RIGMS Testbed for IoT Cybersecurity Research Using Machine Learning Based Approach

Hanlin CAI
Cyber Range Lab, Fuzhou University, China
hanlin.cai@ieee.org

Abstract

This paper proposes a real-time intelligent garbage monitoring system (RIGMS) testbed for IoT cybersecurity research. The testbed is established by realistic devices in the physical world, which is a stage in the process of municipal waste disposal. Multiple-mix-attacks were conducted based on the testbed. During the attack scenarios, the network activities were analyzed, and the traffic features were extracted to design a representative RIGMS dataset for training and verifying the authenticity of the machine learning based models. In this paper, five advanced ML models were utilized to detect the cyber-attacks. Experiment results verified the feasibility of implementing learning based models to detect multiple-mix-attacks.

1. Introduction

Nowadays, the internet of things (IoT) has grown into a global giant, grabbing hold of every facet of our everyday lives and benefiting people with its unrestricted intelligent technologies. However, due to the fast-growing demand for IoT facilities without matched access control, IoT systems can be vulnerable to collapse in the face of massive attacks. According to recent surveys, more than 186% increase in the records of large-scale IoT attacks has been observed in the past three years [1-3].

As an alternative to conventional security schemes, machine learning (ML) based schemes appear to be a promising option for IoT security. Many existing research has proposed some state-of-the-art learning based models to address the specific single attacks and achieved notable performances [4-9]. However, an advanced invader can perform various attacks in a collaborative manner, called multiple-mix-attacks, leading to more serious damage [10, 11]. To resolve this new challenge, a realistic and representative dataset for IoT multiple-mix-attacks is necessary for training and verifying the authenticity of the learning based models.

In this paper, a real-time intelligent garbage monitoring system (RIGMS) testbed was established to investigate the feasibility of implementing learning based models to detect cyber-attacks. This testbed was constructed using facilities conducted in the physical world. Multiple-mix-attacks were deployed on the testbed to understand how IoT devices behaved in the network when infected. Also, the network activities were captured for traffic analysis, and the behaviors and features of the traffic were extracted to build a new RIGMS dataset. Furthermore, five advanced ML models were utilized for training and testing based on the proposed dataset. Ultimately, experiment results were analyzed to evaluate the effectiveness of implementing ML models to detect IoT multiple-mix-attacks.

2. Literature Review

This section goes through the key concepts of this work, including IoT security attacks and learning based detection. Furthermore, many related research works have been reviewed, and gaps in existing knowledge have been identified.

2.1. IoT Security Attack

IoT Security Attack is one of the most crucial challenges in realistic IoT systems, which has gained extensive attention in recent IoT research. Most recently, many literature studies have discussed specific IoT security attacks and corresponding defensive schemes. Vishwakarma and Jain[2] discussed the principle of botnets and malware being deployed to Distributed DoS and proposed a dependable DDoS defense technique to recognize the security gaps. Arshad, et al.[1] proposed the THC-RPL scheme to detect malicious Sybil nodes in the IoT network, which significantly reduced the packet loss rate while maintaining lower power consumption. Besides, there were many papers focusing on the potential security challenges of IoT systems [12-14].

However, most of these works addressed only specific solutions to single security attacks. Still, few considerations are given to an advanced invader, who may collaboratively perform multiple attacks from different sources [15, 16], as shown in Fig.1. To advance the research in this domain, we propose a realistic and representative dataset for multiple-mix-attacks which can be used to train and evaluate relevant attack detection strategies.

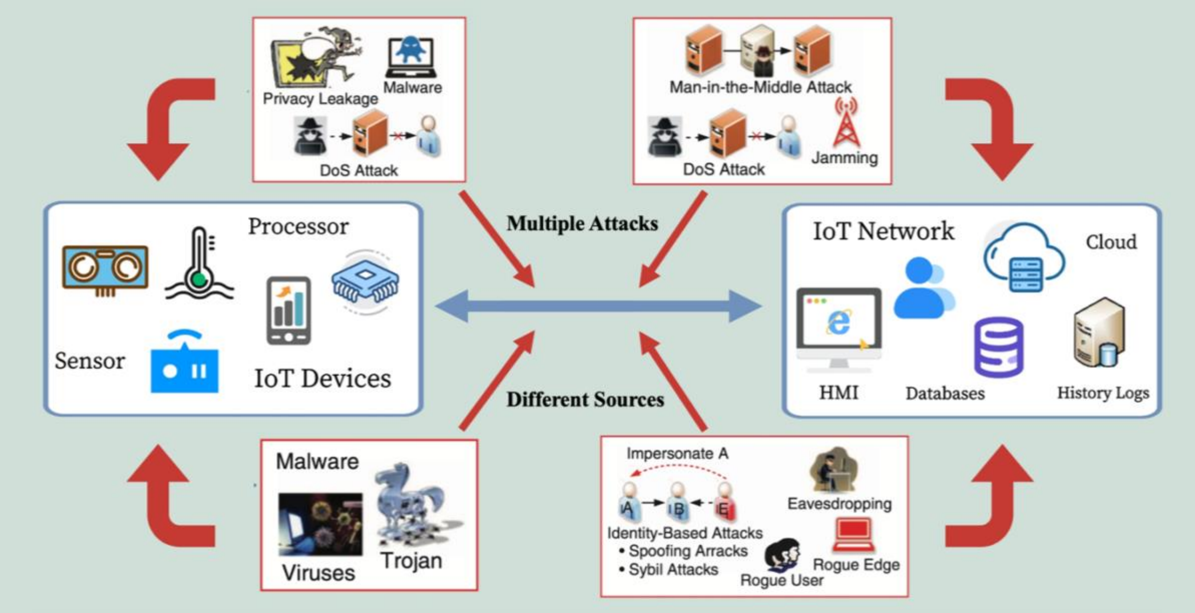


FIGURE 1 AN ILLUSTRATION OF IOT MULTIPLE-MIX-ATTACKS

2.2. Learning Based Detection

Learning Based Detection is based on ML techniques that optimize model performance and effectiveness through learning existing datasets and previous experience [14]. In brief, learning based detection can be classified into four domains, as shown in Fig.2.

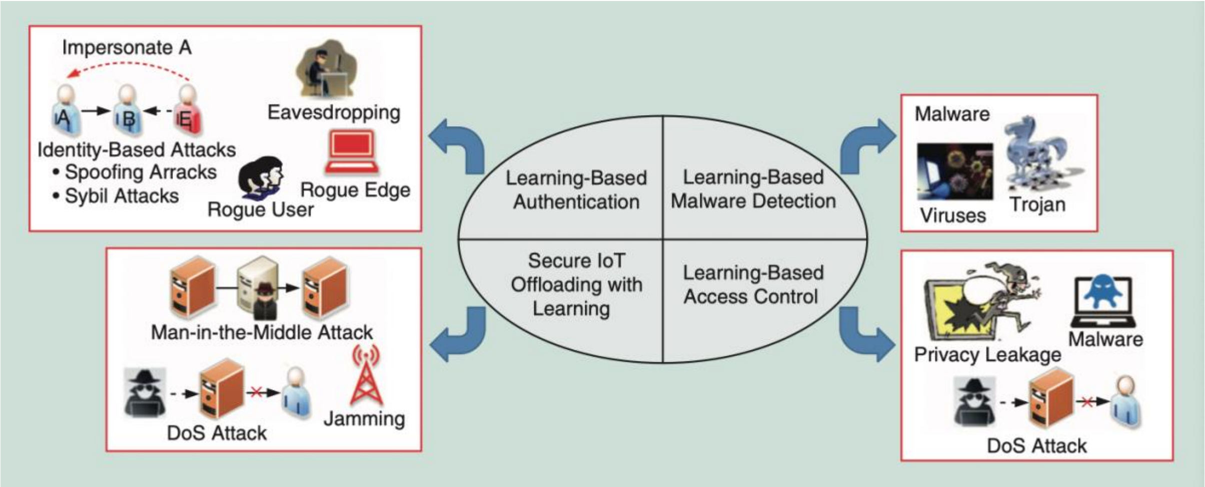
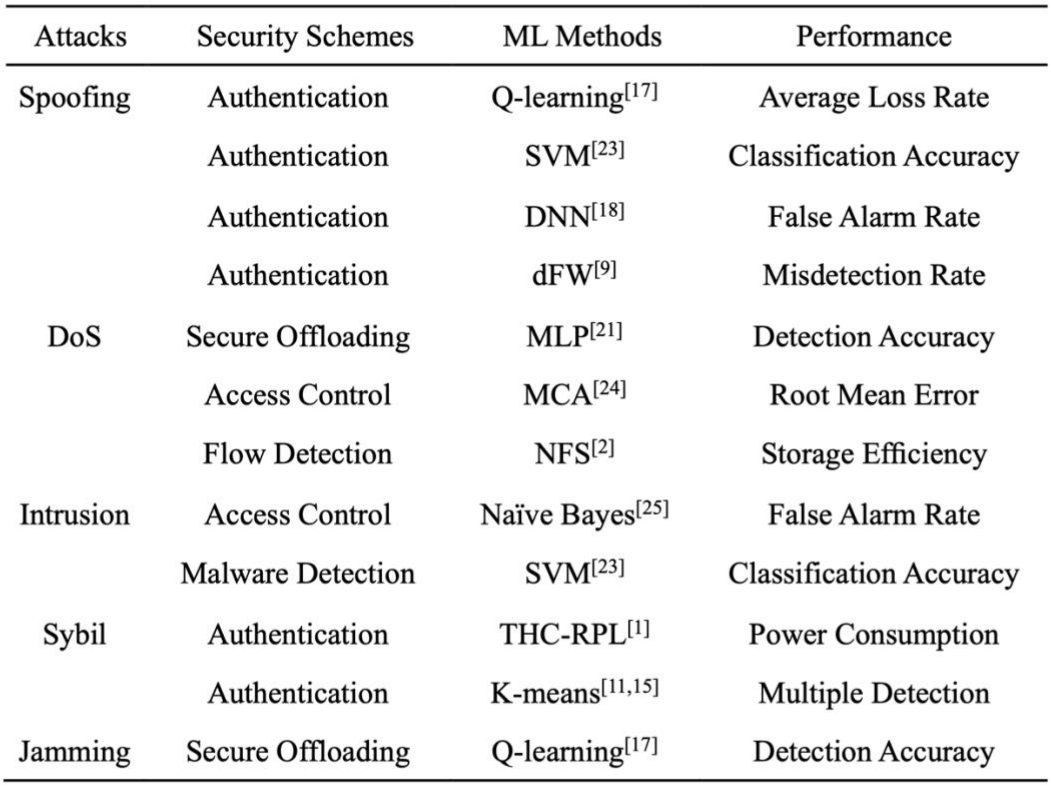


FIGURE 2 IOT SECURITY ATTACKS AND CORRESPONDING DEFENSIVE METHODS

Authentication methods improve the cybersecurity of IoT systems to distinguish benign nodes from malicious nodes and effectively prevent identity-based attacks, such as eavesdropping and Sybil attacks [17, 18]. Secure offloading enables IoT facilities to utilize the resources of the cloud and server for computationally intensive and time-critical tasks [19]. Access control can effectively block unauthorized users from accessing devices and resources on the IoT network [20, 21]. Malware detection techniques help IoT networks defend against malicious malware, such as rootkits, Trojans and viruses [22, 23].

FIGURE 2 AN ILLUSTRATION OF IOT MULTIPLE-MIX-ATTACKS



Attacks	Security Schemes	ML Methods	Performance
Spoofing	Authentication	Q-learning[17]	Average Loss Rate
	Authentication	SVM[23]	Classification Accuracy
	Authentication	DNN[18]	False Alarm Rate
	Authentication	dFW[9]	Misdetection Rate
DoS	Secure Offloading	MLP[21]	Detection Accuracy
	Access Control	MCA[24]	Root Mean Error
	Flow Detection	NFS[2]	Storage Efficiency
Intrusion	Access Control	Naïve Bayes[25]	False Alarm Rate
	Malware Detection	SVM[23]	Classification Accuracy
Sybil	Authentication	THC-RPL[1]	Power Consumption
	Authentication	K-means[11,15]	Multiple Detection
Jamming	Secure Offloading	Q-learning[17]	Detection Accuracy

As illustrated in Table.1, many papers have proposed various advanced learning based detection methods to prevent specific IoT attacks and improve cybersecurity [1, 2, 9, 11, 15, 17, 18, 21, 23-25]. The Q-learning schemes proposed by Xiao, et al.[17] performed well in the face of both spoofing and jamming attacks, which reached an accuracy of 96.7% and a precision of 95.8%. The SVM model proposed by Ozay, et al.[23] could effectively identify and detect intrusion and spoofing attacks from the same sources, which achieved a satisfactory accuracy of 99.86%.

Despite their efforts, most of the solutions suggested by these studies seldom focus on sophisticated attacks from different sources. At the same time, multiple-mix-attacks were absent in most of the training and testing process due to the lack of representative datasets. In this paper, to investigate the feasibility of implementing ML models to detect multiple-mix-attacks, five advanced ML models were trained based on the representative RIGMS dataset.

2.3. Related Literature Works

IoT attacks are constantly evolving and developing to breach security mechanisms. Therefore, utilizing advanced security schemes in IoT systems is paramount to detecting and preventing unknown attacks. In this sense, the design of representative datasets based on physical IoT devices and realistic IoT network advances the research in this domain. Most recently, some related literature works proposed various testbeds and corresponding datasets, as compared in Table.2[26-31].

TABLE 2 A SUMMARY OF RELATED WORKS

Testbeds & Datasets	Benign Records	Malicious Records	Key Words
HBB (2014)[26]	null	77,054	HTTP flooding method, without benign traffic
IRC-centric (2006)[27]	null	227,784	Real-world botnet, without benign traffic
SCADA (2019)[28]	427,934	6,622,054	Realistic IIoT environments, online deployment
Botloader (2014)[29]	7,417,070	29,662,465	Mix of two-way traffic, with large-scale DDoS
DDoSTB (2017)[30]	2,218,761	557,646	Complex and advanced hardware systems
IoT-23 (2020)[31]	30,854,735	298,490,308	Abundant, captures in controlled environment
Proposed RIGMS (2022)	2,070,012	97,086	Multiple-mix-attacks, with detailed features

To build the representative datasets, numerous testbeds were constructed. To begin with, the HTTP-based botnet (HBB) testbed proposed by Alomari, et al.[26] relied on an advanced server, to analyze a real-time HTTP-based botnet attacks. In their work, a complete Web-access-log infected by a botnet was first suggested for researchers. Besides, the IRC-centric testbed designed by Livadas, et al.[27] made use of a real-world botnet called “Kaiten,” which could launch the DDoS attack to the victim host. However, both of HBB-testbed and IRC-centric testbed were not included benign traffic, which led to deviations from the real network environment. Contrary to their approaches, our testbed takes advantage of Ostinato[32] and EXPLIoT[33] to generate benign and malicious traffic simultaneously.

Focusing on cyber-vulnerability assessment and susceptibilities countering, Zolanvari, et al.[28] implemented their SCADA system testbed, in order to design a ML model to efficiently detect malicious Intrusion Detection System (ICS) network traffic. In their work, a notable contribution was the proposed evaluation strategy of online deployment since most related work only presents the performance of models during the training and test phase. Whereas, as mentioned by Zolanvari, et al.[28], their dataset did not contain enough features to describe the relationship between benign activities and malicious activities, which caused the dataset inconvenient to be used by other researchers. Also, similar shortcomings appeared in the Botloader testbed proposed by Bhatia, et al.[29] and DDoSTB suggested by Behal and Kumar[30]. In our work, the proposed RIGMS datasets provided detailed information about the traffic features, which provides more convenient and useful training and testing samples for research.

3. RIGMS Testbed & RIGMS Dataset

This section starts with introduction of our previous work and the novelty of this paper. Further, the design of the proposed RIGMS dataset and the methodology of corresponding feature extraction have been detailedly explained.

3.1. Realistic Testbed Establishment

A realistic IoT testbed is required to explore real cyberattacks and collect representative datasets, including benign and malicious traffic. In this sense, a real-time intelligent garbage monitoring system (RIGMS) testbed is proposed to simulate the real-world environment as closely as possible. Figure 3 shows the main components of the RIGMS testbed, and Table 3 shows brief descriptions of the devices equipped in the system. We built the RIGMS testbed environment at the IoT Cyber Range Lab of Fuzhou University, China. The more detailed establishment process and related parameters of each device can be found in Ref. [34, 35].

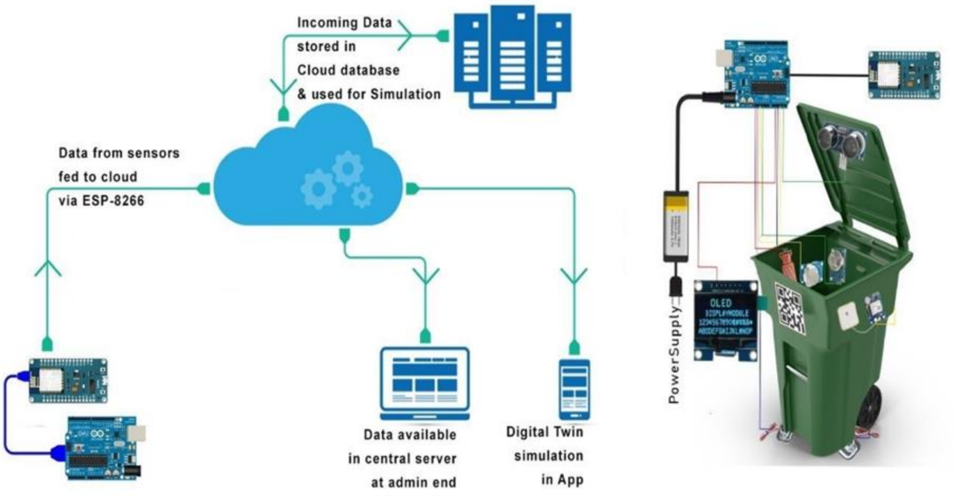

Incoming Data stored in Cloud database & used for Simulation
Data from sensors fed to cloud via ESP-8266
Data available in central server at admin end
Digital Twin simulation in App
OLED
PowerSupply

FIGURE 3 THE MAIN COMPONENTS OF THE RIGMS TESTBED

As illustrated in Figure 3 and Table 3, the RIGMS testbed is equipped with three different sensors: Temperature Sensor (TS), Humidity Sensor (HS), and Ultrasonic Sensor (US). The sensor cluster monitors the temperature, humidity and remaining space of the garbage bins in real-time. Once the state value reaches the threshold level or runs out of the designated range, the related sensor will send alarm signals to the ESP Module. Integrated in the NodeMCU IoT platform, the ESP Module is utilized for data transmission and simple edge computing. When ESP Module receives the warning signals from sensors, it transmits the signal to the Ali Cloud server. Further, the server would display the warning information on the back-end dynamic web page, including the position, accident conditions, and specific status of the garbage bin.

In this paper, we have improved our testbed by adding the Human Machine Interface, which can be used by administrators to control and monitor the testbed in real-time. Also, our testbed takes advantage of Ostinato[32], EXPLIoT[33] and Kali Linux[36] to generate benign and malicious activities simultaneously. Third, Firmware Analysis and Comparison Tool (FACT)[37] is utilized for traffic analysis and Argus Tool [38] is used for feature extraction and forensic analysis. Finally, the proposed RIGMS testbed adopts the Modbus communication protocol [39], which is one of the most popular protocols in the area of Industrial IoT.

TABLE 3 DESCRIPTION OF THE DEVICES IN RIGMS TESTBED

Devices	Descriptions
Temperature Sensor	Monitors the real-time temperature in the bin. When the temperature reaches the threshold level (50°C), the sensor sends alarm signal to ESP Module.
Humidity Sensor	Monitors the real-time humidity level in the bin. When the humidity is out of the designated range (21% to 86%), the sensor sends alarm signal to ESP Module.
Ultrasonic Sensor	Monitors the remaining space in the bin. When the space is less than the threshold level (15%), the sensor sends alarm signal to ESP Module.
ESP Module	ESP module is integrated in NodeMCU IoT-platform, which used to information transmission and edge computing. More details can be found in Ref [34].
Ali-cloud platform (Cloud)	The advanced server, Ali-cloud platform, is utilized for data processing, data storage and complex computing. Additionally, the java-web platform is implemented in the back end of the Ali-cloud for data visualization.
Human Machine Interface (HMI)	Used by the administrator to control and monitor the RIGMS testbed in real-time. Besides, HMI allow the user to modify the parameter and interact with the system.
Digital Twin Application (DTAPP)	The mobile Android application is implemented as Digital Twin of the RIGMS testbed, which accurately reflect and control some physical devices in the system.

3.2. Attack Scenarios Analysis

To the best of our knowledge, no research has focused on a realistic IoT system testbed for multiple-mix-attacks. In this paper, we utilized Ostinato[32] for conducting continuous benign traffic. To ensure the IoT attacks come from different sources, EXPLIoT[33] and Kali Linux[36] were simultaneously employed to generate malicious cyber-attack activities. It should be noted that we deliberately designed our dataset to be unbalanced. The percentage of malicious activities was less than 6.00%, which could make the testbed environment as similar as possible to the real-world networks [40, 41]. In the attack scenarios, twelve types of traffic activities, including Distributed Denial-of-Service (DDoS), SQL Injection and Jamming, were generated from four separate sources (Testbed, Ostinato, EXPLIoT and Kali Linux). Table 4 illustrates the statistical information of the traffic activities in attack scenarios.

TABLE 4 STATISTICAL INFORMATION OF THE CAPTURED TRAFFIC

Type of the Traffic	Source	Percentage	Type of the Traffic	Source	Percentage
Benign Traffic	Ostinato	91.50	Device Identification	*Multiple	1.091
Malicious Traffic	*Multiple	4.480	PortScanner Traffic	EXPLIoT	0.182
DDoS Traffic	*Multiple	2.085	Okiru Traffic	EXPLIoT	0.096
SQL Injection Traffic	Kali Linux	0.795	C&C Traffic	EXPLIoT	0.084
Jamming Traffic	Kali Linux	0.042	Information theft Traffic	EXPLIoT	0.065
Command Injection	Kali Linux	0.040	Other Normal Traffic	Testbed	4.020

The asterisked (*) means traffic activities come from multiple sources.

As shown in Table 4, most of the benign traffic is generated by Ostinato, at the same time, there is also some normal traffic from the testbed itself. Moreover, the multiple-mix-attacks scenarios are conducted by EXPLIoT and Kali Linux. All of the related data generated in the attack scenarios, as well as the benign traffic are collected and recorded by FACT[37], where the recorded average data rate is 890 kbit/s and the average packet size is 294.6 bytes.

3.3. Traffic Feature Extraction & Label Definition

Once the traffic activities are collected and recorded, further work is to extract the traffic feature which can be used to distinguish malicious activities from benign or normal activities. As far as feature extraction is concerned, the researcher in Ref.[28] suggested a valuable method to select traffic features for ML model training. Also, the related work proposed in Ref.[40] demonstrated the effectiveness of Argus Tool[38] for feature capture. Inspired by their works, in this paper, the continuous variation between benign and malicious activities was analyzed using Argus Tool. Based on the literature works and our analysis, the feature extracted for our dataset is shown in Table 5. Ultimately, all the data was labeled either benign traffic (0) or malicious traffic (1).

TABLE 5 FEATURES EXTRACTED FOR PROPOSED DATASET

Features	Types	Features	Types
Total Packets (ToPks)	Integer	Destination Packets (DsPks)	Integer
Total Bytes (ToBys)	Float	Destination Bytes (DsBys)	Float
Total Load (ToLod)	Float	Destination Load (DsLod)	Float
Total Rate (ToRat)	Float	Destination Rate (DsRat)	Float
Total Loss (ToLos)	Float	Destination Loss (DsLos)	Float
Total Port (ToPot)	Integer	Destination Port (DsPot)	Integer
Source Packets (SoPks)	Integer	Mean Flow (mean)	Float
Source Bytes (SoBys)	Float	Source Jitter (SoJtr)	Float
Source Load (SoLod)	Float	Destination Jitter (DsJtr)	Float
Source Rate (SoRat)	Float	Source Interpackets (SoIpk)	Float
Source Loss (SoLos)	Float	Destination Interpackets (DsIpk)	Float
Source Port (SoPot)	Integer	Total Percent Loss (TpLos)	Float

4. Learning Based Models & Performance Evaluation

This section describes the learning based models used in the paper and the evaluation methodology utilized to measure the performance of the models.

4.1. Experiment Analysis

As shown in Figure 4, some cyber-attacks (e.g. DDoS, SQL Injection) can be easily reconnoitered while some other attack activities, such as PortScanner and Data Theft, are difficult to detect. In this case, the rule-based mechanisms suggested in Ref.[42, 43] would fail since the feature of the traffic is too subtle to recognize. On the other hand, the related works proposed in Ref. [28, 40] demonstrated the feasibility of utilizing ML for subtle feature detection.

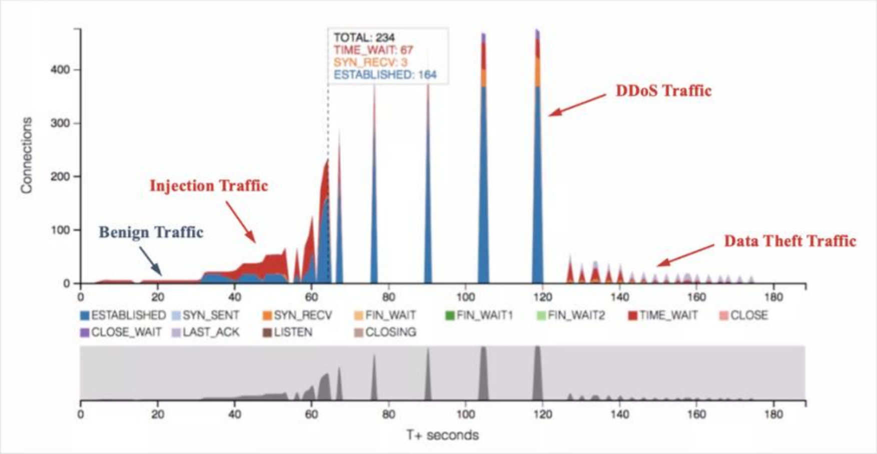


FIGURE 4 VARIOUS TRAFFIC ACTIVITIES

In this case study, our RIGMS dataset includes a total of 2167098 traffic samples for training and testing, where 80% of data was used for model training and 20% for model testing. Figure 5 illustrates the flow of the experiment, where the input is the 24 selected features, while the output is either benign traffic (0) or malicious traffic (1), as motioned in Section 3.

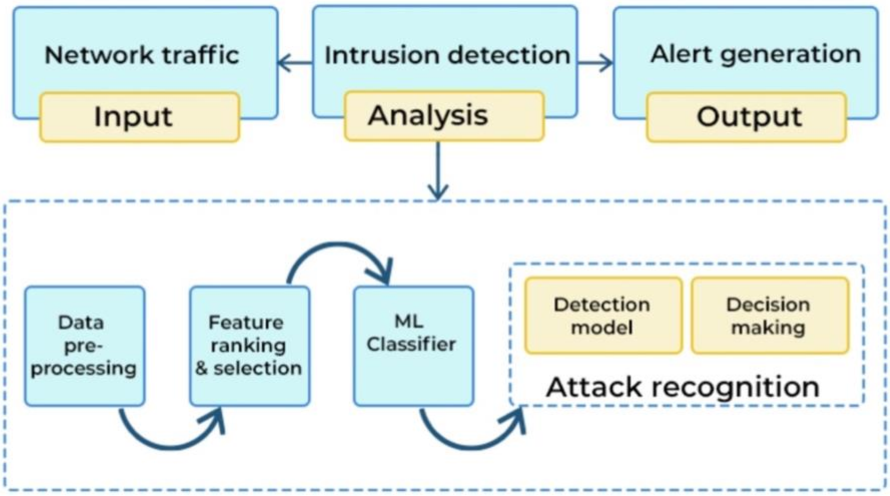


FIGURE 5 THE PROCESS OF EXPERIMENT

4.2. Model Training & Testing

As mentioned in Section 2, this paper aims to investigate the feasibility of implementing ML models to detect multiple-mix-attacks. Therefore, five advanced learning based models are utilized for intrusion detection and attack recognition (Fig 5). Table 6 shows the performance metrics of these five ML models proposed by related literature reviews [12, 14]. Based on the related research [17, 18, 23, 24, 44], this paper also utilized the Keras Library [45] and scikit-learning library [46] to implement these models, which were trained and tested over the proposed RIGMS dataset. Next, the experiment results are compared and discussed in section 5.

TABLE 6 A SUMMARY OF PERFORMANCE METRICS

Ref.	ML Model	Accuracy	Precision	Recall	F1-Score
[17]	Q-learning	96.7%	95.8%	98.9%	97.3%
[18]	NN	99.03%	97.89%	100%	98.9%
[23]	SVM	99.86%	96.71%	99.23%	97.95%
[24]	MCA	97.2%	96.4%	95.7	96.1%
[44]	Random Forest	98.5%	96.7%	95.1%	95.9%

NN: **N**eural **N**etwork SVM: **S**upport **V**ector **M**achine MCA: **M**ultivariate **C**orrelation **A**nalysis

4.3. Performance Evaluation

After model training and testing, the next step is performance evaluation. Generally speaking, the experiment result of training and testing is usually assessed by metrics derived from the confusion matrix [4, 47], as shown in Table 7.

TABLE 7 CONFUSION MATRIX FOR EVALUATION

Traffic Data	Classified as Benign	Classified as Malicious
Benign Sample	True Negative (TN)	False Negative (FN)
Malicious Sample	False Positive (FP)	True Positive (TP)

As mentioned in Section 3, to simulate the real-world network environment as similar as possible, the dataset was deliberately designed to be unbalanced, where the amount of benign traffic is far more than malicious traffic. In this case, the benign sample is dominant in number, leading to biased results. So, the Accuracy metric is not representative and reliable to evaluate the performance of the ML models in this scenario. In order to avoid a biased analysis, credible metrics (False Alarm Rate and Un-Detection Rate) have been suggested [48]. Therefore, in addition to Accuracy, the FAR and UND metrics also be used in the performance evaluation. Table 8 illustrates several evaluation metrics and their corresponding formulas.

TABLE 8 EVALUATION METRICS AND EXPLANATION

Evaluation Metrics	Corresponding Formula	Index
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}\times 100\%$	(1)
False Alarm Rate (FAR)	$\frac{FP}{FP+TN}\times 100\%$	(2)
Un-Detection Rate (UND)	$\frac{FN}{FN+TP}\times 100\%$	(3)

TN: This represents the number of benign samples correctly classified as benign.

TP: This represents the number of malicious samples correctly classified as malicious.

FN: This represents the number of malicious samples incorrectly classified as benign.

FP: This represents the number of benign samples incorrectly classified as malicious.

5. Result & Analysis

The section presents the numerical results of the experiment described in Section 4, as well as the discussion based on comparison and analysis.

5.1. Numerical Results

Table 9 shows the numerical results for the three performance metrics of the five learning based models. In brief, considering the training evaluation, Random Forest (RF) model has the best accuracy, FAR and UND performance. As for the testing evaluation, Support Vector Machine (SVM) model gets the best results compared to other ML models.

TABLE 9 NUMERICAL RESULTS OF TRAINING AND TESTING

Evaluation		Learning Based Models				
Metrics		QL	NN	SVM	MCA	RF
Training (80%)	Accuracy	98.24%	98.89%	99.03%	98.17%	**99.41%**
	FAR	0.06%	0.03%	0.01%	0.06%	**0.01%**
	UND	0.87%	0.86%	0.41%	0.98%	**0.22%**
Testing (20%)	Accuracy	83.60%	96.62%	**98.01%**	97.63%	96.02%
	FAR	2.47%	0.19%	**0.09%**	0.15%	0.24%
	UND	56.73%	2.94%	**1.28%**	2.46%	3.71%

5.2. Comparison & Analysis

Figure 6 shows the performance metric for Accuracy of the five ML models. Random Forest model has the best accuracy of 99.41% in the training results, while SVM model achieves the highest performance of 98.01% in the testing results. Considering the accuracy metric, the difference between training and testing is slight, except for the Q-learning model, which indicates that the Q-learning model does not perform well in realistic testing. Besides, as motioned in Section 5, the Accuracy metric is not representative and reliable to measure the performance of the ML models in an unbalanced evaluation scenario. Therefore, other credible metrics are required to compare the difference between the five ML models.

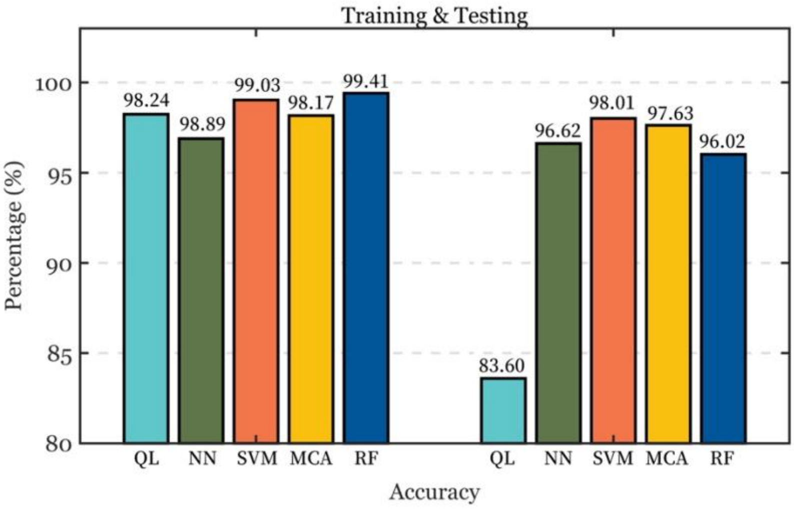


FIGURE 6 ACCURACY PERFORMANCE METRIC

Figure 7 illustrates the comparison for FAR metric of five ML models, where the FAR represents the percentage of the benign traffic misclassified as malicious. Compared to the other models, Random Forest gets the best FAR of 0.01% in training experiments, while SVM model conducts the best performance of 0.09% in testing experiments. However, considering the difference between training and testing scenarios, the Random Forest model experienced a relatively large rise from 0.01% to 0.24%, showing potential instability in realistic testing.

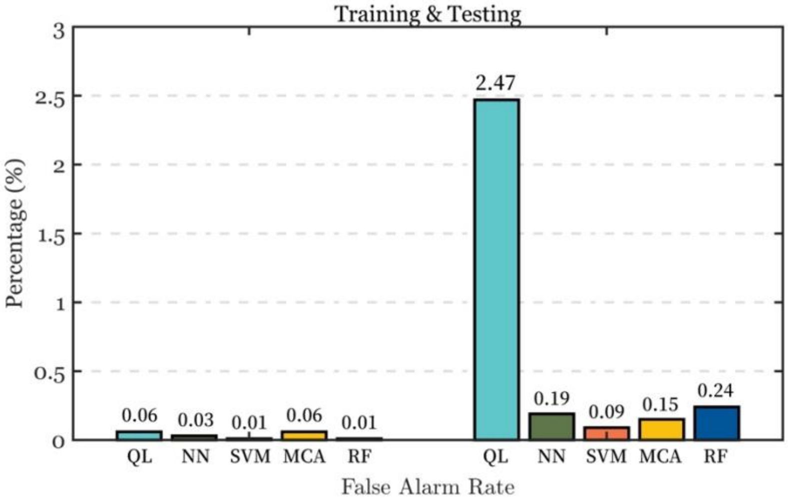


FIGURE 7 FAR PERFORMANCE METRIC (LOWER IS BETTER)

Figure 7 illustrates the comparison for the UND metric of five ML models, where the UND represents the percentage of malicious traffic that is incorrectly classified as benign traffic. The SVM model performs best (1.28%), followed by the Multivariate Correlation Analysis (MCA, 2.46%), Neural Network (NN, 2.94%), and Random Forest (RF, 3.71%), while the Q-learning model gets an unsatisfactory result of 56.73%.

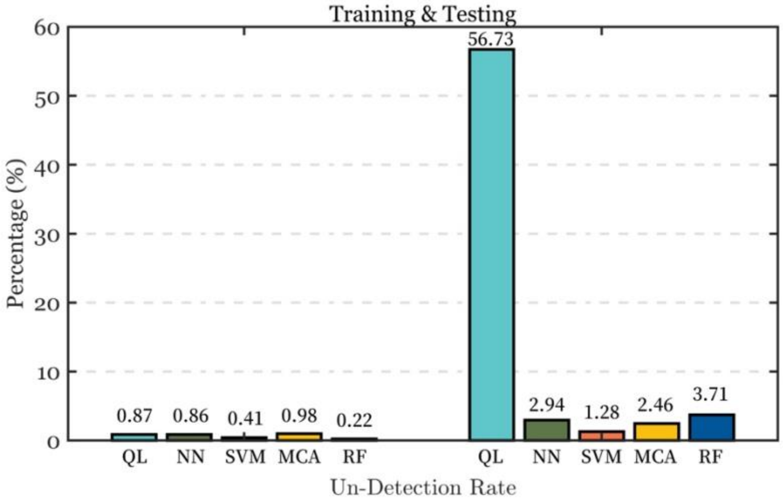


FIGURE 8 UND PERFORMANCE METRIC (LOWER IS BETTER)

As illustrated in Figure 6-8, the performance metrics for Accuracy, FAM and UND of five ML models have been explicitly compared. The evaluation analysis indicates that Support

Vector Machine (SVM) model performs best compared to other ML models. In addition, the Multivariate Correlation Analysis (MCA) model also shows good performance and stability during the training and testing experiment. Therefore, SVM and MCA models can be utilized to resolve the challenge of multiple-mix-attacks in a realistic network environment.

6. Conclusion

In this paper, a real-time intelligent garbage monitoring system (RIGMS) testbed was established for IoT cybersecurity research. Based on the proposed RIGMS testbed, a realistic and representative RIGMS dataset for multiple-mix-attacks was designed. The detailed features have been extracted for training and testing purposes. Experiments have been conducted to evaluate the effectiveness of implementing ML models to detect cyber-attacks. Three performance metrics were utilized for performance measurement: Accuracy, FAR and UND. Results show that SVM and MCA models performed best compared to other models. The feasibility of implementing learning based models to detect multiple-mix-attacks has been verified in this work. As for future plan, more attack scenarios will be generated and analyzed. In addition, more ML models will be utilized for experiments and verifications.

Acknowledgement

First and foremost, the authors gratefully acknowledge Dr. Pietro Liò and Dr. Chin-Hong Wong for their generous guidance and support during this research. Also, the authors would like to thank Mr. Kehai Qiu for his constructive suggestions. Lastly, none of this work would have been possible without the support of the Cyber Range Lab at Fuzhou University, China.

References

[1] Arshad D, Asim M, et al. THC-RPL: A lightweight Trust-enabled routing in RPL-based IoT networks against Sybil attack [J]. PLoS One, 2022, 17(7): e0271277.

[2] Vishwakarma R, Jain A K. A survey of DDoS attacking techniques and defence mechanisms in the IoT network [J]. Telecommunication Systems, 2019, 73(1): 3-25.

[3] Alladi T, Chamola V, et al. Consumer IoT: Security Vulnerability Case Studies and Solutions [J]. IEEE Consumer Electronics Magazine, 2020, 9(2): 17-25.

[4] Strecker S, Dave R, et al. A modern analysis of aging machine learning based IOT cybersecurity methods [J]. arXiv preprint arXiv:211007832, 2021.

[5] Dr S S, Dr. Abul B, et al. Hybrid Intrusion Detection System for Internet of Things (IoT) [J]. Journal of ISMAC, 2020, 2(4): 190-199.

[6] Bao J, Hamdaoui B, et al. Iot device type identification using hybrid deep learning approach for increased iot security; proceedings of the 2020 International Wireless Communications and Mobile Computing (IWCMC), F, 2020 [C]. IEEE.

[7] Adi E, Anwar A, et al. Machine learning and data analytics for the IoT [J]. Neural

Computing and Applications, 2020, 32(20): 16205-16233.

[8] Ioannou C, Vassiliou V. Classifying Security Attacks in IoT Networks Using Supervised Learning [Z]. 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS). 2019: 652-658.10.1109/dcoss.2019.00118

[9] Xiao L, Wan X, et al. PHY-Layer Authentication With Multiple Landmarks With Reduced Overhead [J]. IEEE Transactions on Wireless Communications, 2018, 17(3): 1676-1687.

[10] Ahmad R, Alsmadi I. Machine learning approaches to IoT security: A systematic literature review [J]. Internet of Things, 2021, 14.

[11] Ma Z, Liu L, et al. Towards multiple-mix-attack detection via consensus-based trust management in IoT networks [J]. Computers & Security, 2020, 96.

[12] Hussain F, Hussain R, et al. Machine learning in IoT security: Current solutions and future challenges [J]. IEEE Communications Surveys & Tutorials, 2020, 22(3): 1686-1721.

[13] Mohamad Noor M b, Hassan W H. Current research on Internet of Things (IoT) security: A survey [J]. Computer Networks, 2019, 148: 283-294.

[14] Xiao L, Wan X, et al. IoT Security Techniques Based on Machine Learning: How Do IoT Devices Use AI to Enhance Security? [J]. IEEE Signal Processing Magazine, 2018, 35(5): 41-49.

[15] Ma Z, Liu L, et al. DCONST: Detection of Multiple-Mix-Attack Malicious Nodes Using Consensus-Based Trust in IoT Networks [M]. Information Security and Privacy. 2020: 247-267.

[16] Liu L, Ma Z, et al. Detection of multiple-mix-attack malicious nodes using perceptron-based trust in IoT networks [J]. Future Generation Computer Systems, 2019, 101: 865-879.

[17] Xiao L, Li Y, et al. PHY-layer spoofing detection with reinforcement learning in wireless networks [J]. IEEE Transactions on Vehicular Technology, 2016, 65(12): 10037-10047.

[18] Shi C, Liu J, et al. Smart User Authentication through Actuation of Daily Activities Leveraging WiFi-enabled IoT [Z]. Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing. Chennai, India; Association for Computing Machinery. 2017: Article 5.10.1145/3084041.3084061

[19] Xiao L, Xie C, et al. A mobile offloading game against smart attacks [J]. IEEE Access, 2016, 4: 2281-2291.

[20] Alsheikh M A, Lin S, et al. Machine learning in wireless sensor networks: Algorithms, strategies, and applications [J]. IEEE Communications Surveys & Tutorials, 2014, 16(4): 1996-2018.

[21] Kulkarni R V, Venayagamoorthy G K. Neural network based secure media access

control protocol for wireless sensor networks; proceedings of the 2009 International Joint Conference on Neural Networks, F 14-19 June 2009, 2009 [C].

[22] Xiao L, Li Y, et al. Cloud-based malware detection game for mobile devices with offloading [J]. IEEE Transactions on Mobile Computing, 2017, 16(10): 2742-2750.

[23] Ozay M, Esnaola I, et al. Machine Learning Methods for Attack Detection in the Smart Grid [J]. IEEE Transactions on Neural Networks and Learning Systems, 2016, 27(8): 1773-1786.

[24] Tan Z, Jamdagni A, et al. A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis [J]. IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 447-456.

[25] Alsheikh M A, Lin S, et al. Machine Learning in Wireless Sensor Networks: Algorithms, Strategies, and Applications [J]. IEEE Communications Surveys & Tutorials, 2014, 16(4): 1996-2018.

[26] Alomari E, Manickam S, et al. Design, deployment and use of HTTP-based botnet (HBB) testbed; proceedings of the 16th International Conference on Advanced Communication Technology, F, 2014 [C]. IEEE.

[27] Livadas C, Walsh R, et al. Usilng machine learning technliques to identify botnet traffic; proceedings of the Proceedings 2006 31st IEEE conference on local computer networks, F, 2006 [C]. IEEE.

[28] Zolanvari M, Teixeira M A, et al. Machine Learning-Based Network Vulnerability Analysis of Industrial Internet of Things [J]. IEEE Internet of Things Journal, 2019, 6(4): 6822-6834.

[29] Bhatia S, Schmidt D, et al. A framework for generating realistic traffic for Distributed Denial-of-Service attacks and Flash Events [J]. Computers & Security, 2014, 40: 95-107.

[30] Behal S, Kumar K. Detection of DDoS attacks and flash events using information theory metrics–An empirical investigation [J]. Computer Communications, 2017, 103: 18-28.

[31] Sebastian Garcia A P, & Maria Jose Erquiaga. (2020). IoT-23: A labeled dataset with malicious and benign IoT network traffic (Version 1.0.0) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.4743746.

[32] Ostinato, Traffic Generator for Network Engineers, URL https://ostinato.org/ [Z]. 2022

[33] EXPLIoT, Internet of Things Security Testing and Exploitation Framework, URL https://gitlab.com/expliot_framework/expliot [Z]. 2022

[34] NodeMCU IoT Platform, URL https://www.nodemcu.com/index_en.html [Z]. 2022

[35] Hanlin C, Jiaqi H, et al. An IoT Garbage Monitoring System for Effective Garbage Management [Z]. have been accepted by 2022 Computer Engineering, Network, and Intelligent Multimedia (CENIM). 2022

[36] Kali Linux Penetration Testing Distribution, URL https://www.kali.org/ [Z]. 2022

[37] Firmware Analysis and Comparison Tool (FACT), URL https://fkie-cad.github.io/FACT_core/ [Z]. 2022

[38] Argus Tool , URL https://openargus.org/ [Z]. 2022

[39] Modbus communication protocol, URL https://www.modbustools.com/modbus.html [Z]. 2022

[40] Koroniotis N, Moustafa N, et al. Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset [J]. Future Generation Computer Systems, 2019, 100: 779-796.

[41] Teixeira M, Salman T, et al. SCADA System Testbed for Cybersecurity Research Using Machine Learning Approach [J]. Future Internet, 2018, 10(8).

[42] Dayanandam G, Rao T, et al. DDoS attacks—analysis and prevention [M]. Innovations in Computer Science and Engineering. Springer. 2019: 1-10.

[43] Rehman S U, Manickam S. Rule-based mechanism to detect Denial of Service (DoS) attacks on Duplicate Address Detection process in IPv6 link local communication; proceedings of the 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), F, 2015 [C]. IEEE.

[44] Narudin F A, Feizollah A, et al. Evaluation of machine learning classifiers for mobile malware detection [J]. Soft Computing, 2014, 20(1): 343-357.

[45] Keras Library, URL https://keras.io/ [Z]. 2022

[46] Scikit-Learn Library, URL https://scikit-learn.org/stable/ [Z]. 2022

[47] Sokolova M, Lapalme G. A systematic analysis of performance measures for classification tasks [J]. Information processing & management, 2009, 45(4): 427-437.

[48] Buda M, Maki A, et al. A systematic study of the class imbalance problem in convolutional neural networks [J]. Neural networks, 2018, 106: 249-259.