An SDN System for Intelligent Botnet Behavior Suppression

Shih-Chen Wang[1], Yi-Chen Lee[1], Wei-Che Chien[1], Guanling Lee[1], Sheng-Lung Peng[2]*

[1] Department of Information Engineering, National Dong Hwa University, Taiwan
[2] Department of Creative Technologies and Product Design, National Taipei University of Business, Taiwan
{810621002, 611121201, wcc, guanling}@gms.ndhu.edu.tw, slpeng@ntub.edu.tw

Abstract

In recent years, with the widespread application of the Internet of Things (IoT), its coverage has expanded to encompass numerous devices and communication entities. However, with the proliferation of these services, hackers resort to various means to infiltrate computer systems for the purpose of stealing valuable information or extortion, turning them into Botnet. Furthermore, hackers can utilize these botnets to launch Distributed Denial of Service (DDoS) attacks against target devices, depleting the resources of the target system, thereby rendering services unusable. Therefore, there is an urgent need to develop an effective mechanism to prevent such attacks. In our research, we introduce a DDoS detection model that is based on the ELK (Elasticsearch, Logstash, Kibana) system and incorporates the Federated Learning (FL) framework. The model is designed for internal traffic inspection, and ensuring data privacy through the utilization of FL, thereby eliminating the need for data sharing among different IoT devices during the training process. Following the identification and prevention of Botnet hosts and DDoS attacks, automated responses are executed via Software-Defined Networking (SDN) systems. This approach not only proposes a more efficient intelligent system but also alleviates the workload on network management personnel.

Keywords: Distributed Denial of Service (DDoS), Federated Learning (FL), Software-Defined Networking (SDN), Internet of Things (IoT)

1 Introduction

In recent years, under the influence of the rapid development of IoT (Internet of Things) [1], all kinds of electronic devices have been continuously added to network applications, coupled with the emergence of information security attack methods in recent years, virtually increasing the pressure on network management personnel, but the corresponding measures often require a lot of manpower to set up and manage.

The research is established by discussing the rapid development of IoT (Internet of Things) and the significant increase in connected devices, leading to increased pressure on network management personnel due to emerging information security attack methods.

Since 2003, the number of objects on the Internet of Things has grown from less than one million objects to 50 billion objects connected to the Internet of Things. It can be seen that the objects on the Internet of Things are growing rapidly. According to the network behavior of each device (including people and equipment), combined with the analysis of big data, and using the model of artificial intelligence, this research will determine the role of the device and its required reasonable network configuration and permissions. Through SDN (Software Defined Networking) [2] technology, appropriate permissions are distributed, so that all devices (including people and equipment) connected to the network can operate in a highly automated manner, and at the same time optimize the use of information security resources.

In the past, it was not easy for hackers to launch a wave of DDoS attacks [3]. They had to raise troops for thousands of days and then use them for a while. Usually, they had to invade and cultivate a sufficient number of zombie PCs [4] and servers to launch them. IoT devices have grown significantly. These devices are usually in a state of continuous power supply. Even if they are idle, they are connected to the Internet 24 hours a day without interruption. These devices are also embedded with small operating systems that are responsible for the operation of the devices. Therefore, these devices have also become the best criminal hotbeds for DDoS.

The packet is the smallest unit of information transmitted by the network [5]. Various messages and services are transmitted to the destination through packets, and then reassembled into meaningful information and provided services. DDoS attacks use this mechanism to penetrate excessive packet transmission making normal network devices unable to bear its huge amount of data for a while, and then paralyzing network functions.

Fortunately, some information security equipment manufacturers and software companies are able to detect whether the packet is a DDoS attack through packet detection, and further initiate data shunting to direct the attack packet to another server for processing, thereby resolving the DDoS attack.

From another perspective, DDoS attacks rely on infiltrating a large number of devices to serve as their botnets [6] or attack proxies. Thus, mitigating DDoS attacks can also involve effectively managing these devices to reduce their susceptibility to infiltration. Especially

*Corresponding Author: Sheng-Lung Peng; Email: slpeng@ntub.edu.tw

DOI: https://doi.org/10.70003/160792642025112606007

in the era of IoT, where a vast number of devices are interconnected, each with varying functionalities, managing them poses significant challenges and increases the likelihood of cybersecurity issues. Beyond security concerns, efficiency also becomes a paramount issue. While most devices boast plug-and-play capabilities, without robust device interface management mechanisms, efficiency and security issues, as mentioned earlier, can easily arise.

We now discuss the vulnerability of IoT devices to DDoS attacks due to their continuous connectivity and highlights the need for better network configuration and permissions to manage these devices efficiently. It also addresses the difficulty of manually monitoring network packets in a complex network environment.

This research aims to utilize artificial intelligence and deep learning theories, combined with federated learning, to analyze device behavior patterns and DDoS detection systems and determine appropriate security permissions for newly added devices. This approach enables precise device management and optimizes resource utilization. However, to ensure data privacy [7] and security, we adopt deep learning combined with federated learning training models, thereby avoiding the need for data sharing between different devices during the training process.

The main contribution of this paper is the proposal of a method using artificial intelligence, deep learning, and federated learning to analyze device behavior and detect DDoS attacks, thereby assigning appropriate security permissions and optimizing resource utilization.

Through commonly used packet inspection tools, information personnel can see the source and purpose of the packet, which protocols are used, and how many bytes are transmitted. However, in a network environment, it is impossible for information personnel to manually monitor the packets of each device and find out key information. Therefore, based on the above-mentioned concept of packet detection technology, Figure 1 is one of the scenarios considered in this study. The users in it refer generally to people and equipment.

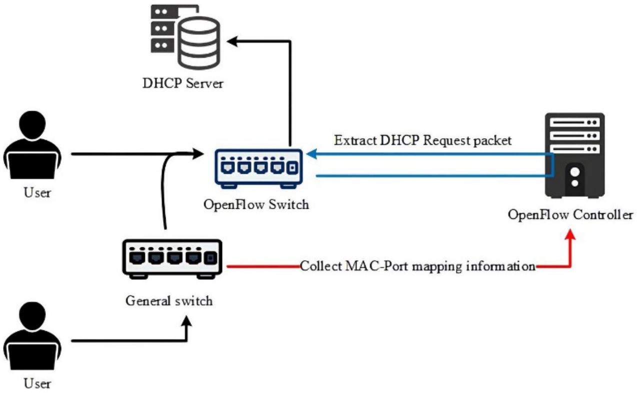

Figure 1. Concept of AI + SDN controlling user access to the network

The structure of this paper is as follows. Section 2 covers background and related works on botnets, ELK data storage, smart network packet detection, SDN, and federated learning. Section 3 presents the proposed approach, detailing the architecture, packet behavior detection, and the overall mechanism. Section 4 discusses the experiments, including test architecture, behavioral statistics, and detection results. Finally, Section 5 concludes the paper and suggests future work.

2 Background and Related Works

2.1 Botnet

A botnet is a group of computers that have been infected with malware and come under the control of malicious actors. The term botnet [8] is a combination of the words robot and network, and each infected device is called a bot. Botnets can be designed to accomplish illegal or malicious tasks, including sending spam [9], stealing data, ransomware, phishing, or distributed denial-of-service (DDoS) attacks.

While some malware, such as ransomware [10], will directly affect the owner of the device, DDoS botnet malware may have a different level of visibility. Some malware is designed to take full control of the device, while others run silently as background processes, silently waiting for instructions from the attacker.

Botnets capable of autonomous propagation infect other botnets through a variety of different channels. Infection methods include exploiting website vulnerabilities, Trojan horses [11], or system vulnerabilities to gain remote access. Once access is gained, all of these infection methods install malware on the target device, allowing remote control by botnet operators. Once a device is infected, it may attempt to self-propagate botnet malware by recruiting other hardware devices in the surrounding network.

2.2 ELK (Elasticsearch, Logstash, Kibana) Data Storage Technology

When it comes to big data, it is necessary to explain the relevant and important software technology “Hadoop” [12]. Hadoop is an open-source software framework developed by the Apache Software Foundation, written and developed by the Java language. It is free, highly scalable, fast to deploy, and can automatically distribute the system load. It is very popular in big data implementation technology. On the basis of this high-quality storage architecture, the application of ELK (Elasticsearch, Logstash, Kibana) technology makes the application of data more perfect [13].

ELK Stack [14] is the abbreviation of the software collection Elasticsearch, Logstash, and Kibana. These three layers of software and their related combinations can create a large-scale real-time log processing system. Among them, Elasticsearch [15] is a Lucene-based distributed storage and indexing engine that supports full-text indexing. It is mainly responsible for indexing and storing logs to facilitate retrieval and query during program applications. Logstash is an intermediary component for log collection, filtering, and forwarding. It is mainly responsible for collecting and filtering various logs of various applications, and forwarding them to Elasticsearch for further processing [16]. Finally, Kibana [17] is a visualization tool that is mainly responsible for querying Elasticsearch data and presenting it to applications in a visual way, such as GUI generating various pie charts, bar charts, distribution charts, etc. [18].

In short, ELK is a system architecture for collecting, collapsing, querying, and analyzing log files, not a set of software. No complicated manual processing is required, as long as the rules are written in advance, log files of different sources and types can be automatically collected, and the results are stored in a common space. After analysis, these log files are counted and presented in a visual way. Process overview. The ELK components are shown in Figure 2.

2.3 Smart Network Packet Behavior Detection and Data Analysis

The botnet formed by various cyber crimes based on Peer to Peer (P2P) has become one of the main threats in network security [19]. Although some detection methods claim to be effective in detecting centralized botnets, they are still powerless in the problem of detecting P2P botnets. Therefore, in this research project, intelligent network packet behavior detection (a system based on intranet behavior analysis based on network packet flow) is used to detect P2P botnets [20].

The research will start from the detection of P2P network traffic behavior, the connection and diffusion behavior between bot hosts, and analysis to detect potential botnets. Even if the attackers take precautions against the monitoring technology in advance, they try to make P2P zombie hosts imitate the behavior of legitimate P2P application programs to evade the system. Relevant literature points out that Peer to Peer can obtain a higher detection rate, and the false alarm rate is extremely low, and more efficient intelligence information is proposed [21].

2.4 Intelligent Software-Defined Networking

Software Defined Networking (SDN) is a new type of network architecture. Use the OpenFlow protocol [22] to separate the router's control plane (control plane) from the data plane (data plane), and implement it in software. This architecture allows network administrators to re-plan the network with a central control method without changing the hardware device, which provides a new solution for controlling network traffic and also provides a new solution for core network and application innovation. Both Facebook and Google use the OpenFlow protocol in their data centers and have established the Open Network Foundation to promote this technology. Intranet traffic is controlled and managed by the SDN switch [23], and then an independent controller issues diversion commands to the SDN switch to achieve fast and flexible network device control, and can be carried out by copying network traffic to the ELK system Processing and calculation, and finally fed back to the SDN switch for alarm or blockade.

2.5 Federated Learning

Federated learning [24] is a decentralized machine learning framework where owners of datasets can collaborate to train a global model for a given task, such as classification, prediction, or regression. In each round of communication, a subset of participating users is selected to perform local training on their own data. Instead of sending raw data to a central server, the owners of the data upload their trained model parameters to the cloud. After aggregating local updates, the central server distributes the updated global model to another subset of model users. This process continues iteratively until a stopping criterion is met. In federated learning [25], dataset owners collaborate to train models without exchanging raw data, and no dataset owner can infer the private information of other dataset owners.

The primary goal of federated learning is to ensure that the performance of federated learning models approaches the performance of models trained using centralized methods while also protecting the privacy of each data owner.

Elasticsearch
Store, Add, Read, Modify, Delete
ELK
Logstash
Collect, Filter, Transmit
Kibana
Graphical representation

Figure 2. ELK components

3 Proposed Approach

3.1 Architecture

This research focuses on using ELK to process these huge amounts of L2, L3, and L4 packet header data. Cooperate with the DDOS detection system to find the problematic internal host, so that subsequent solutions can be solved through devices with OpenFlow functions and SDN technology. Finally, supplemented by the deep learning theory of artificial intelligence, the imported device is systematically analyzed to automatically set its permissions and security levels. With appropriate resources, the equipment can be quickly introduced and operated efficiently. The research structure is shown in Figure 3.

3.2 Smart Network Packet Behavior Detection and Data Analysis

The architecture must ensure that internal traffic passes through SDN switches to facilitate traffic detection and network control. First, the traffic is copied to the ELK system for storage through the SDN switch, where only the packet header (Header), that is, L2, L3, and L4 information, is retained. Secondly, intelligent network packet behavior detection is used to analyze whether there are abnormal behaviors in this network flow information. This system is based on federated learning and can detect whether SDN is under DDOS attack. When an abnormal host is found, the third step is to notify the SDN control system of IP and Mac information. Finally, the SDN control system automatically sends a block Mac command (the IP may change), blocks the traffic of the abnormal host first, and simultaneously notifies the network administrator to handle it.

3.3 Network Packet Behavior Detection

We have proposed a federated learning-based approach, employing the LSTM model, aimed at effectively addressing the challenges of DDoS attack detection. The dataset we used is CIC-DDoS2019, released by The Canadian Institute for Cybersecurity (CIC), which covers flow features of 13 types of DDoS attacks and benign samples. To maximize user privacy protection, our detection method does not utilize sensitive information such as port and IP.

The training process of this approach follows the following steps: Firstly, the server sends model parameters to each local client. Subsequently, local clients train the model locally and send the updated parameters back to the server. Finally, the server utilizes the FedAvg [26] algorithm to aggregate models from each training round and distributes the updated model parameters to all local clients. This mechanism of federated learning can be observed in Figure 3.

The challenges of FL include handling heterogeneous data distributions across clients, ensuring communication efficiency, maintaining model performance despite limited client participation, and addressing privacy concerns.

3.4 Mechanism

The overall system mechanism is divided into four layers. Network flow control layer is composed of OpenFlow Controller and Switch, which is responsible for blocking (Deny) or allowing (Allow) traffic forwarding; Packet filtering and storage layer is responsible for the ELK system, which will the copied packets filtered, the required information is extracted and stored in a regularized manner, and even tags or corresponding information (Mapping) are created; the Packet Behavior Detection Layer is dominated by the analysis model, which is responsible for continuous inspection and calculation of whether There is a host abnormality. When there is an abnormality, blocking rules will be generated and fed back to the switch according to the Response Processing Layer.

The study has developed a smart network packet behavior detection and response mechanism, regarding Figure 4, primarily comprising the following stages:

1) Build an OpenFlow switch.
2) Build a Web System manage the switch, and issue OpenFlow commands.
3) Build a big data processing system based on ELK architecture.
4) Apply the intelligent packet behavior detection model to the ELK system.
5) The switch must follow the OpenFlow instructions and copy specific traffic to the ELK system while maintaining normal traffic services.
6) Find the abnormal host list (Mac list) through intelligent packet behavior detection.
7) Follow steps 2 and 5 to block the corresponding traffic.

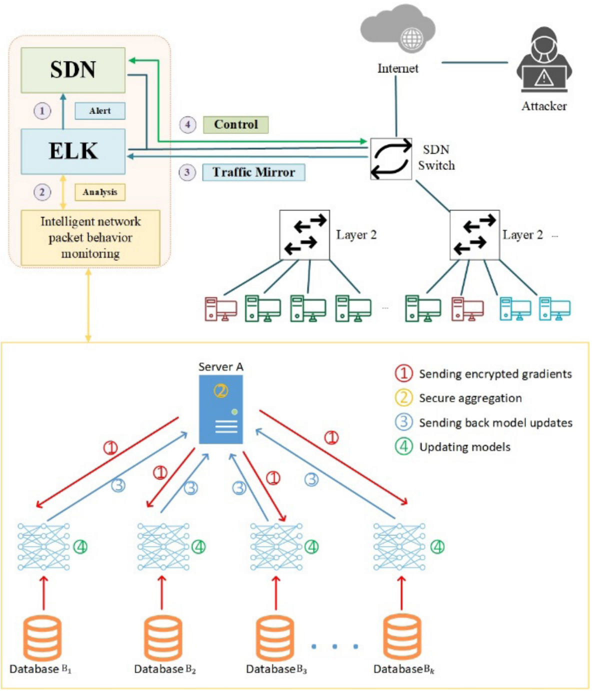

Figure 3. Architecture of the proposed approach

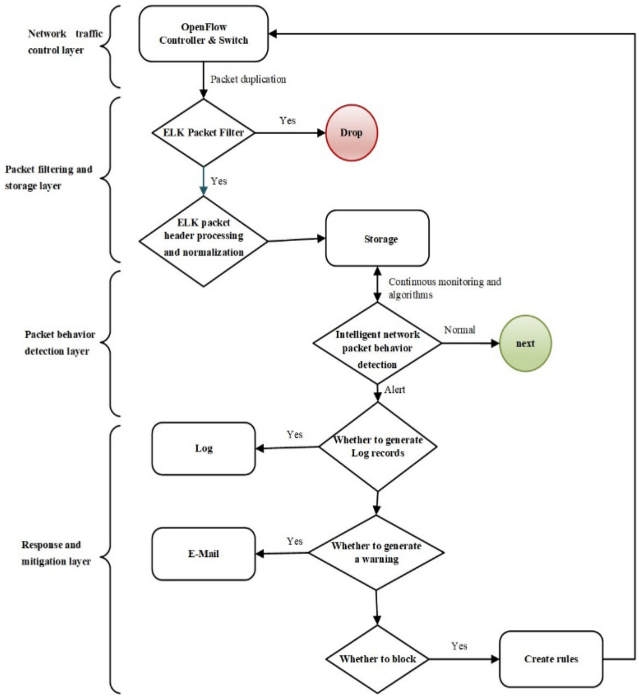
Network traffic control layer
Packet filtering and storage layer
Packet behavior detection layer
Response and mitigation layer
OpenFlow Controller & Switch
Packet duplication
ELK Packet Filter
Yes
Drop
Yes
ELK packet header processing and normalization
Storage
Continuous monitoring and algorithms
Intelligent network packet behavior detection
Normal
next
Alert
Whether to generate Log records
Yes
Log
Whether to generate a warning
Yes
E-Mail
Whether to block
Yes
Create rules

Figure 4. System mechanism flow chart

4 Experiment

4.1 Test Architecture and Authority Division

This research aims to verify the control of network connections through SDN in a network environment. In the network architecture, User1 and User2 are in the same office, and access the Internet through the office SDN Switch to capture and analyze network behavior. Architecture As shown in Figure 5.

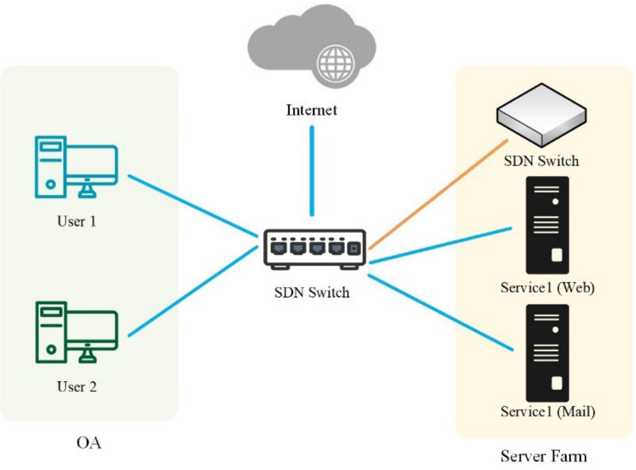

Figure 5. Research testing framework

Four devices are prepared in the architecture, two of which are servers (Service) and two are end users (users). The distribution is as follows in Table 1.

Table 1. Network component list

Device	Role	Explanation
Service1	Mail server	MALWARE actions will be communicated laterally to User2.
Service2	Mail server	MALWARE behavior will communicate laterally to another third-party website.
User1	MIS network administrator	MALWARE actions will be communicated laterally to Service2.
User2	Office staff	MALWARE behavior will communicate laterally to another third-party website.
The communication between User2, service2 and the third website forms a triangular communication. Therefore, it will be enhanced security filtering by the United Defense Force or directly rejected.		

4.2 Behavioral Statistics

In the simulation environment, the network usage of each User is recorded through SDN, recorded every 5 seconds, and observed for 10 minutes (600 seconds). These observations can be made by referring to Figure 6 and Figure 7. For User1 (MIS network administrator), the traffic records using SSH (blue) can be observed. During the process, files are transferred through SFTP, so the traffic increases slightly. Using the traffic record of the HTTP web page (orange), there is continuous traffic. The SMTP service (green) is not used, so the SMTP traffic is 0.

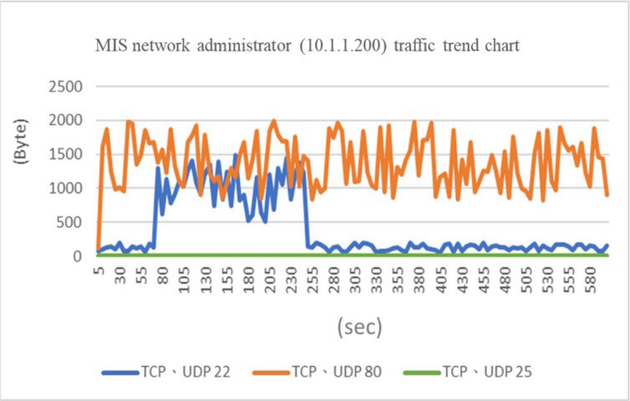

Figure 6. MIS network administrator (10.1.1.200) traffic trend chart

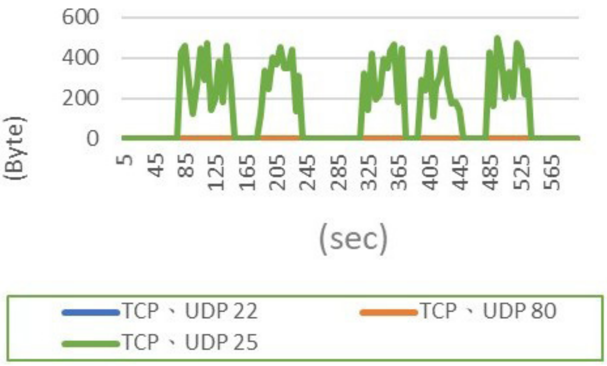

Figure 7. Office staff (10.1.1.210) traffic trend chart

4.3 Detection Result

In our experiment, we implemented a collaborative training classifier using the LSTM model with the PyTorch framework in a conda DL development environment. During the training process, the datasets of the workers may contain different types and quantities of attack samples, allowing our model to address the diversity of real-world scenarios comprehensively (see Figure 8).

We set the parameters of the experiment to train for 100 epochs, with a batch size of 128 and a learning rate set to 0.01. Through evaluation of the CICDDOS2019 dataset, our approach achieved a satisfactory accuracy of 92%. This demonstrates the effectiveness and applicability of our collaborative training classifier.

Figure 9 represents the model's loss rate, while Figure 10 represents the model's accuracy. These two graphs illustrate the model's performance, with Figure 10 reaching convergence by the 50th epoch and achieving an accuracy of 88.64%.

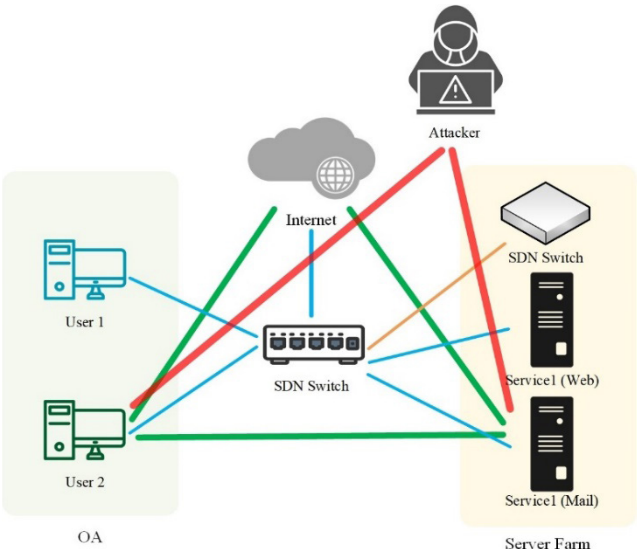

Figure 8. Triangle communication network behavior diagram of abnormal network behavior

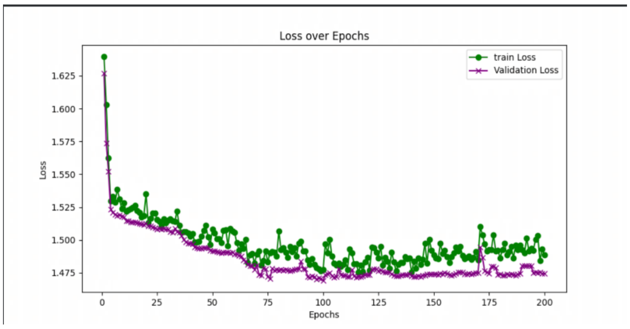

Figure 9. Performance of the proposed strategy in terms of loss rate

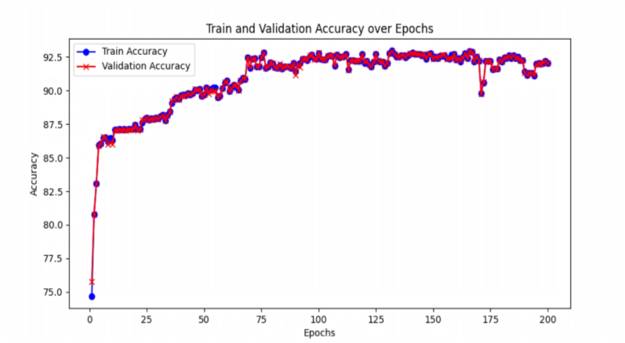

Figure 10. Performance of the proposed strategy in terms of accuracy

We compared our method with centralized training using the CICDDOS2019 dataset to evaluate accuracy performance. We can found that Figure 11 that centralized training achieved an accuracy of 94%, while our method achieved 92%. Despite ensuring privacy preservation, our method maintains an accuracy close to that of centralized training, demonstrating a strong balance between privacy protection and accuracy.

By applying Dirichlet distribution [27] to our dataset, we introduced an imbalanced data distribution to evaluate our model's performance under different data conditions. Specifically, we used Dirichlet distributions with parameters of 0.9 and 0.3 to distribute the data among 10 clients, denoted as α, where α = 0.9 and α = 0.3. A larger alpha value results in a more concentrated distribution, while a smaller value leads to a more dispersed distribution. The results are summarized in Table 1. As observed in Table 2, even under different alpha values (indicating varying levels of data dispersion), our method consistently ensures high accuracy for each client without any significant drop in performance. This demonstrates the excellent performance of our method in personalized approaches.

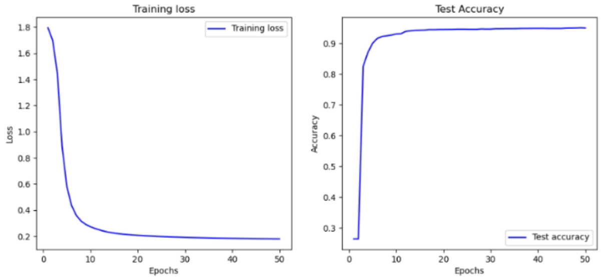

Figure 11. Performance of the centralized training

Table 2. Our proposed strategy evaluates the model's accuracy across different clients under various alpha values

	α=0.9	α=0.3
Worker1	92.87	82.24
Worker2	91.82	96.55
Worker3	91.07	89.63
Worker4	94.8	99.24
Worker5	94.59	99.91
Worker6	85.78	94.38
Worker7	95.86	98.59
Worker8	99.25	95.71
Worker9	88.29	96.19
Worker10	86.48	86.4
Avg	92.08	93.89

5 Conclusion

This study employs SDN technology to record and analyze the network behaviors of hosts, serving as a complement to traditional network behavior analysis. In addition to employing policy methods specified in network management rules to determine traffic forwarding or blocking, it also integrates deep learning and federated learning to detect and analyze Distributed Denial of Service (DDoS) attacks on the organization's internal network. Moreover, the application of federated learning ensures data privacy. When abnormal access behaviors of internal hosts are detected, the system issues alerts and automatically deploys blocking commands via SDN to prevent unauthorized intrusion attempts. These mechanisms collectively enhance the overall network security posture. However, today's landscape of network attacks is highly diverse, and merely detecting DDoS attacks falls short. Therefore, in the future, we aim to utilize such a mechanism to detect a broader range of network attacks in IoT devices and evaluate the performance of our proposed model on alternative datasets.

Acknowledgments

The authors would like to express their sincere thanks for the support from the National Science Council of the R.O.C. under grants 112-2221-E-259 -004 -MY2 and 112-2622-E-259 -001 -.

References

[1] I. Rafiq, A. Mahmood, S. Razzaq, S. H. M. Jafri, I. Aziz, IoT applications and challenges in smart cities and services, *The Journal of Engineering*, Vol. 2023, No. 4, Article No. e12262, April, 2023. https://doi.org/10.1049/tje2.12262

[2] P. Charanarur, B. T. Hung, P. Chakrabarti, S. S. Shankar, Design optimization-based software-defined networking scheme for detecting and preventing attacks, *Multimedia Tools and Applications*, Vol. 83, No. 28, pp. 71151–71169, August, 2024. https://doi.org/10.1007/s11042-024-18466-8

[3] A. K. Jain, H. Shukla, D. Goel, A comprehensive survey on DDoS detection, mitigation, and defense strategies in software-defined networks, *Cluster Computing*, Vol. 27, No. 9, pp. 13129-13164, December, 2024. https://doi.org/10.1007/s10586-024-04596-z

[4] H. B. Bae, M. W. Park, S. H. Kim, T. M. Chung, Zombie pc detection and treatment model on software-defined network, in: J. Park, I. Stojmenovic, H. Jeong, G. Yi (Eds.), *Computer Science and its Applications: Ubiquitous Information Technologies*, pp. 837–843, 2015. https://doi.org/10.1007/978-3-662-45402-2_119

[5] V. Cerf, R. Kahn, A protocol for packet network intercommunication, *IEEE Transactions on communications*, Vol. 22, No. 5, pp. 637–648, May, 1974. https://doi.org/10.1109/TCOM.1974.1092259

[6] S. U. Rehman, S. Manickam, N. F. Firdous, Impact of dos/ddos attacks in iot environment: A study, *AIP Conference Proceedings*, Vol. 2760, No. 1, Article No. 020020, June, 2023. https://doi.org/10.1063/5.0150000

[7] J. Li, Z. Zhang, Y. Li, X. Guo, H. Li, Fids: Detecting ddos through federated learning based method, *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Shenyang, China, 2021, pp. 856–862. https://doi.org/10.1109/TrustCom53373.2021.00121

[8] K. Srinarayani, B. Padmavathi, D. Kavitha, Detection of botnet traffic using deep learning approach, *2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, Erode, India, 2023, pp. 201–206. https://doi.org/10.1109/ICSCDS56580.2023.10104633

[9] K. J. Higgins, Srizbi botnet sending over 60 billion spams a day, May, 2008. [Online]. Available: https://www.darkreading.com/cyber-risk/ srizbi-botnet-sending-over-60-billion-spams-a-day

[10] A. Gazet, Comparative analysis of various ransomware virii, *Journal in computer virology*, Vol. 6, No. 1, pp. 77–90, February, 2010. https://doi.org/10.1007/s11416-008-0092-2

[11] H. Kanaker, N. Salameh, S. A. Awwad, N. H. Ismail, J. Zraqou, A. M. F. Al Ali, Trojan horse infection detection in cloud based environment using machine learning, *International Journal of Interactive Mobile Technologies*, Vol. 16, No. 24, pp. 81–106, 2022. https://doi.org/10.3991/ijim.v16i24.35763

[12] J. Nandimath, E. Banerjee, A. Patil, P. Kakade, S. Vaidya, D. Chaturvedi, Big data analysis using apache Hadoop, *2013 IEEE 14th International Conference on Information Reuse & Integration (IRI)*, San Francisco, CA, USA, 2013, pp. 700–703. https://doi.org/10.1109/IRI.2013.6642536

[13] P.-C. Tung, H.-Y. Hsieh, Use elk to visually analyze information security equipment logs, *TANET 2019 Taiwan Internet Seminar*, Kaohsiung, Taiwan, 2019, pp. 546–551. https://doi.org/10.6924/TANET.201909.0099

[14] S. J. Son, Y. Kwon, Performance of elk stack and commercial system in security log analysis, *2017 IEEE 13th Malaysia international conference on communications (MICC)*, Johor Bahru, Malaysia, 2017, pp. 187–190. https://doi.org/10.1109/MICC.2017.8311756

[15] O. Kononenko, O. Baysal, R. Holmes, M. W. Godfrey, Mining modern repositories with elasticsearch, *Proceedings of the 11th working conference on mining software repositories*, Hyderabad India, 2014, pp. 328–331. https://doi.org/10.1145/2597073.2597091

[16] S.-W. Yang, C.-D. Yang, C.-J. Chen, Implement ETC open data visualization and traffic flow analysis using elk stack environment, *TANET2017 Taiwan Internet Seminar*, Taichung, Taiwan, 2017, pp. 973–978. https://doi.org/10.6728/TANET.201710.0169

[17] V. Sharma, Getting started with kibana, in: *Beginning Elastic Stack*, Apress, Berkeley, CA, 2016, pp. 29–44. https://doi.org/10.1007/978-1-4842-1694-1_3

[18] C.-D. Yang, C.-J. Chen, S.-H. Cao, Y. Sun, G.-Q. Zhang, C.-L. Lai, Implement a campus wireless network log analysis system using the elk stack environment, *TANET 2016 Taiwan Internet Seminar*, Hualien, Taiwan, pp. 628–633. https://doi.org/10.6679/TANET.2016.113

[19] D. Zhuang, J. M. Chang, Peerhunter: Detecting peer-to-peer botnets through community behavior analysis, *2017 IEEE Conference on Dependable and Secure Computing*, Taipei, Taiwan, 2017, pp. 493–500. https://doi.org/10.1109/DESEC.2017.8073832

[20] K. Singh, S. C. Guntuku, A. Thakur, C. Hota, Big data analytics framework for peer-to-peer botnet detection using random forests, *Information Science*, Vol. 278, pp. 488–497, September, 2014. https://doi.org/10.1016/j.ins.2014.03.066

[21] D. Zhuang, J. M. Chang, Enhanced peerhunter: Detecting peer-to-peer botnets through network-flow level community behavior analysis, *IEEE Transactions on Information Forensics and Security*, Vol. 14, No. 6, pp. 1485–1500, June, 2019. https://doi.org/10.1109/TIFS.2018.2881657

[22] A. Bianco, R. Birke, L. Giraudo, M. Palacin, Openflow switching: Data plane performance, *2010 IEEE International Conference on Communications*, Cape Town, South Africa, 2010, pp. 1–5.

https://doi.org/10.1109/ICC.2010.5502016

[23] G. Callet, J. Faraj, O. Jardel, C. Charbonniaud, J.-C. Jacquet, T. Reveyrand, E. Morvan, S. Piotrowicz, J.-P. Teyssier, R. Quéré, A new nonlinear hemt model for algan/gan switch applications, *International Journal of Microwave and Wireless Technologies*, Vol. 2, Special Issue 3-4, pp. 283–291, August, 2010. https://doi.org/10.1017/S1759078710000541

[24] T. Zhang, S. Mao, An introduction to the federated learning standard, *GetMobile: Mobile Computing and Communications*, Vol. 25, No. 3, pp. 18–22, September, 2021. https://doi.org/10.1145/3511285.3511291

[25] E. C. P. Neto, S. Dadkhah, A. A. Ghorbani, Collaborative ddos detection in distributed multi-tenant iot using federated learning, *19th Annual International Conference on Privacy, Security Trust (PST)*, Fredericton, NB, Canada, 2022, pp. 1–10. https://doi.org/10.1109/PST55820.2022.9851984

[26] W. Zhang, T. Zhou, Q. Lu, Y. Yuan, A. Tolba, W. Said, Fedsl: A communication-efficient federated learning with split layer aggregation, *IEEE Internet of Things Journal*, Vol. 11, No. 9, pp. 15587–15601, May, 2024. https://doi.org/10.1109/JIOT.2024.3350241

[27] Y. Wu, S. Zhang, W. Yu, Y. Liu, Q. Gu, D. Zhou, H. Chen, W. Cheng, Personalized federated learning under mixture of distributions, *International Conference on Machine Learning*, Honolulu Hawaii USA, 2023, pp. 37860–37879.

Biographies

Shih-Chen Wang received his B.S. and M.S degrees in Transportation and Communication Management from the National Cheng Kung University, Tainan, Taiwan in 1996 and 1998, and the Ph.D. degree in the Department of Engineering Science from National Dong Hwa University, Hualien, Taiwan, in 2024. He is currently an Ph.D student at the Department of Computer Science and Information Engineering, National Dong Hwa University, Hualien, Taiwan. His research interests include software-defined networks, system design, and AIoT

Yi-Chen Lee is a master's student in the Department of Computer Science and Information Engineering at National Dong Hua University. Her research interests focus on deep learning, exploring the design and optimization methods of deep neural networks. Recently, her research has focused on improving federated learning. She is exploring methods to enhance the performance and scalability of federated learning systems through optimizing model aggregation algorithms and improving the accuracy of participant model updates. Additionally, she is studying potential applications of federated learning across various domains.

Wei-Che Chien received his B.S. and M.S degrees in Computer Science and Information Engineering from the National ILan University, I-Lan, Taiwan in 2014 and 2016, and the Ph.D. degree in the Department of Engineering Science from National Cheng Kung University, Tainan, Taiwan, in 2020. He is currently an Assistant Professor with the Department of Computer Science and Information Engineering, at National Dong Hwa University, Hualien, Taiwan. His research interests include wireless rechargeable sensor networks, 5G mobile networks, AIoT, fog computing, and cloud computing.

Guanling Lee received the Ph.D. degree in computer science from National Tsing Hua University Taiwan in 2001. She joined National Dong Hwa University in the Department of CSIE, and became an associate professor in 2005. Her research interests include resource management in the mobile environment and data mining.

Sheng-Lung Peng is a Professor at the Department of Creative Technologies and Product Design, and the Dean of the College of Innovative Design and Management, National Taipei University of Business in Taiwan. He received the PhD degree from Computer Science Department of National Tsing Hua University in Taiwan. He is an adjunct Professor at National Dong Hwa University in Taiwan and Kazi Nazrul University in India. He is an Honorary Adjunct Professor at Sir Padampat Singhania University and Baroda University, India. He serves as the president of the Association of Taiwan Computer Programming Contest and the Association of Algorithms and Computation Theory. He is a co-director of the ICPC Asia Pacific, and a director of the Institute of Information and Computing Machinery and the Taiwan Association of Cloud Computing. He is also a supervisor for the Chinese Information Literacy Association. His research interests are algorithm design in the fields of artificial intelligence, bioinformatics, combinatorics, data mining, and networking.