Design and Implementation of IoT DDoS Attacks Detection System based on Machine Learning

Yi-Wen Chen, Jang-Ping Sheu, Yung-Ching Kuo, and Nguyen Van Cuong

Institute of Communication Engineering, National Tsing Hua University

Hsinchu, 30013, Taiwan

ujikolp753@gapp.nthu.edu.tw, sheujp@cs.nthu.edu.tw, cludbuster@gmail.com, cuongnv243@gmail.com

***Abstract*—DDoS attacks often happen in cloud servers and cause a devastating problem. However, an increasing number of Internet of Things (IoT) devices makes us not ignore the influence of large-scale DDoS attacks from IoT devices. In this paper, we propose a machine learning-based on a multi-layer IoT DDoS attack detection system, including IoT devices, IoT gateways, SDN switches, and cloud servers. Firstly, we build eight smart poles with various sensors on our campus and collect sensor data as our datasets through wireless networks or wired networks. Next, we extract the features based on DDoS attack types. The feature selection can result in high accuracy DDoS attack detection in the real IoT environment. The experimental results show that our multi-layer DDoS detection system can accurately detect DDoS attacks. And the SDN controller can block venomous devices effectively according to blacklists from the results of our IoT DDoS attacks detection system.**

Keywords- Distributed Denial of Service; Internet of Things; Machine Learning; Software Defined Networking

I. INTRODUCTION

Distributed Denial of Service (DDoS) attacks has become a severe problem in network security [1]. Unlike simple DoS attacks, DDoS attacks take advantage of a great number of compromised hosts and aim to exhaust network resources as fast as possible. Once the network is disrupted, it cannot provide any services to legitimate users [2]. There are various types of DDoS attacks [2, 3] such as Internet Control Message Protocol (ICMP) flood, SYN (Synchronize) flood, User Datagram Protocol (UDP) flood, and Domain Name Server (DNS) reflection attack. DDoS attacks are hard to defend against because of their distributed nature. Most attacks are very easily launched and difficult to trace back to the true hackers. Therefore, it causes a tremendous problem for users in the networks.

Nearly 50 billion IoT devices are used by 2020 [4]. However, most of the IoT devices are insecure and easy to be infected with malware. The objective of this paper is to detect the DDoS attack in IoT. Because of the rapid development of IoT devices, IoT security is a hot topic in recent years. The literature in [5-6] discusses the attack type in IoT devices and provides protocols with encryption algorithms. The authors in [7-9] generate dynamic rules by the software-defined networking (SDN) controller to mitigate DDoS attacks. Recently, machine-learning algorithms have been extensively applied in various fields and achieved high accuracy. In [3, 10], the authors combine SDN with machine-learning methods to defend DDoS attacks. In [2, 11], the authors propose a machine learning-based source side DDoS detection system in a cloud computing environment. The authors in [12-16] propose machine learning-based IoT DDoS attacks detection. The literature [2, 11, 12, 16] focus on analyzing the feature of DDoS attacks to extract the features precisely.

In this paper, we focus on the DDoS attacks for IoT systems, including IoT devices, IoT gateways, SDN switches, and cloud servers. In IoT applications, there are IoT devices with various sensors connected to the IoT gateways via different network protocols such as Ethernet, Wi-Fi, Bluetooth, ZigBee, and LoRa. The Wi-Fi APs also provide Internet connectivity to the pedestrian who can access to the campus internet. The IoT gateways aggregate sensor data and transmit it to cloud servers through SDN switches. We propose a method to detect DDoS attacks by machine learning techniques in IoT infrastructure, and then block these attacks by the SDN controller. Because there exists a significant difference between IoT data packets and those data packets generated by mobile devices of the user. We divide packets into two categories. One is sensor data, and the other one is network data from pedestrians. We extract the features from two kinds of packets and label them as normal packets or DDoS attack packets. The labeled packets are trained by decision trees and saved as two training models that one is transmitted to the IoT gateways, and the other one is transmitted to the SDN controller for on-line detection. Our contributions are:

1)We implement a heterogeneous gateway to collect sensor data and to authenticate the IoT devices and then propose a machine learning-based DDoS attack detection for malicious sensor data in the gateway.

2)We analyze the features of different kinds of DDoS attacks, including ICMP flood, SYN flood, and UDP flood, to improve the accuracy of DDoS attacks detection. We take advantage of the blacklists from the SDN controller to block the malicious packets.

3)We propose decision trees for off-line training and on-line DDoS attacks detection in the IoT system. An alert message will be sent to the administrator when malicious packets are detected.

The rest of this paper is organized as follows. In Section II, we review the related works. In Section III, we propose a method of the DDoS attacks detection based on machine learning and blocking DDoS attacks by the SDN controller. In Section IV, we launch DDoS attacks in our IoT environment and present the experimental results. Finally, we conclude in Section V.

This work was supported by the Ministry of Science and Technology under Grant MOST 106-2221-E-007-019-MY3 and Hsinchu Science Park, under Grant 108A25B, Taiwan, R.O.C.

978-1-7281-4355-2/20/$31.00 ©2020 IEEE

II. Related Works

For a better understanding of how security and DDoS attack detection works in the IoT environment, we review some related works on the authentication for IoT devices and machine learning-based DDoS attack detection as follows.

A. Authentication for IoT devices

Authentication is indispensable for network access, especially in those wireless IoT devices. The authors in [5, 6] depict the types of DDoS attacks in IoT devices and the encryption algorithms of LoRa, Bluetooth, and ZigBee. The encryption algorithms of the LoRaWAN [5] are proposed to ensure the security and provide end devices and the gateway with encryption capability. The DDoS attacks in the ZigBee and Bluetooth network [6] are Mirai IoT botnet attacks that the malware infects and controls the end devices to attack the target server. ZigBee end devices could be easily added to the coordinator in the discover state that is the lack of security. In DDoS attacks, stealing or altering information makes the devices paralyzed and the resources unavailable to the victim. We can observe the device behavior to judge whether attacks happen or not. ZigBee and Bluetooth have the encryption standards to make the messages more secure through AES-CRT, AES-CBC-MAC, or AES-CCM. However, the encryption and decryption of the messages may cause latency problems and be accompanied by computing, storage, and energy limitations.

B. DDoS Attacks Detection

Traditionally, most DDoS attacks are analyzed in cloud servers. The literature in [2] combines the features of different type DDoS attacks to train and defend attacks from the source side in the cloud. In DNS attacks, they monitor the inbound and outbound traffic to calculate the inbound/outbound packets ratio to detect the DNS reflection attack. For ICMP flood, if there are a large number of ICMP packets in a short time, it means that DDoS attacks have probably occurred. For the SYN flood, the author uses the SYN/ACK ratio as the SYN flood indicator. When the SYN/ACK ratio reaches high, it is abnormal. The authors in [10] present a faster and accurate DDoS attack detection system based on the C4.5 algorithm and signature detection techniques in the cloud computing environment.

For DDoS attacks detection in IoT, the authors in [16] select stateless features, packet size, inter-packet interval, protocol, bandwidth, and the count of distinct destination IP addresses to detect DDoS attacks from IoT devices. The literature [17] proposes a multi-level DDoS mitigation framework and provides a solution to prevent and detect DDoS attacks for every layer. In [12], the authors combine new features with old features for machine learning-based DDoS attacks to make an early detection. The paper in [13] proposes a classification-based DDoS attack detection in IoT. The authors in [14] propose deep learning models, including MLP (Multilayer Perceptron), CNN (Convolutional Neural Network), and LSTM (Long Short Term Memory), to detect whether the packet is anomalous or not in the IoT network. The literature [15] aims to analyze the botnet attacks through the SVM algorithm in IoT and focuses on the protocols, HTTP, TCP, and ICMP.

For DDoS attacks detection in SDN, the author in [7] proposes SDNShield, an NFV-based defense framework that joins the strengths of software switches and a two-stage filtering algorithm to protect the centralized controller. The proposed model in [8] devises a statistical solution that evaluates an entropy-based security scheme to enhance the SDN security and mitigates the DDoS attacks. The proposed framework in [9] is a multi-layer framework that consists of a controller pool containing main SD-IoT controllers, SD-IoT switches integrated with an IoT gateway, and IoT devices. The framework uses the threshold value of the cosine similarity of the vectors to judge whether a DDoS attack has occurred and blocked the DDoS attack at the source site.

The machine learning techniques make the DDoS defense systems in SDN efficiently. The paper in [3] proposes an authorization module to check whether the controller can send requests to the server or not and a machine learning-based prediction module to detect potential DDoS attacks. If the learning model detects that the packet is abnormal, the controller will increase a new rule to SDN switches to block the attacker's IP address. In [10], the authors proposed a smart DDoS mitigation system, including two modules for information collection and DDoS mitigation in the application plane.

Although there are some existing works focus on DDoS attack detection and mitigation in IoT devices, most of them use available datasets for testing. In contrast to previous works, we establish a real integrated IoT system, collect the data from real IoT devices, and launch a real DDoS attack. Then, we extract the desired DDoS attack features from our collected data. Finally, we combine machine learning techniques with the SDN controller to mitigate the DDoS attacks in the IoT system to verify the feasibility of DDoS detection in the IoT system.

III. A Multi-Layer DDoS Detection Methodologies in IoT

In this section, we present our four-layer IoT system architecture, including IoT devices, IoT gateways, SDN switches, and cloud servers, as shown in Fig. 1. There are two kinds of data in the IoT device layer, including sensor data from IoT sensors and network data from Wi-Fi by the user equipment. The packets of sensor data are transmitted by various protocols, including Bluetooth, ZigBee, LoRa, and Wi-Fi. IoT gateways are responsible for aggregating sensor data from various sensors.

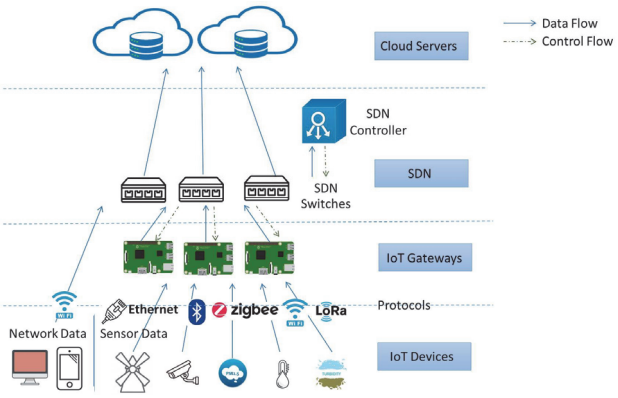

Fig. 1. A scenario of the IoT system architecture

In this work, we implement a multi-layer DDoS attack detection system which aims to the IoT environment. We divide the sources of DDoS attacks into two types in IoT. One is the sensor data flood from sensors, and the other one is the network data flood from the user equipment. We propose the following steps to detect and block DDoS attacks. First, we

design the IoT security authentication that has lower latency than encryption standards in the IoT device layer. Second, we select the features from the packets capturing in the IoT gateways and the SDN switches according to the attack types. Third, we introduce the data flow, control flow, and training process in our system architecture. Finally, we implement our DDoS detection system to detect and block the DDoS attacks by the machine learning technique in the SDN environment.

A. IoT Security Authentication

First of all, we implement the IoT authentication mechanism for non-IP based devices, such as Bluetooth, ZigBee, and MQTT. Every Bluetooth module has a unique address that makes a specific device to be identified. We create a whitelist that consists of the equipment ID, protocol, and MAC address of the Bluetooth device to authenticate that the device is legal to transmit data. If the device address is not in the whitelist, the IoT gateway will reject the connection.

ZigBee certifies end devices through setting a secret key and PAN ID as the first line of the device authentication. Since ZigBee devices connect to the coordinator without authentication in the discovery state, we set a secret key and a PAN ID to all ZigBee modules to stop the illegal devices from sending packets to the IoT gateway. The second line of ZigBee module authentication is similar to Bluetooth authentication. Additionally, the coordinator in the IoT gateway still has a whitelist composed of the equipment ID, protocol, and MAC address. It can discard the illegal packets from the attacker even if the hacker knows the secret key and PAN ID.

In LoRaWAN, we use ABP (Activation by Personalization) mode as our encryption way since every device has its unique NwkSKey and AppSKey [5]. An attacker has no idea about the key encryption of other devices, even if he knows one of the key encryptions. In this way, our IoT security authentication not only makes sure the authentication of the device but also saves the time of encrypting and decrypting packets. IoT security authentication is the first line of DDoS attack defense in our IoT system.

The MQTT protocol is based on the principle of publishing messages and subscribing to topics. If hackers know what the topic is, they can publish any message to the topic without authentication, and that is very dangerous. We solve this problem by disabling the function of allowing an anonymous publisher for MQTT and setting a username and a password for MQTT authentication.

B. Features Selection

The second line of DDoS attack defense is using decision trees to detect whether packets are abnormal. Before we train the sensor data and network data, we need to select the useful features including the packet length, timestamp, protocol, source IP, source MAC address, destination IP, destination MAC address. Additionally, the total number of packets is also important. The type of DDoS attacks also influences our features selection. According to the previous works, we do not launch DNS reflection attacks since our environment is the IP-based network. We consider three common DDoS attack types, ICMP flood, SYN flood, and UDP flood for network data, and sensor data flood for sensor data.

In the sensor data flood, hackers infect IoT devices and take control of IoT sensors to send data continuously. Sensors send data with a fixed frequency in the normal situation. Therefore, we can select the timestamp of sensor data and calculate the total number of packets for a fixed period to identify whether the IoT devices are in a normal situation.

It is easy to know that there are ICMP attacks since the ICMP packets are normally few in the network. Therefore, it means that a device is attacked by the hacker when the device receives a lot of ICMP packets in a short time. Depending on the ICMP packet state, we can get more information to analyze. Ping of death (POD) is a kind of ICMP attacks. Ping is used to inspect the Internet connection through echo requests and echo replies. To generate a dataset for ICMP, we ping the target IP as our ICMP attack packets and make the types of ICMP attack numerical. We select the type of ICMP and the number of ICMP as our features to detect ICMP flood. SYN flood takes advantage of the characteristic of TCP and does not send an ACK to the client in the final step. Thus, we can observe the flags of TCP and accumulate the number of SYN and ACK. We can know that there are SYN flood attacks when the SYN/ACK ratio reaches high.

The attack mode of the UDP flood is purer than the SYN flood. Unlike the complicated three-way handshake, UDP flood attacks send many big size UDP packets to a specific IP address and port. Thus, we select the length of UDP, UDP stream, UDP source port, and UDP destination port as our features to detect the UDP flood attacks. Next, we present the rules of the SDN controller to defend DDoS attacks.

C. System Architecture

A multi-layer DDoS detection architecture for IoT is proposed as shown in Fig. 2. In the IoT system, there are four device layers, including the IoT devices, IoT gateways, SDN switches, and cloud servers. In the machine learning engine, it separately collects sensor data and network data for off-line learning and deploys the training models for on-line DDoS attack detection.

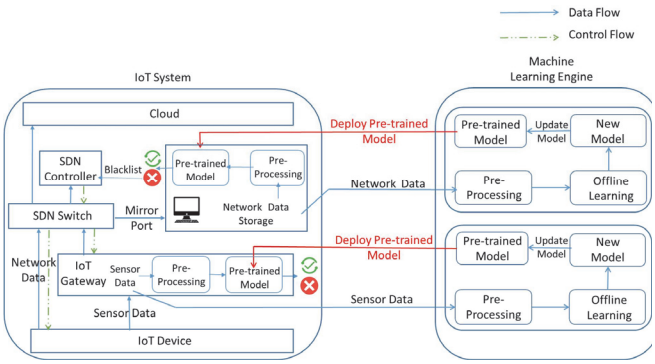

Fig. 2. A multi-layer DDoS detection architecture for IoT

All sensor data will be transmitted to IoT gateways. Then the gateway forwards data to the cloud server via SDN switches. The IoT gateway captures sensor data packets for pre-processing to detect the sensor data flood attacks by the pre-trained model of sensor data. Once the gateway detects the sensor data flood attacks, the IoT gateway notifies the administrator. The latest sensor data gathering in the IoT gateway will be sent to the machine learning engine for learning and update a new model of sensor data.

Since network data is transmitted by Wi-Fi and directly sent to the SDN switch, we cannot capture the network data

in the IoT gateway. Therefore, we take advantage of port mirroring that the switch sends a copy of all network data seen on one port to another, where the network data can be stored in a server to analyze the received packets. The server will send a blacklist to the SDN controller when the pre-trained model of network data detects the abnormal packets. Then the latest network data is transmitted to the machine learning engine to update the training model of network data. Meanwhile, the SDN controller setups the rules for SDN switches and the IoT end devices to block DDoS attacks.

In our IoT system, there are three lines of DDoS attack defense. The first line of DDoS attack defense is the IoT security authentication for protocols in IoT devices. We propose whitelists for ZigBee and Bluetooth to block the illegal devices. There is a set of the username and the password for MQTT to prevent illegal users from publishing messages to the IoT topics. The second line of DDoS attack defense is using machine learning technology to detect DDoS attacks. According to the literature in [16], the authors compare different machine learning technologies to detect DDoS attacks for consumer IoT devices. The results show that the decision tree performs a high accuracy of over 99%.

Therefore, we take the decision tree for our IoT DDoS attack detection. Firstly, we make the features of machine learning datasets numerical since the original datasets cannot be trained directly. According to the features, trees split data into two or more sub-nodes and hope that the splitting results can return the highest information gain. When the pre-trained models of decision trees detect the DDoS attacks, the MAC address and IP address of the malicious packets are sent to the SDN controller as the blacklist and notify the administrator. The final line is the rules of the SDN controller. The SDN controller blocks IP addresses and MAC addresses of malevolent devices based on the blacklists. However, the SDN controller crashes and is unable to block the virulent devices in time when the IoT system is attacked by large-scale DDoS attacks in a short period. Therefore, we use bandwidth control to manage the egress and ingress bandwidth of all IoT devices so that we have enough time to handle DDoS attacks and prevent the IoT system from paralysis.

IV. Implementation

In this section, we implement our proposed multi-layer DDoS attack detection model and show the experimental results. In the following subsections, we introduce our experimental environment, captured packets of sensor data, and network data. First, we introduce the sensor distribution, network transmission for protocols, and the sense frequency of sensors. Second, we depict the data collection for sensor data in IoT gateway and network data through Wi-Fi and present the implementation of DDoS attack detection for sensor data flood and network flood, including ICMP flood, SYN flood, and UDP flood. Third, we discuss the detection results of different types of DDoS attacks. Finally, we utilize bandwidth control to restrict the network traffic and block IP addresses and MAC addresses of spiteful devices.

A. Experimental Environment

To create a more realistic experimental environment, we construct eight smart poles on our campus as shown in Fig. 3. Each smart pole is equipped with an LED lamp, an access point (AP), camera, smart signage, communication box, and an equipment box. A Raspberry Pi 3 with the ability to communicate with Ethernet, Bluetooth, ZigBee (through I^2C interface), and Wi-Fi devices as a heterogeneous gateway and various sensors are placed in a communication box. The equipments like power module, ethernet switch, and Industrial PC (IPC) for edge computing, which need more space, are sheltered in the equipment box. The network architecture is shown in Fig. 4, where SPi represents smart pole i for $1 \leq i \leq 8$. Sensors in the communication box keep collecting temperature, humidity, PM2.5, wind speed data of the environment and voltage, current information of the lamp itself. The video stream and image from the cameras will be preprocessed at the IPC. The sensor distribution is shown in Table I. The sensing frequency of most sensors is 10 minutes, except the anemometer, which sensing frequency is 5 minutes. Smart poles change the sensing frequency of the voltage and the current from 10 minutes to 5 minutes if they turn on at night.

Fig. 3. The real experimental environment on our campus

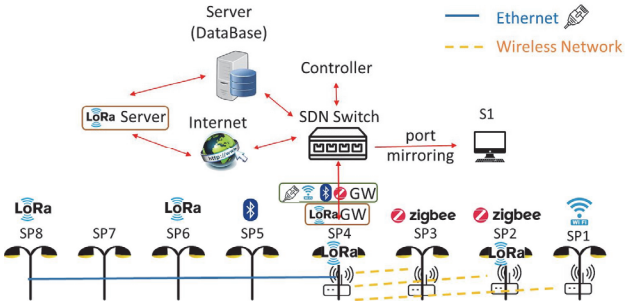

Fig. 4. The network architecture of the smart poles

TABLE I.
Sensor distribution

Smart pole / Sensor	SP1	SP2	SP3	SP4	SP5	SP6	SP7	SP8
Temperature, Humidity					✓			
PM2.5							✓	
Anemometer	✓							
Camera	✓	✓					✓	✓
voltage, current	✓	✓	✓	✓	✓	✓	✓	✓

There is a Raspberry Pi 3 on each smart pole responsible for collecting the sensor data. In our IoT system, sensor data can be transmitted through Wi-Fi, Bluetooth, ZigBee, and LoRa. Each Wi-Fi access point installed on smart poles SP1~SP4 makes these poles able to access the Internet without backbone connections, while smart poles SP4 ~ SP8 can use

Ethernet to transmit and receive packets. Pedestrians can also connect to the Wi-Fi AP and access the Internet. Besides, there are ZigBee on SP2 and SP3, Bluetooth on SP5, and LoRa on SP2, SP4, SP6, and SP8.

We design and implement a heterogeneous gateway on SP4 to aggregate the sensor data from other smart poles which transmit data by various protocols. SP4 has two gateways to receive sensor data from other smart poles. One is the heterogeneous gateway, including Ethernet, Wi-Fi, Bluetooth, and ZigBee, and the other is the LoRa gateway. The IoT gateways are shown in Fig. 5.

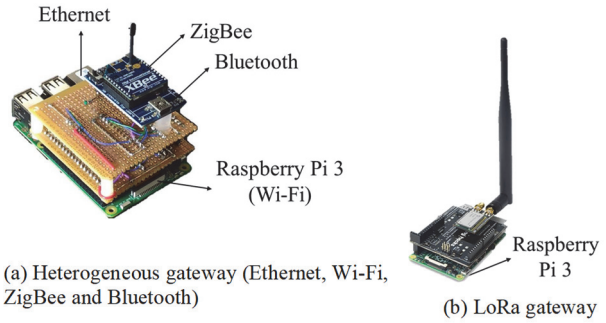

Fig. 5. The IoT Gateways

The data flow of sensor data is shown in Fig. 6. Since Ethernet and Wi-Fi are IP-based, the sensor data can be sent directly. However, the sensor data transmitted via Bluetooth, ZigBee, and LoRa need a specific interface in the IoT gateway. The heterogeneous gateway will aggregate packets when it receives sensor data. After aggregating sensor data on SP4, SP4 will send sensor data to the SDN switch. The SDN switch receives these packets and then transfers them to the database. Although the LoRa gateway also sends packets to the SDN switch, the packets are transmitted to the LoRa server, and then sent to the database finally.

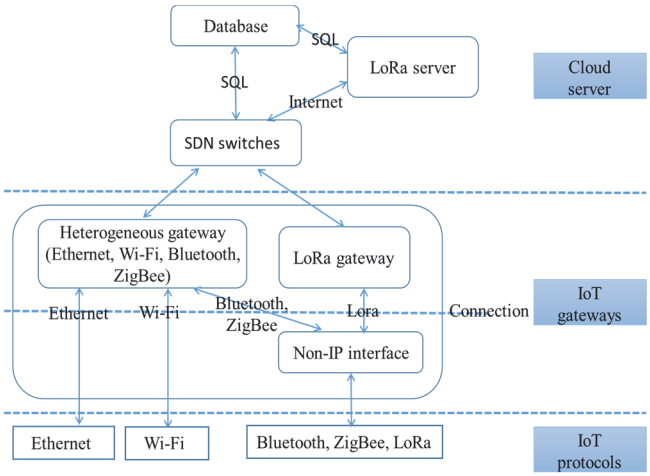

Fig. 6. The data flow of sensor data

B. Data Collection and DDoS Attacks

Our dataset is available in [18]. In our experiment, we separately collect sensor data in the gateway and network data by port mirroring. In detail, we collect sensor data from SP4 for 73 minutes and then launch sensor data flood from three smart poles, including SP1, SP5, and SP8. Each smart pole uses six types of sensors to attack SP4 with various periods, including 0.05 seconds, 0.2 seconds, 0.3 seconds, one second, one minute, and two minutes. The duration of the attack frequencies is set to one minute, one minute, one minute, 2 minutes, 35 minutes, and 30 minutes, respectively. Staggering the different attack frequency is convenient for us to label abnormal packets. There are 31,521 packets in total that 4864 packets are sensor data flood attacks, and others are normal packets. According to the time of sensor data flood attack, we label these attack packets as abnormal packets. We collect 22,350 abnormal packets to training our models.

Before our model training, we first digitize the IP address. For example, 192.168.200.211 will be encoded into 192168200. Since all the packets cannot have every feature of the data set, for example, ICMP packets cannot have the characteristics of TCP packets so that each packet may have missing values in different features. The performance of the training model is affected by the missing values. Therefore, we perform data preprocessing based on the functional characteristics of the network packets. After prepocessing the missing value of the data set, we normalize the distribution of the data set into Gaussian distribution to reduce the effect of outlier data on the performance of the model.

C. Detection Results

After collecting the dataset, we split our collected data into training samples (70%) and testing samples (30%), and use four metrics to evaluate our performance. These metrics are composed of true positive (TP), false positive (FP), false negative (FN), and true negative (TN) defined in the confusion matrix. FP and FN indicate false alarms and misses, respectively. The four metrics, *Accuracy, Precision, Recall,* and *F1-score,* are defined as follows:

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

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

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

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Accuracy shows the ratio of the number of correct detections to the total number of detections. *Precision* shows what a portion of alerts is a true alert. *Recall* shows the part of the DDoS attacks that are detected. *F1-score* is a standard to balance FP and FN. The higher *F1-score* indicates better performance for a model. We show the results in Table II. In our experiment, decision trees achieve over 97% in both accuracy and F1-score. Since we train data for off-line learning, our on-line detection models take a short while only for 0.36 seconds to detect the DDoS attack packet.

TABLE II
DETECTION RESULTS OF DECISION TREES

Attack types	Accuracy (%)	Precision (%)	Recall (%)	F1- score (%)
Sensor data flood	97.39	97.38	97.39	97.33
Network data flood	99.98	99.98	99.98	99.98

D. SDN Controller

The SDN controller and switches in our environment are products of EstiNet Technologies Inc. The version of the SDN controller is 1.5.4.5. The used model of SDN switches is RT166P. First, we set up rules of bandwidth control and blacklist by SDN controller. Second, we analyze the bandwidth by Wireshark that is a network tool to trace network traffic. The bandwidth is the average of the ratio of the amount of information by the total time. Third, we analyze the bandwidth in the normal situation and the DDoS attack environment. Next, we utilize port mirroring such that the switch sends a copy of all packets received on one port to a mirroring port, where the packets can be stored and be used for data analysis. Finally, we experiment with the bandwidth in the mirror port, which collects all packets from the eight smart poles for different quantity of malicious devices that launched DDoS attacks on the Internet. If there is no DDoS attack, the bandwidth measured from the mirror port is about 10 Mbps. If we launched a UDP flood from a single device, we could receive about 80~100 Mbps data flow, while the SYN flood and ICMP flood can generate little data flow. Here, we use Dell PowerEdge R630 as our cloud server, which equipped with Intel Ethernet I350 Quad Port 1Gb Server Adapter. Based on the maximum bandwidth of the connected port on our hardware equipment, we set the bandwidth control rule of the SDN switch to 800 Mbps to prevent devices falling into overload status when hackers launch a large-scale DDoS attack. There are two modes for bandwidth control that are reservation and slicing. Reservation means that if the capacity is left, other users can use the remaining capacity. Slicing means that if there is residual capacity, other users cannot use the residual capacity that is kept by its original owner. Consequently, we take advantage of slicing technology to guarantee there is no problem for our smart poles to transmit sensor data.

V. Conclusion

In this paper, we propose a multi-layer DDoS detection system based on machine learning to prevent DDoS attacks in IoT gateway. We extract features of four types of DDoS attacks, including sensor data flood, ICMP flood, SYN flood, and UDP flood and make these features to be numerical. We launch DDoS attacks from eight smart poles as a real IoT scenario and show that our multi-layer DDoS detection system can distinguish normal packets and DDoS attack packets from IoT devices accurately. Our proposed system can detect DDoS attacks with high accuracy. The F1-score is over 97%. When abnormal packets are detected, the IP addresses and MAC addresses of the malicious devices are sent to the SDN controller. The SDN controller adds these IP addresses, and MAC addresses into blacklists and sets the rules for SDN switches from the blacklists. The devices in the blacklists are blocked immediately in the SDN switches. In this way, our proposed multi-layer DDoS detection system not only detects the DDoS attacks but also blocks the malicious devices. Since the features are marked artificially, and the trained model is not portable to a new field. This solution can be improved with other unsupervised learning methods in future work.

References

[1] J. Mirkovic and P. Reiher, "A Taxonomy of DDoS Attack and DDoS Defense Mechanisms," *ACM SIGCOMM Computer Communications Review*, vol. 34, no. 2, pp. 39-53, 2004.

[2] Z. He, T. Zhang, and R.B. Lee, "Machine Learning Based DDoS Attack Detection from Source Side in Cloud," *Proceedings of the IEEE 4th International Conference on Cyber Security and Cloud Computing* (CSCloud), New York, NY, USA, July 2017.

[3] S.S. Mohammed, R. Hussain, B. Bimaganbetov, and J. Lee, "A New Machine Learning-based Collaborative DDoS Mitigation Mechanism in Software-Defined Network," *Proceedings of the 14th International Conference on Wireless and Mobile Computing, Networking and Communications* (WiMob), Limassol, Cyprus, Oct. 2018.

[4] Strategy Analytics. (2020) Global Connected and IoT Device Forecast Update. [Online]. Available: https://www.strategyanalytics.com/access-services/devices/connected-home/consumer-electronics/reports/report-detail/global-connected-and-iot-device-forecast-update

[5] E. Aras, G.S. Ramachandran, P. Lawrence, and D. Hughes, "Exploring the Security Vulnerabilities of LoRa," *Proceedings of the 3rd IEEE International Conference on Cybernetics* (CYBCONF), Exeter, UK, June 2017.

[6] S. Dowling, M. Schukat, and H. Melvin, "A ZigBee Honeypot to assess IoT Cyberattack Behaviour," *Proceedings of the 28th Irish Signals and Systems Conference* (ISSC), Killarney, Ireland, June 2017.

[7] K. Chen, A. R. Junuthula, I. K. Siddhrau, Y. Xu, and H. J. Chao, " SDNShield: Towards More Comprehensive Defense against DDoS Attacks on SDN Control Plane," *Proceedings of* the *IEEE Conference on Communications and Network Security* (CNS), Philadelphia, PA, USA, Feb. 2017.

[8] K. Kalkan, L. Altay, G. Gür, and F. Alagöz, "JESS: Joint Entropy-Based DDoS Defense Scheme in SDN," *IEEE Journal on Selected Areas in Communications*, Sep. 2018.

[9] D. Yin, L. Zhang, and K. Yang, "A DDoS Attack Detection and Mitigation With Software-Defined Internet of Things Framework," *IEEE Access*, Apr. 2018.

[10] Y. Liu, M. Dong, K. Ota, J. Li, and J. Wu, "Deep Reinforcement Learning based Smart Mitigation of DDoS Flooding in Software-Defined Networks," *Proceedings of the IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks* (CAMAD), Barcelona, Spain, Sept. 2018.

[11] M. Zekri, S.E. Kafhali, N. Aboutabit, and Y. Saadi, "DDoS Attack Detection Using Machine Learning Techniques in Cloud Computing Environments," *Proceedings of the 3rd International Conference of Cloud Computing Technologies and Applications* (CloudTech), Rabat, Morocco, Oct. 2017.

[12] Y. Feng, H. Akiyama, L. Lu, and K. Sakurai, "Feature Selection for Machine Learning-Based Early Detection of Distributed Cyber Attacks," *Proceedings of the IEEE 16th Intl Conf on DASC/PiCom/DataCom/CyberSciTech*(DASC/PiCom/DataCom/CyberSciTech), Athens, Greece, Oct. 2018.

[13] V. Selis and A. Marshall, "A Classification-Based Algorithm to Detect Forged Embedded Machines in IoT Environments," *IEEE Systems Journal*, May 2018.

[14] M. Roopak, G. Y. Tian, and J. Chambers, "Deep Learning Models for Cyber Security in IoT Networks," *Proceedings of the IEEE 9th Annual Computing and Communication Workshop and Conference* (CCWC), Las Vegas, NV, USA, Mar. 2019.

[15] K. Gurulakshmi and A. Nesarani, "Analysis of IoT Bots Against DDOS Attack Using Machine Learning Algorithm," *Proceedings of the 2nd International Conference on Trends in Electronics and Informatics* (ICOEI), Tirunelveli, India, Dec. 2018.

[16] R. Doshi, N. Apthorpe, and N. Feamster, "Machine Learning DDoS Detection for Consumer Internet of Things Devices," *Proceedings of the IEEE Security and Privacy Workshops* (SPW), San Francisco, CA, USA, May 2018.

[17] Q. Yan, W. Huang, X. Luo, Q. Gong, and F. Richard Yu, "A Multi-Level DDoS Mitigation Framework for the Industrial Internet of Things," *IEEE Communications Magazine*, Feb. 2018.

[18] "Dataset," *http://smartcity.cs.nthu.edu.tw/dataset/download.php*, 2019.