan received the PhD degree in Computer Engineering from Universiti Teknologi Malaysia, Malaysia. He is currently an Associate Professor at Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya. His research interests include computer network, Intrusion Detection/Prevention System, and heterogeneous network

Mohd Yazid Idris is an Associate Professor at School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia. He obtained his M.Sc and Ph.D. in the area of Software Engineering, and Information Technology (IT) Security in 1998 and 2008 respectively. In software engineering, he focuses on the research of designing and development of mobile and telecommunication software. His main research activity in IT security is in the area of Intrusion Prevention and Detection (IPD).

Rahmat Budiarto received B.Sc. degree from Bandung Institute of Technology in 1986, M.Eng. and Dr.Eng. in Computer Science from Nagoya Institute of Technology in 1995 and 1998, respectively. Currently, he is a full Professor at College of Computer Science and IT, Albaha University, Saudi Arabia. His research interests include intelligent systems, brain modeling, IPv6, network security, Wireless sensor networks, and MANETs.