A DDoS attack detection based on deep learning in software-defined Internet of things

Jiushuang Wang, Ying Liu, Wei Su, Huifen Feng
National Engineering Laboratory for Next Generation Internet Technologies
Beijing Jiaotong University
Beijing, China
jswang0630@163.com, {yliu, wsu, 19111012}@bjtu.edu.cn

2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall) | 978-1-7281-9484-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/VTC2020-Fall49728.2020.9348652

Abstract—**With the popularity of Internet of Things (IoT) applications, security has become extremely important. A recent distributed denial-of-service (DDoS) attack revealed vulnerabilities that are prevalent in IoT, and many IoT devices accidentally contributed to the DDoS attack. software-defined network provides a way to securely manage IoT devices. In this paper, we first present a general framework for software-defined Internet of Things (SD-IoT). The proposed framework consists of a SD-IoT controller, SD-IoT switches integrated with an IoT gateway, and IoT devices. We then propose a deep learning detection algorithm based on time series using the proposed SD-IoT framework. Finally, experimental results show that the proposed algorithm has good performance.**

Keywords—Software-defined Internet of Things, distributed denial of service, deep learning, time series, attack detection

I. Introduction

The Internet of Things is regarded as the third wave of the development of the world's information industry after computers and the Internet. The IoT is the technology that aims to connect anything, at any place, at anytime[1]. In recent years, with the development of microelectronics technology, embedded technology and wireless network technology, the IoT technology has made great progress and is widely used in intelligent transportation, environmental monitoring, industrial control, e-commerce, defense and military fields [2][3]. The security issues of the IoT have become increasingly prominent and become a major bottleneck restricting the development of the IoT. At the same time, due to the large number of IoT devices, a large number of IoT devices with security vulnerabilities have been captured by hackers, and the phenomenon of forming botnets and launching cyber attacks has become more frequent [4]. At present, IoT devices have become a major source of growth in distributed denial-of-service attacks, causing great disruption to normal data transmission at the IoT network layer. Therefore, how to effectively detect and prevent network attacks is an urgent problem that needs to be solved in the process of IoT security management.

The emergence of Software-Defined Networking (SDN) provides a new opportunity for solving the above problems. The key feature of SD-IoT is that it decouples network control and forwarding functions[5]. The control layer's SD-IoT controller usually runs on a powerful server platform with centralized management and real-time monitoring capabilities. It can easily implement security policies and detection mechanisms that are difficult to implement in traditional IoT and Wireless Sensor Network in the controller and utilize the global topology view [6][7]. As the data layer of the IoT, SD-IoT switches such as switches, base stations, routers and wireless access points only carry out flow forwarding. Utilizing a unified southbound interface, SD-IoT can effectively respond to multiple types of IoT network protocols, simplify the configuration and management of IoT devices, and reduce the cost of business implementation and operation and maintenance. This decoupling avoids potential operational failures and service interruptions, ensures continuous availability of IoT devices, prevents unauthorized access to peripheral devices, monitors and controls devices that change the Internet, detects legitimate and malicious traffic patterns on IoT devices, and ultimately reduce the risk of IoT security [8].

II. Related Work

Currently, DDoS attack detection methods in SDN are roughly divided into three types: statistical-based approaches, policy-based approaches and machine-learning approaches. Statistical-based methods use statistical methods to analyze SDN network traffic to distinguish between normal data traffic and DDoS attack traffic. Wang et al. [9] proposed an entropy-based DDoS attack detection system in SDN. By processing the statistics of the flow table in the switch, the detection system can prevent DDoS attacks. The detection scheme based on information entropy is simple and occupies less resources, but the detection information is single. Policy-based detection schemes are based on enforcement of certain policies on traffic flows. If the flows are in accordance with these policies or rules, they are considered legitimate flows otherwise reported as attack data[10]. Shin et al. [11] proposed a policy-based solution Avant-Guard, which is a DDoS attack detection and mitigation scheme. This scheme uses TCP connection state information to classify legitimate and malicious traffic. The advantages of policy-based detection of attack methods are higher accuracy, but the disadvantage is that features must be extracted again when a new attack mode appears. The method based on machine learning is to use machine learning algorithms to learn, and then obtain the best learning model through training according to the characteristics of the traffic samples in SDN. Ye et al. [12] extended the six-tuples in the switch flow table to feature vectors. These features are then combined with SVM to detect DDoS attacks. Machine learning methods are very accurate. but their detection is complicated.

How to use SDN technology to strengthen and improve the security of the IoT is a preliminary research topic. Nobakht et al. [13] proposed a host-based intrusion detection and mitigation framework for IoT based on SDN. Their mechanism, called IoT-IDM, identifies and addresses potential attacks against a target host. Salman et al. [14] proposed an identity-based authentication scheme for IoT based on SDN. It can be an effective solution for IoT security. Gonzalez et al. [15] proposed an SDN-based security solution for the IoT, a routing protocol suitable for distributed clusters, and built a test platform to evaluate the protocol. Nguyen et al.

978-1-7281-9484-4/20/$31.00 ©2020 IEEE

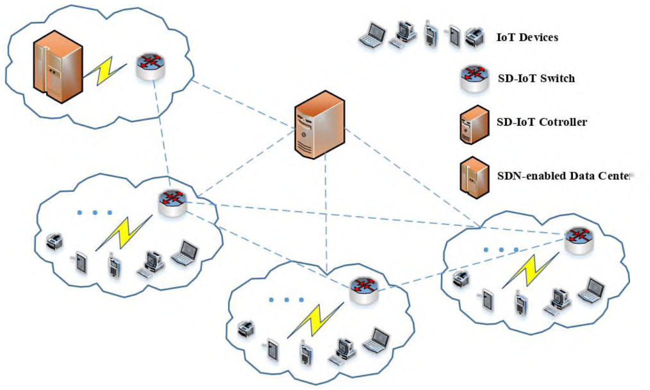

Fig. 1. SD-IoT framework

[16] proposed a hybrid strategy to prevent link spoofing attacks in SD-IoT controllers. Bull et al. [17] proposed a flow-based security approach for IoT devices using an SDN gateway. It aims to mitigate DDoS attacks that violate the system's availability. Ahmed et al. [8] discussed the mitigation of DDoS attacks in the IoT. They tried to use the SDN to alleviate the constraints of the sample-based anomaly detection system to achieve detection accuracy.

Research on the integration of SDN and IoT networks is still in its infancy. Many issues remain, such as network frameworks and security. In this paper, we research the DDoS attack detection method in the proposed SD-IoT framework. This article contributes as follows:

- A security framework combining SDN and IoT is proposed. The proposed framework includes a single controller, an IoT switch integrated with an IoT gateway, and an IoT device.
- The time series of data flow is used as the feature vector to detect DDoS attacks.
- Using deep learning algorithm to detect DDoS attacks improves the accuracy of detection.
- Simulation results show that the proposed deep learning algorithm is better than other traditional machine learning algorithms.

This paper is organized as follows. We summarize the related work and main contributions of this paper in section II. In section III, we introduce the proposed SD-IoT framework and describe the problem of DDoS attacks in IoT. We propose an algorithm for detecting DDoS attacks based on the proposed SD-IoT framework in Section IV. Section V presents the experimental settings and evaluates the performance. Finally, the conclusion is given in Section VI.

III. DDoS Attacks in the sd-IoT Environment

A. SD-IOT Framework

In this study, a network model created by applying the IoT infrastructure to a basic SDN environment is presented. As illustrated in Fig.1, the proposed SD-IoT framework can be seen as an extended version of the SDN architecture applied to IoT. The framework can be divided into three layers: the application layer, control layer and infrastructure layer.

Application layer: The application layer consists of an IoT server in the data center and is connected to the SD-IoT controller. Through the service interface provided by the controller, developers can easily implement intelligent IoT applications at the application layer [18].

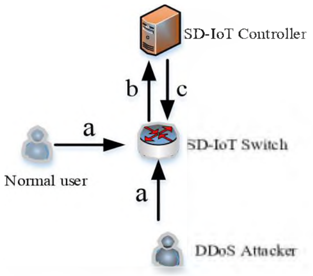

Fig. 2. DDoS attacks in SD-IoT

Control layer: The control layer consists of the SD-IoT controller. This controller provides logical centralized control and topology view for IoT data forwarding.

Infrastructure layer: The infrastructure layer is where the network devices are located and consists of a mass of SD-IoT switches. In the scheme proposed in this paper, every SD-IoT switch integrates the function of an IoT gateway and an SDN switch.

B. DDos Attacks in SD-IoT

A DDoS attack is a type of attack in which devices are attacked from multiple sources in a distributed manner, resulting in denial of service to users. SDN provides detection solutions for DDoS attacks through its programmability feature [19]. In the proposed framework, the SD-IoT controller is responsible for the logic centralized control of the entire IoT environment. The logic centralized control is easy to manage and configure but also causes security issues. Similar to SDN, SD-IoT is capable of offering constructive schemes for detecting DDoS attacks through the programmability of the proposed SD-IoT framework. In the following, we will analyze the process of DDoS attacks in SD-IoT, as illustrated in Fig. 2.

a) The DDoS attacker sends a new packet to a certain SD-IoT switch, where the attack packet is generated by the attack script.

b) The SD-IoT controller periodically collects statistical information on the SD-IoT switch of the current flow entries based on the openflow protocol.

c) According to the result given by the detection algorithm in the controller, the SD-IoT switch performs the next action.

IV. DDoS Attack Detection Model

This model enables detection of DDoS attacks in SD-IoT. This solution consists of four modules, and each module will be described in detail below.

A. Flow Entry Collection Module

The main function of the flow entry collection module is to collect statistics information about current flow entries items of the switch periodically. This module mainly collects the flow entry of the switch through the OpenFlow protocol. The flow entry is the basis of flow forwarding. The controller sends an ofp-flow-stats-request message to the switch to collect the flow table by setting it. The definition of the time interval to collect flow entries is of great importance. If collection is made at infrequent time intervals, then there will

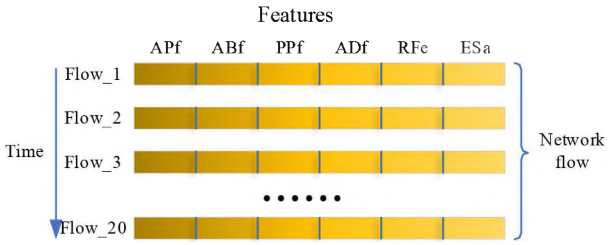

Fig. 3. The input of CNN model

be a delay to detect an attack and could be fatal to the system. On the other hand, if the time interval for collection is too short, there will be an increase of packets requesting flows which will lead to an increase in the overhead of our detection mechanism. Here, the flow collection module sends requests to switches every 6 seconds to obtain the flow entry information and submits the information to the feature extraction module.

B. Feature Extraction Module

The main function of the feature extraction module is to compute the characteristics of the data flow through the flow entry collection module. In order to implement the deep learning algorithm in this paper, we need to determine the features. DDoS attackers can use multiple attack means and methods, but most of the attack traffic comply with some rules. Therefore, the flow features can be used for detection. For example, when a DDoS attack occurs, the main attack mode is source IP address spoofing, which makes source IP more dispersed. Through the above analysis, we used 6-element features related to DDoS attacks.

1) Average of Packets per flow （APf）: Through research, we know that the traffic in the attack state differs from the normal traffic in the number of packets. Because DDOS attacks can use source IP spoofing to randomly generate fake IP addresses, this feature quickly generates a large number of flows, but the number of packets in each flows will be reduced. Each flow in normal traffic contains a large number of packets. Therefore, APf can be used to represent the flow characteristics. The formula is as follows：

$$APf = \frac{\sum_{i=1}^{flow\_count_k} packet\_count_i}{flow\_count_k} \tag{1}$$

Where $flow\_count_k$ is the number of flow entries whose destination address is IP_k, and $packet\_count_i$ is the number of packets of i_{th} flow entry.

2) Average of Bytes per flow （ABf）: In order to improve the efficiency of DDoS attacks, the payload of the attack state flow is usually very small. Therefore, ABf is an important feature for detecting DDOS attacks. The formula is as follows：

$$ABf = \frac{\sum_{i=1}^{flow\_count_k} flow\_bytes_i}{flow\_count_k} \tag{2}$$

Where $flow\_bytes_i$ is the byte size of i_{th} flow entry.

3) Percentage of Pair-flows （PPf）: Normal traffic in the network is interactive. Because the attack traffic forges IP addresses, the number of single flow is increased. The formula is as follows：

$$PPf = \frac{2 \times pair\_flows\_count_k}{flow\_count_k} \tag{3}$$

Where $pair\_flows\_count_k$ is the number of pair flow of the flow whose destination address is IP_k.

4) Average of Duration per flow （ADf）: The flow entry rule of a normal flow lasts for a long time. Anomalous flow is where the attacker randomly sends a lot of useless packets, so most flow rules will soon be idle. The formula is as follows：

$$ADf = \frac{\sum_{i=1}^{flow\_count_k} durations_i}{flow\_count_k} \tag{4}$$

Where $durations_i$ is the duration of ith flow entry whose destination address is IP_k.

5) Rate of Flow entries (RFe): When a DDoS attack occurs, a large amount of useless traffic is sent to the victim host, and the request of the victim host in the network increases. so the number of flow entries related to the host increases for a certain period of time. Therefore, the rate of flow entries is also an important feature of DDoS attack. The formula is as follows：

$$RFe = \frac{flow\_count_{kT} - flow\_count_{kt}}{interval} \tag{5}$$

Where $flow\_count_{kt}$ is the number of flow entries with the dstIP_k at time t, and $interval$ is the time interval from T to t.

6) Entropy of Source IP addresses(ESa): DDoS attacks generate a large number of forged source IP addresses(srcIP). The source IP addresses are relatively scattered and random, so the entropy value of the srcIP of the attack traffic is greater than the normal srcIP entropy value. The formula is as follows：

$$ESa = -\sum_{i=1}^{m} \frac{srcIP_{ik}}{flow\_count_k} \log \frac{srcIP_{ik}}{flow\_count_k} \tag{6}$$

Where $srcIP = \{srcIP_{1k}, srcIP_{2k}, \dots, srcIP_{ik}, \dots, srcIP_{mk}\}$, $srcIP_{mk}$ is the number of flow entries from source address IP_i to destination IP_k.

C. DDoS Attack Detection

The features of normal traffic and attack traffic are different. Therefore, attack detection can be regarded as a classification problem to detect whether the current network is normal. The flow entry collection module collects flow table information. The feature extraction module extracts 6 features. Finally, samples are trained through the attack detection module. After training, the attack detection module can detect whether a DDoS attack has occurred.

a) Data processing: The controller periodically obtains packets from the switch every 6 seconds, and the number of packets obtained each time is n. In this n packets, 50 packets as a window. There are m flows in a window. If $m < 20$, 6 features of each flow are extracted, and the remaining $20 - m$ flow features are filled with zeros as the input of CNN. If $m > 20$, then every 20 flow features constitute CNN input.

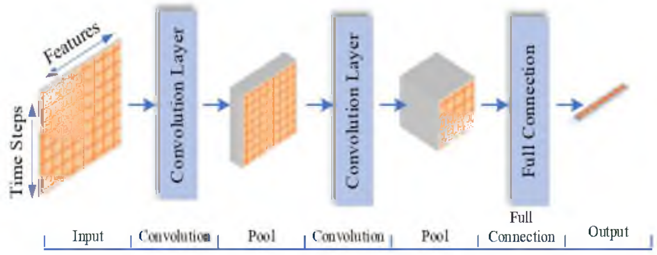

Fig. 4. Structure of CNN

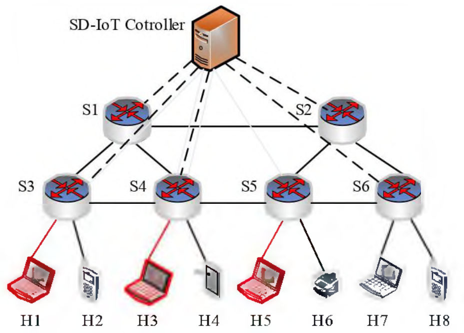

Fig. 5. Experimental topology

As shown in Fig. 3. Finally, this time series is used as the input of the deep learning model in this paper for DDoS attack detection.

b) Deep learning model: The basic structure of convolutional neural network (CNN) includes input layers, convolutional layers, pooling layers, fully connected layers, and output layers. The input layer is used to receive multi-dimensional data, and the convolution layer extracts features from the data input. The pooling layer further simplifies functionality by reducing the amount of data by changing the pooling size, steps, and padding parameters. The fully connected layer performs nonlinear combination and classification of features. The final output layer outputs the classification results. Fig. 4 is a model structure of a CNN.

V. Performance Evaluation

A. Simulation Settings

The essential tools of the experimental platform include software-defined controllers, software-defined switches and DDoS attack software. All software is deployed on the VirtualBox platform of Windows 10 system, and the operating system is Ubuntu 16.0.4 LTS. In particular, the open source controller Floodlight is used as the SD-IoT controller; Open vSwitch is used as the SD-IoT switch; deep learning module is developed based on Tensorflow framework; Mininet is deployed on Ubuntu 16.0.4 LTS to simulate the network topology of SD-IoT. Its API interface is used for development and testing. In addition, the DDoS attack flow and normal flow are both generated through the Scapy library of the python script. The Scapy library is a powerful interactive program that developers can use to forge or parse the contents of packets and current network protocols. The Scapy library supports real-time capture and generation of data packets. The python script can simulate complex network operations through the Scapy library and also control the number and rate of sending packets.

An instance of the SD-IoT topology generated based on Mininet, is used to validate the proposed algorithm, as illustrated in Fig.5. we choose one Floodlight controller as the current SD-IoT controller and use six Open vSwitches as SD-IoT switches, which are referred to as S0, S1, . . ., and S6. These are 8 terminal devices of the IoT, and they are referred to as H0, H1, . . ., and H8. Select three of the hosts H1, H3, and H5 as the DDoS attack source to launch an attack on the SDN network.

TABLE I. Number of Samples

Dataset	Normal samples	DDoS samples
Training Set	12000	9000
Test Set	8000	6000

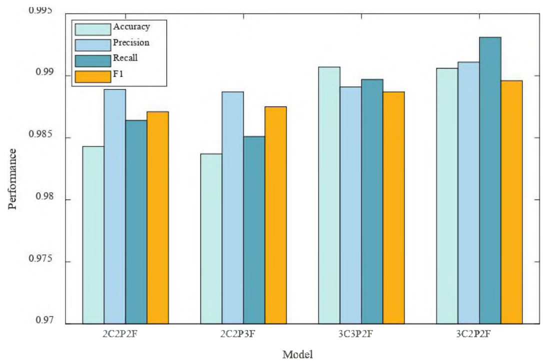

Fig. 6. Performance comparison of different layers

B. Analysis of CNN Experimental Results

The experimental data set is the real traffic in the SDN network we collected. We collected a total of 35000 pieces of traffic data, including 20000 pieces of normal traffic and 15000 pieces of DDoS attack traffic. We use 60% of the total data traffic as the training set and 40% of the data as the test set. As shown in TABLE I：

When building a CNN model, convolution layers of different depths will have a greater impact on the detection accuracy of the model. CNN models with 4 different depths were established during the experiment. Among them, C2P2F3 indicates that the model contains 2 convolutional layers, 2 maximum pooling layers, and 3 fully connected layers.

This paper conducts experiments on the constructed CNN model, and evaluates the performance of the model through the defined accuracy, precision, recall, and F1 evaluation indicators. CNN model evaluation indicators at 4 different depths are shown in Fig. 6.

The neural network model (3C2P2F, 3C3P2F) using 3 convolution layers is significantly better than the neural network model (2C2P2F, 2C2P3F) using 2 convolution layers. The 3C2P2F model has higher accuracy, recall and F1 scores than the 3C3P2F model, but its accuracy is slightly lower than the 3C3P2F model.

In order to compare the detection results of 3C2P2F model and 3C3P2F model, four models were analyzed by the confusion matrix. The confusion matrix of four CNN models is shown in Fig. 7. The top two confusion matrices are 2C2P2F and 2C2P3F, and the bottom two confusion matrices are 3C3P2F and 3C2P2F. A neural network model using a two-layer convolutional layer is weaker than a neural network

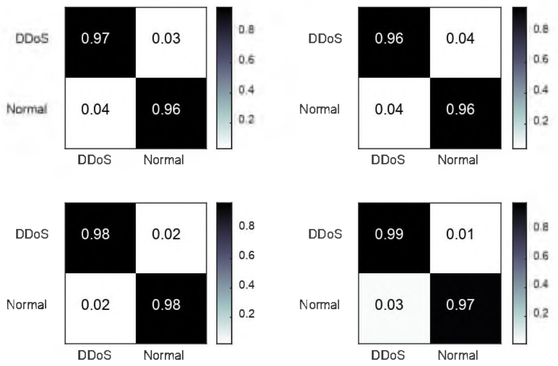

Fig. 7. Confusion matrix of 4 CNN models

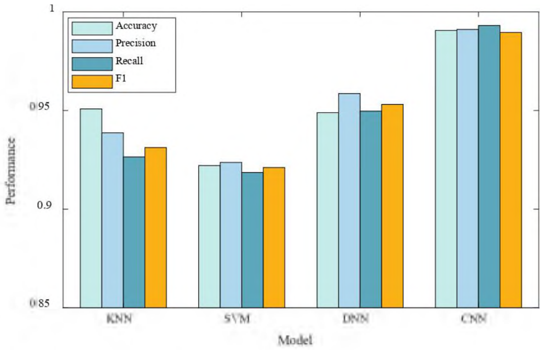

Fig. 8. Performance comparison of different models

model using a three-layer convolutional layer in its ability to judge normal and attack packets. At the same time, the ability of the 3C2P2F model to judge attack packets (0.99) is better than the 3C3P2F model (0.98). Considering that DDoS attacks may jeopardize the security of the entire system, the detection model adopted must be highly sensitive to the attack. Taking into account the CNN model, the 3C2P2F model is the best.

According to the above experimental results, the 3C2P2F model is selected for performance comparison with three typical machine learning methods. The evaluation indicators remain accuracy, precision, recall and F1. It can be seen from Fig. 8 that the CNN algorithm can obtain higher accuracy, precision, recall, and F1. It illustrates the superiority of convolutional neural networks in traffic detection and has the potential to effectively distinguish DDoS traffic from normal traffic.

VI. Conclusion

In this paper, we describe a general framework for SD-IoT composed of an SD-IoT controller, SD-IoT switches integrated with the IoT gateway, and terminal IoT devices. Then, we propose an algorithm for detecting DDoS attacks with the proposed SD-IoT framework. In this algorithm, we use the extracted features as the input of the deep learning algorithm in a time series manner, and finally detect whether it is a DDoS attack. By comparing different combinations of convolutional layers, maximum pooling layers, and fully connected layers, the model of the best convolutional neural network 3C2P2F in the proposed framework is found. Evaluations were performed by comparing the method with the DNN, SVM, and KNN. The proposed method has some outstanding properties in terms of accuracy, precision, recall, and F1.

Acknowledgment

This was supported by the National Key Research and Development Program of China under Grant 2018YFB1800305. (Corresponding author: Ying Liu)

References

[1] J. Zheng, D. Simplot-Ryl, C. Bisdikian, and H. Mouftah, “The Internet of Things,” IEEE Commun. Mag., vol. 49, no. 11,2011, pp. 30–31.

[2] X. Liu, S. Zhao, A. Liu, N. Xiong, and A. V. Vasilakos, ‘‘Knowledgeaware proactive nodes selection approach for energy management in Internet of Things,’’ Future Generat. Comput. Syst., Aug. 2017.

[3] Y. Liu, A. Liu, S. Guo, Z. Li, Y.-J. Choi, and H. Sekiya, ‘‘Context-aware collect data with energy efficient in cyber–physical cloud systems,’’ Future Generat. Comput. Syst., Jun. 2017.

[4] C. Kolias, G. Kambourakis, A. Stavrou and J. Voas, "DDoS in the IoT: Mirai and Other Botnets," in Computer, vol. 50, no. 7, pp. 80-84, 2017.

[5] D. Yin, L. Zhang and K. Yang, "A DDoS Attack Detection and Mitigation With Software-Defined Internet of Things Framework," in IEEE Access, vol. 6, pp. 24694-24705, 2018.

[6] K. Kalkan and S. Zeadally, "Securing Internet of Things with Software Defined Networking," in IEEE Communications Magazine, vol. 56, no. 9, pp. 186-192, Sept. 2018.

[7] Habib Mostafaei and Michael Menth. Software-Defined Wireless Sensor Networks: A Survey[J]. Journal of Network and Computer Applications(JNCA), 2018, 119:42-56.

[8] M. E. Ahmed and H. Kim, ‘‘DDoS attack mitigation in Internet of Things using software defined networking,’’ in Proc. IEEE 3rd Int. Conf. Big Data Comput. Service Appl., Apr. 2017, pp. 271–276.

[9] R. Wang, Z. Jia and L. Ju, "An Entropy-Based Distributed DDoS Detection Mechanism in Software-Defined Networking," 2015 IEEE Trustcom/BigDataSE/ISPA, Helsinki, 2015, pp. 310-317.

[10] Dayal N , Maity P , Srivastava S , et al. Research Trends in Security and DDoS in SDN[J]. Security and Communication Networks, 2016, 9(18):6386-6411.

[11] Shin S, Yegneswaran V, Porras P, Gu G. AVANTGUARD: scalable and vigilant switch flow management in software-defined networks. Proceedings of Conference on Computer and Communication Security(CCS), ACM, 2013; 413–424.

[12] J. Ye, X. Cheng, J. Zhu, L. Feng, L. Song, A ddos attack detection method based on SVM in software defined network, Secur. Commun. Netw. 2018(4) (2018) 1–8.

[13] M. Nobakht, V. Sivaraman, and R. Boreli, “A Host-Based Intrusion Detection and Mitigation Framework for Smart Home IoT using OpenFlow,” Proc. IEEE 11th Int’l. Conf. Availability, Reliability and Security (ARES), 2016, pp. 147–56.

[14] O. Salman et al., “Identity-Based Authentication Scheme for the Internet of Things,” Proc. IEEE 21st Symposium on Computers and Communication (ISCC), Italy, 2016, pp.1109–11.

[15] C. Gonzalez, O. Flauzac, F. Nolot and A. Jara, "A Novel Distributed SDN-Secured Architecture for the IoT," 2016 International Conference on Distributed Computing in Sensor Systems (DCOSS), Washington, DC, 2016, pp. 244-249.

[16] T. H. Nguyen and M. Yoo, “A hybrid prevention method for eavesdropping attack by link spoofing in software-defined Internet of Things controllers,” Int. J. Distrib. Sensor Netw., vol. 13, no. 11, pp. 1-9, Nov. 2017.

[17] P. Bull et al., “Flow Based Security for IoT Devices using an SDN Gateway,” Proc. IEEE 4th Int’l. Conf. Future Internet of Things and Cloud (FiCloud), Austria, 2016, pp.157–63.

[18] Liu, Jiaqiang , et al. "Software-defined internet of things for smart urban sensing." IEEE Communications Magazine 53.9(2015):55-63.

[19] D. B. Rawat and S. R. Reddy, ‘‘Software defined networking architecture, security and energy efficiency: A survey,’’ IEEE Commun. Surveys Tuts., vol. 19, no. 1, pp. 325–346, 1st Quart., 2017.