Security and risk analysis in the cloud with software defined networking architecture

Venkata Nagaraju Thatha[1], Swapna Donepudi[2], Miriyala Aruna Safali[3], Surapaneni Phani Praveen[2], Nguyen Trong Tung[4], Nguyen Ha Huy Cuong[5]

[1]Department of Information Technology, MLR Institute of Technology, Hyderabad, India
[2]Department of Computer Science and Engineering, Prasad V Potluri Siddhartha Institute of Technology, Vijayawada, India
[3]Department of Computer Science and Engineering, Dhanekula Institute of Engineering and Technology, Vijayawada, India
[4]Office of Quality Assurance, Dong A University, Dang Nang City, Vietnam
[5]Software Development Centre, The University of Danang, Dang Nang, Vietnam

Article Info

Article history:

Received Dec 30, 2022
Revised Mar 28, 2023
Accepted Apr 7, 2023

Keywords:

Cloud computing
Distributed denial of service attack
secCloud
Software defined network
Virtual network

ABSTRACT

Cloud computing has emerged as the actual trend in business information technology service models, since it provides processing that is both cost-effective and scalable. Enterprise networks are adopting software-defined networking (SDN) for network management flexibility and lower operating costs. Information technology (IT) services for enterprises tend to use both technologies. Yet, the effects of cloud computing and software defined networking on business network security are unclear. This study addresses this crucial issue. In a business network that uses both technologies, we start by looking at security, namely distributed denial-of-service (DDoS) attack defensive methods. SDN technology may help organizations protect against DDoS assaults provided the defensive architecture is structured appropriately. To mitigate DDoS attacks, we offer a highly configurable network monitoring and flexible control framework. We present a dataset shift-resistant graphic model-based attack detection system for the new architecture. The simulation findings demonstrate that our architecture can efficiently meet the security concerns of the new network paradigm and that our attack detection system can report numerous threats using real-world network data.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Nguyen Ha Huy Cuong
Software Development Centre, The University of Danang
Dang Nang, Vietnam
Email: nhhcuong@sdc.udn.vn

1. INTRODUCTION

SDN is a transformative technique for network design and implementation that focuses on decoupling network functions (NFs) control from networking devices (load balancers, firewalls, switches, routers) [1]. The open flow protocol allows software-defined networking (SDN) switches to take advantage of the flexibility provided by the ability to access header information from different open system interconnections (OSI) stack layers, allowing it to satisfy the traditionally fulfilled functionalities through a physical device multitude. The flexibility made SDN an excellent platform for multi-tenant data center deployments that needed dynamism and flexibility thanks to the inclusion of SDN programmable network interfaces. This was particularly true in the infrastructure-as-a-service (IaaS) model, where tenants wanting technological and financial flexibility maintained virtual machines (VMs) [2].

Figure 1 represents and describes the SDN design in light of the SDN structural model presented in [3]. The information plan contains the management and planning teams. This part executes management approaches that meet the organization's manager's goals. It would be best if you had a justification of choice, which is intended for fast shipping. Packets that do not match one of the approaches are dropped or transported to the control plane via the southbound application programming interface (API). Southbound API is a free correction of the guidelines implemented in the hardware management information plan. Improve bidirectional correspondence between information and control planes.

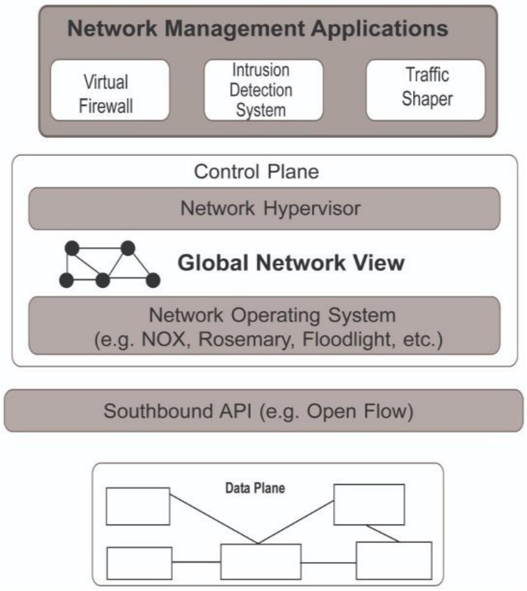
Network Management Applications
Virtual Firewall
Intrusion Detection System
Traffic Shaper
Control Plane
Network Hypervisor
Global Network View
Network Operating System
(e.g. NOX, Rosemary, Floodlight, etc.)
Southbound API (e.g. Open Flow)
Data Plane

Figure 1. High-level overview of the SDN architectural model

Privacy and security are the primary concerns of data center administrators, cloud service providers (CSP), and cyber security. The cloud network infrastructure plays a critical role in ensuring the security of cloud domain resources in data centers by providing network segmentation, virtual private networks (VPNs), network monitoring, firewalls, and IDS/IPS, load balancers, and traffic shaping to protect against various security threats. SDN [4] is a networking architecture that separates the control plane from the data plane in a network. In traditional networking, the control plane and data plane are tightly coupled, meaning that network devices such as switches and routers perform both functions. The control plane is responsible for making decisions about how traffic should be forwarded through the network, while the data plane is responsible for actually forwarding the traffic. The most rapidly adopted paradigm in modern data centers is network function virtualization (NFV) [5] which allows virtualized functions and network services as VNFs. In traditional networking, each tenant or user would have their own physical network infrastructure, which could be difficult and costly to manage. With SDN, however, network resources can be virtualized and shared among multiple tenants, allowing for more efficient use of network resources and easier management.

SDN provides more efficiency and flexibility application by allowing the centralized controller to monitor everything. The dynamic nature of SDN also enables the rapid deployment and reconfiguration of security policies and controls to protect against cyber-attacks. For example, the centralized controller can quickly detect and isolate suspicious traffic, redirect it to a honeypot, or implement security policies to protect against distributed denial-of-service (DDoS) attacks [6]. Such DDoS attack targets, including any CSP from larger cloud enterprises to smaller private-scale campus clouds, lead to billions of dollars in damage to the tenants and cloud providers. The most basic volumetric DDoS attack targets resources of the Cloud, and the attacker gains complete or partial regular user services disruption quality of experience (QoE) through flooding the VM hosts the services with numerous packet volumes [7].

A cloud SDN security framework is presented, and a security model is implemented with an attack-detecting approach in the data plane and control of mitigation in the SDN control plane by this research work. Experiments prove that only a marginal variation in processing cost to the cooperative security approach in SDN. Moreover, this approach protects SDN architecture from getting into the saturation of control-plane, flow-table/miss attacks in the network, and defending middle-box appliances and downstream servers. In the data plane, packets can be processed, and switches must be allowed with the processing functions of new packets to detect DDoS coarse-grained attacks and mitigation actions. An SDN-integrated cloud managing system contains data plane security monitoring and analyzing control plane threats. These approach evaluations prove that the extensible data plane of stately SDN in the model with the NF service chain gives greater security compared to classical solutions of perimeter/firewall. Security and risk analysis in the cloud with SDN architecture is important to ensure the security and availability of cloud systems and protect against various security threats. Contributions are summarized as follows: i) A stateful/security-aware SDN data plane is implemented; therefore, certain less-weight computation/detection functions can be offloaded to switches to processing in-line; ii) The OvS data plane stack is implemented by data plane development kit (DPDK), which contains network interface card (NIC) APIs/libraries/drivers, to process high-speed data packets. The flow-analysis pipeline process throughput is exceptionally high in switch due to the acceleration with DPDK, faster path kernel processing; and iii) As a result of these enhancements, the controller processing power and network nodes throughput is freed up to other functions

The upcoming sections of this article are structured as follows. Section 2 describes the relevant works that discuss the technical gaps associated with the state-of-the-art models. Section 3 briefs the methodology and internal workings of the framework. Section 4 details the validity and robustness of the proposed method based on experimental results. Section 5 gives the conclusion statement.

2. RELATED WORK

The cloud-security assessment and several studies investigate auditing methodologies. Ghosh *et al.* [8] presented a framework based on SDN to guarantee performance and security in information-centric cloud networks. Cloud providers' primary challenge is using network resources addressed through virtual provisioning networks, which can facilitate information services by establishing a barrier between the control plane and the cloud environment. Further, this will compute the path to provide security and network performance. Initial experiment over average round trip delay among producers and customers is reported. Farahmandian and Hoang [9] designed an software-defined security service (SDS2) to protect cloud domains. This SDS2 mainly focuses on defining the concerns of security related to the virtual and physical boundaries of tenants, resources, data, and detection of security breaches via limits violations.

SDS2 and its initial implementation are described in this work. In addition, this can give policy-defined boundary examples and exhibits the feasibility and effectiveness of this design while detecting the invisible security boundaries by simulating the security control architecture and dynamic, intelligent VSFs. Pisharody *et al.* [10] described how a security policy is checked in distributed clouds based on SDN. In SDN, Separating the network control from devices offers the implantation of centralized network and security policy management in CC infrastructure.

Seeber and Rodosek [11] described enhanced network security by SDN in cloud environments. In the past, services and single systems' security were widely treated. Cloud services and systems need a most detailed security requirements observation and their satisfaction because services and systems coexist over one virtualization layer without becoming aware of other layers' systems [12]. The basic goal is to keep a centralized database organized logically. That provides each system's most recent security-related information or product. Using this knowledge model, which is referred to as a system security rating and security specifications are given through the reconfiguration of system operators and cloud service providers for the network to meet each system's security requirements [13].

In [14], [15] presented a framework based on analytic hierarchy process (AHP) for quantitatively comparing, benchmarking, and ranking the level of security provided through various CSPs depending on its security level agreements (SLA) based on the security requirements of cloud users. Nguyen *et al.* [16] described an approach for assessing user satisfaction with the provided cloud service with two major stages: a first stage is a conceptual approach that contains various attributes like adaptability, cost, performance, security, and efficiency. The second stage is a fuzzy inference system (FIS) structure that includes five significant rules and 11 inputs (attributes) [17].

The SDN paradigm allows network control programming and infrastructure abstraction for applications and network services [18]. It uses control and data plane abstractions. The former governs network programming and management (i.e., routing logic), whereas the latter is the interconnected virtual or physical network infrastructure of switches, routers, and other network equipment [19]. These devices process packets according to control plane rules. While the idea of control and data plane separation is present in Internet Engineering Task Force (IETF) working group works [20] and even earlier with the concept of programmable and active networks [21], [22], the 2008 OpenFlow work [23], [24] is considered the first appearance of SDN in modern literature [25]. These articles demonstrate the potential of combining SDN and DRL techniques for efficient resource allocation in fog computing environments. This approach has the potential to improve the performance and energy efficiency of fog computing applications and enable more effective use of cloud and edge resources [26]–[29].

3. METHOD

The framework of cloud network SDN for security and threat analytics architecture is shown in Figure 2. A solution is presented in this work based on SDN architecture for securing cloud infrastructure against DDoS attacks. It can identify DDoS attackers between legitimate users and, in real time, blocks them from stabilizing the system. The primary objective is to design a holistic, automated defense monitoring architecture with faster attack detection. Since centralized (single point) defense does not eliminate the threats in more extensive networks, in multi-plane schemes, distributed security compliance and monitoring are controlled from a centralized location method used at strategic nodes in a network.

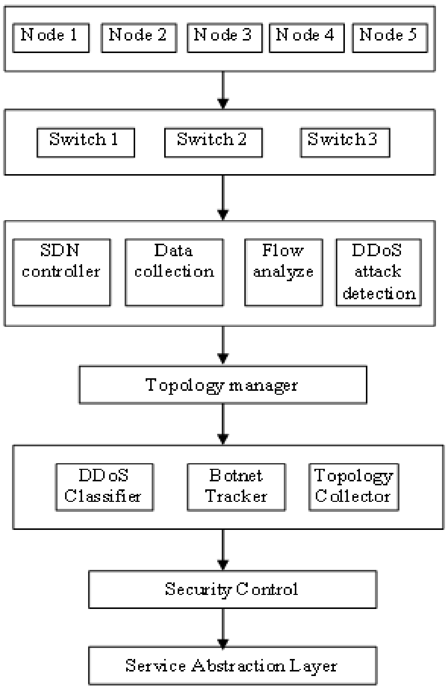

Figure 2. Framework of cloud network SDN

The control layer performs attack analysis overflows and controlling functions. The fine-grained anomalies must be identified by the controller, attacks, and developing more advanced protection mechanisms, using its comprehensive picture of the network as whole, rich tools for storing and processing data to analyze past events information. Whenever abnormal flow is observed (DDoS attack), a particular switch notices a relevant core switch. Optimize controller workload for responding fast against a DDoS attack and implement a coarse-grained attack-detecting algorithm and triggering schemes in the data layer hierarchy. Security proxy/middlebox switches act as a go-between for the controller. Activating the main switch causes certain events, using challenge- and-response proxies to identify DoS attacks or packet data plane handling with the right guidelines. When an appropriate set of actions is missing, the core switch will contact the edge switches for sampling if there is an abnormal flow. This causes unexpected fine-grained categorization in the control-plane assault detection technique with the extracted information about attack characteristics. Indicator of risk analytics system takes advantage of the control plane attack features uploaded classification of distributed denial of service (DDoS) attacks and the global topology of trying to pin down threat origins.

Switches: user calls for specific data attributes and sends the information to the controller by an open-flow protocol. The client’s request is processed, and “flow tables” are generated. The router revises the storage address of required metrics in a table of processes. Following the collection of these metrics, they are sent to the controller. The I.P. addresses of all clients are included in the flow table. It could save the I.P. address for each inquiry. The I.P. address of the source location and I.P. address in the cloud, and a counter representing the total number of requests were sent via the same remote server.

SDN controller: one of the most vital parts of any solution offered. These SDN switches can be used to receive packets. It has a liability of care to avoid any potential DDoS assaults on the cloud. For this purpose, after completing a series of operations (such as collecting information, data analysis, and attack detection). Whenever the SDN controller gets flow packets from the SDN switches, it makes available a collection of algorithms designed to identify DDoS attacks. The plane is outfitted with a specialized SDN controller (open daylight (ODL)) with added functionality to keep an eye on emerging safety, defense, and harm prevention during an assault. A summary, feature-digest, and in-band message can be used to categorize how global in scope the network topology and the type of attack are and potential dangers. After then, it calls to the library of defensive actions and changes in the field to establish customized defensive measures along the line of fire or close to the origin of the assault. The "modular layer 2 (ML2)" plugin implemented a generic API as a "plug-and-play" driver. Driver's strategy is utilized in ODL and open v switch (OVS). The plugin performs all networking services ("creation, updating, and deletion of networks, subnets and port resources, port binding"). It connects the cloud controller and VMs to the outside network. Already open stack has adopted certain SDN networking function implementations.

For optimizing and securing the deployments of open stack cloud, native SDN elements are developed, and interfacing modules to the basic legacy switches. Based on the virtual switches, OvS are used in more than 60% of SDN/NFV-allowed applications. Data centers, which serve as baseline additions, are implemented. The SDN stack's security strategy is detection and reaction. The control plane and data used during the detection stage offered a less weight anomaly detection on the plane. Statistical-based serving flow monitoring algorithm acts like a DDoS assault sensor of the data plane. The volume of DDoS attacks will show up in greater numbers and asymmetry to the network; these characteristics can be looked over to spot attackers.

A new method for unloading defense mechanisms is proposed. They were activating actuators for defense against DDoS attacks using the SDN's peripheral and central switches. The SDN controller is free from performing particular defensive actions, which results in attack-reacting efficiencies and optimizes total load traffic. Primarily concentrate on exploiting the switch central processing unit (CPU) computational resources and southbound interface flexibility for deploying defense actuator NFs over switches nearer to the botnet. The SDN switch updates the request properties (counter, internet protocol (IP) addresses of cloud, and source) for every new client from flow tables. Next, these properties are transmitted to the controller. Later it takes statistics from all the connected switches for storing them in a worldwide flow table. After collecting data, the controller supervises traffic evaluation by its worldwide flow table using the flow count column.

Administrators in the cloud limit the total flow to prevent it from counting. If this total exceeds a certain limit during a given interval, the scheme given here initiates an attack detection method. The controller will request the cloud provider if the user does not enter a PIN. When a tenant requests resources for open stack, the Nova module provisions need VM examples in cloud. The open daylight SDN controller scheduled a virtual network (VN) via a RESTful call. The ODL calls the OVS database (OvSDB) and OpenFlow (OF) for configuring VNs and transmits flow rules to OF switches. The topology controller stores the network's configurations and topologies of VN in the cloud, this pool of resources might be reconfigured or provisioned based on the operational cases. An experimental SDN and OpenStack integration network incorporates a set of the anti-DDoS applications package

Figure 3 shows a significant level representation of an SDN organization engineering, extended with some parts, to be precise, constant strategy control and disconnected fixes control. Verification of the ongoing strategy: as the name suggests, the fundamental task of this part is to conduct a continuous and persistent confirmation of the organization's corrections. Furthermore, this party is responsible for initiating the labeling of strategies so that the responsible party can follow each given procedure. After continuous verification, the recognized agreements are transmitted to the regulator for the organization in the information plan devices. All information about inbound confirmed and discarded offers (including the starting point and execution settings) is stored in a database for occasional review by the offline strategy control.

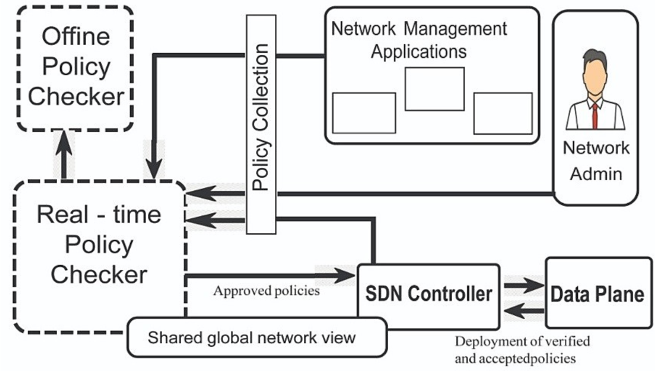
Offline Policy Checker
Policy Collection
Network Management Applications
Network Admin
Real - time Policy Checker
Approved policies
SDN Controller
Data Plane
Shared global network view
Deployment of verified and acceptedpolicies

Figure 3. High-level overview of the proposed SDN policy verification framework

Offline strategy verification: unlike real-time approach verification, offline correction verification performs static and intermittent confirmation of approaches. While this cannot prevent strategy conflicts from occurring in every case, the disconnected fix checker can occasionally conduct internal and external investigations into the state of the organization concerning invariants. The motivation of the disconnected availability checker is to approve correlative organization properties such as activity, network accessibility, and disconnection of occupants, as well as to examine transmission network applications for retaliatory moves. In this way, the disconnected strategy check can work much like an interrupt identification framework but with an emphasis on recognizing malicious solutions.

In addition to the verification and authorization strategy, the framework must remember several components for applying a reliable and secure SDN framework. An input control model explicitly intended to guarantee a productive sandboxing of the organization’s applications and the separation between the various levels of honor; Systems for a reasonable settlement of the bases and readiness of aircraft information devices, isolation of powerful multi-dweller SDN agreements, and the need for strong participation for reliable and fair dissemination of goods among inhabitants are examples of such views.

4. RESULT ANALYSIS

An evaluation design strategy is designed with the perspective of security, and a set of network characteristics and key performance indicators (KPI) can be computed. To evaluate and perform the comparative study, every computing node is loaded with various network hypervisor switches: Linux bridge firewall (LBFW), native OvS firewall module, cloud SDN OvS security modules. Memory utilization: the collector collects the amount of memory by every running procedure within 30 seconds. For every 5 minutes interval, the collector computes these samples’ average value for the past 5 minutes to send it later to the Engine. The average memory usage is defined as the execution of average memory usage before being aggregated.

Table 1 indicates that the SDN OvS approach consumes more memory than legacy Linux bridge (LB) due to SDN/OVS OpenFlow pipeline tables. If the VMs/node increases, then all three schemes’ memory utilization is normalized to a level that is equal to or less than that is utilized through legacy LB Figure 4 describes the graphical representation of the memory utilization for different methods as LBFW, native OvS firewall module and cloud SDN firewall (FW). The memory utilization is higher for the cloud SDN firewall than for the remaining methods.

Table 1. Memory utilization

Nodes	Linux BRFW(%)	OvS classic FW (%)	Cloud SDN FW (%)
1	9.2	19.4	24.7
2	14.9	22.5	34.2
3	25.4	31.7	39.4
4	32.5	39.8	42.5
5	41.6	47.8	58.3

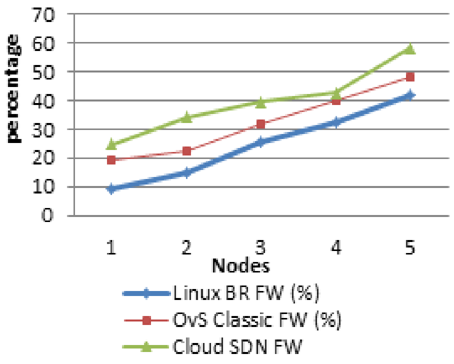

Figure 4. Memory utilization

CPU utilization: in CPU usage, the collector takes CPU load samples of all running processes every 30 seconds. The CPU sufficient utilization time in I aggregated execution can be known as CPU usage. The CPU load is computed in terms of percentage from 0 to 100 for every logical processor present in the device.

Hence CPU load would be greater than 100% to the devices with multiple logical processors. Table 2 represents the CPU utilization for different methods LBFW. module, native OvS FW module, and cloud SDN FW. All these three approaches are shown in Figure 5.

Table 2. CPU utilization

Nodes	Linux B.R. F.W.(%)	OvS Classic F. W. (%)	Cloud SDN FW (%)
2	6	6.1	6.5
4	6.4	6.3	6.7
6	6	6.2	6.4
8	10	10.4	10.8
10	10.5	10.7	11.1
12	11	11.5	11.8
14	11.4	11.7	11.9
	11	11.3	12

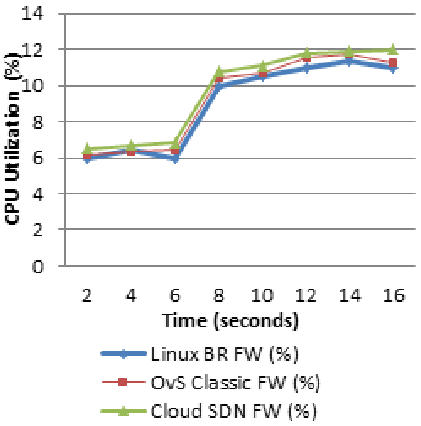

Figure 5. CPU utilization

Throughput: how much information is transmitted from source to destination in a given time frame will be referred to as network throughput. The number of packets that reaches the destination successfully is measured by throughput. Mostly the capacity of throughput is measured in bits per second, and it will also calculate as data per second. By altering the number of nodes/clients, and external client flooding one server, a more sustained TCP throughput is observed with OvS-based FW than with the LB mechanism. This can ensure that OvS is optimum in the applications of OpenStack Cloud. The overall aggregated throughput to all TCP flows is getting nearer to the maximum available bandwidth in the network interface for the long run. Table 3 denotes the throughput of three methods: LBFW, native OvS FW module, and cloud SDN FW. In Figure 6, the clients transmit traffic to server 1, and the overall TCP throughput is approximately 9.24 Gbps. As the number of clients increases, full bandwidth is used by total aggregated flows. The throughput of cloud SDN FW is high compared to remaining methods such as LBFW and native OvS FW module.

Table 3. TCP throughput (GBPS)

Nodes	Linux BRFW(%)	OvS Classic FW (%)	Cloud SDN FW (%)
	6.75	7.74	8.87
5	6.78	7.81	8.89
10	6.82	7.85	8.92
15	6.84	7.89	8.98
20	6.87	7.92	9.14
25	6.89	7.95	9.24

Cloud traffic is controlled by this presented solution and intervenes when a DDoS attacker increments the number of packets for consuming resources of all victims. In this condition, SecCloudDD can stabilize the traffic to a normalized level by blocking the attacking sources and signaling it in all switches in real-time. As shown in Figure 7, these simulations demonstrated that anomaly detection and its blocking would be performed in 15 seconds whenever many packets cross 25,000/second. Figure 8 exhibits that the approach with the same parameters based on the distance estimation reacts in 35 seconds. The described cloud network SDN framework is essential if abnormal packet traffic increases to avoid DDoS attacks before reaching the cloud network.

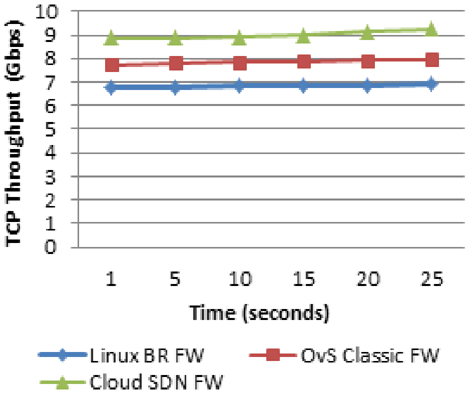

Figure 6. TCP throughput (GBPS)

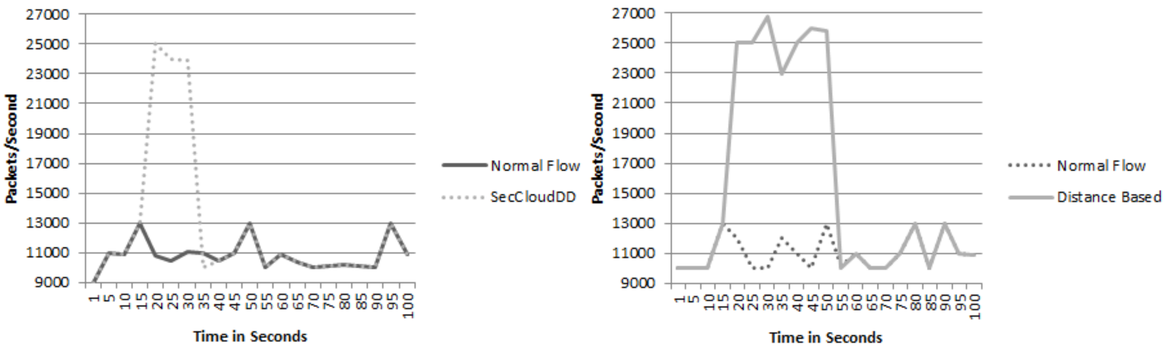

Figure 7. DDOS attack detection using SecCloudDD method

Figure 8. DDOS attack detection using distance estimation method

5. CONCLUSION

The new paradigm software network defines networking offers new opportunities for network security in cloud infrastructure. Due to the recent advances, SDN generates a unique opportunity for enabling complex scientific applications for running over tailored and dynamic architecture that includes network resources, computing, and storage. In this work, the cloud computing integrated view and SDN in different cases, particularly in the presence of DDoS/botnet attacks, network attacks are presented. Cloud network is presented based on SDN, which has threat analytics, multi-plane collaborative security monitoring notion, and attack mitigation/detection in emerging SDNFV-enabled CC larger-scale applications. In addition, the key extensions and plugins to the OpenStack/SDN-based cloud domain, specifically the network's infrastructure, are contributed to solving certain security problems in terms of security and reliability. The experimental results exhibited that it is robust while protecting CC infrastructure against these attacks. Hence presented approach helped to acquire the trust of CC users. The cloud-based on SDN is a platform agnostic, extensible for heterogeneous network schemes to any larger cloud applications like 5G, Industry 4.0, and the internet of things (IoT).

REFERENCES

[1] Y. Fazea and F. Mohammed, “Software defined networking based information centric networking: An overview of approaches and challenges,” in *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, Jul. 2021, pp. 1–8, doi: 10.1109/ICOTEN52080.2021.9493541.

[2] J. Singh, A. Refaey, and J. Koilpillai, “Adoption of the software-defined perimeter (SDP) architecture for infrastructure as a service,” *Canadian Journal of Electrical and Computer Engineering*, vol. 43, no. 4, pp. 357–363, 2020, doi: 10.1109/CJECE.2020.3005316.

[3] A. Kelkawi, A. Mohammed, and A. Alyatama, “Incremental deployment of hybrid IP/SDN network with optimized traffic engineering,” in *2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN),* Nov. 2020, pp. 57–63, doi: 10.1109/NFV-SDN50289.2020.9289859.

[4] M. Jalalitabar, Y. Wang, and X. Cao, “Branching-aware service function placement and routing in network function virtualization,” in *2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN),* Nov. 2019, pp. 1–6, doi: 10.1109/NFV-SDN47374.2019.9039981.

[5] W. Jia, Y. Liu, Y. Liu, and J. Wang, “Detection mechanism against DDoS attacks based on convolutional neural network in SINET,” in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Jun. 2020, pp. 1144–1148, doi: 10.1109/ITNEC48623.2020.9084918.

[6] O. Panchenko, A. Polishuk, M. Seliuchenko, and M. Beshley, “Method for adaptive client oriented management of quality of service in integrated SDN/CLOUD networks,” in *2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T),* Oct. 2017, pp. 452–455, doi: 10.1109/INFOCOMMST.2017.8246437.

[7] S. R. Basnet, R. S. Chaulagain, S. Pandey, and S. Shakya, “Distributed high performance computing in OpenStack cloud over SDN infrastructure,” in *2017 IEEE International Conference on Smart Cloud (SmartCloud),* Nov. 2017, pp. 144–148, doi: 10.1109/SmartCloud.2017.29.

[8] U. Ghosh, P. Chatterjee, D. Tosh, S. Shetty, K. Xiong, and C. Kamhoua, “An SDN based framework for guaranteeing security and performance in information-centric cloud networks,” in *2017 IEEE 10th International Conference on Cloud Computing (CLOUD),* Jun. 2017, pp. 749–752, doi: 10.1109/CLOUD.2017.106.

[9] S. Farahmandian and D. B. Hoang, “SDS 2: A novel software-defined security service for protecting cloud computing infrastructure,” in *2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)*, Oct. 2017, pp. 1–8, doi: 10.1109/NCA.2017.8171388.

[10] S. Pisharody, A. Chowdhary, and D. Huang, “Security policy checking in distributed SDN based clouds,” in *2016 IEEE Conference on Communications and Network Security (CNS)*, Oct. 2016, pp. 19–27, doi: 10.1109/CNS.2016.7860466.

[11] S. Seeber and G. D. Rodosek, “Improving network security through SDN in cloud scenarios,” in *10th International Conference on Network and Service Management (CNSM) and Workshop*, Nov. 2014, pp. 376–381, doi: 10.1109/CNSM.2014.7014198.

[12] S. P. Praveen, S. Sindhura, A. Madhuri, and D. A. Karras, “A novel effective framework for medical images secure storage using advanced cipher text algorithm in cloud computing,” in *2021 IEEE International Conference on Imaging Systems and Techniques (IST),* Aug. 2021, pp. 1–4, doi: 10.1109/IST50367.2021.9651475.

[13] H. H. C. Nguyen, V. S. Le, and T. T. Nguyen, “Algorithmic approach to deadlock detection for resource allocation in heterogeneous platforms,” in *2014 International Conference on Smart Computing*, 2014, pp. 97–103, doi: 10.1109/SMARTCOMP.2014.7043845.

[14] A. Taha, R. Trapero, J. Luna, and N. Suri, “AHP-based quantitative approach for assessing and comparing cloud security,” in *2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, Sep. 2014, pp. 284–291, doi: 10.1109/TrustCom.2014.39.

[15] P. Praveen S. and K. T. Rao, “Client-awareness resource allotment and job scheduling in heterogeneous cloud by using social group optimization,” *International Journal of Natural Computing Research*, vol. 7, no. 1, pp. 15–31, Jan. 2018, doi: 10.4018/IJNCR.2018010102.

[16] H. H. C. Nguyen, H. V. Dang, N. M. N. Pham, V. S. Le, and T. T. Nguyen, “Deadlock detection for resource allocation in heterogeneous distributed platforms,” in *Recent Advances in Information and Communication Technology 2015*, 2015, pp. 285–295, doi: 10.1007/978-3-319-19024-2_29.

[17] S. P. Praveen and K. T. Rao, “An effective multi-faceted cost model for auto-scaling of servers in cloud,” in *Smart Intelligent Computing and Applications*, 2019, pp. 591–601, doi: 10.1007/978-981-13-1921-1_58.

[18] P. P. Ray and N. Kumar, “SDN/NFV architectures for edge-cloud oriented IoT: A systematic review,” *Computer Communications*, vol. 169, pp. 129–153, Mar. 2021, doi: 10.1016/j.comcom.2021.01.018.

[19] E. Ahvar, S. Ahvar, S. M. Raza, J. M. Sanchez Vilchez, and G. M. Lee, “Next generation of SDN in cloud-fog for 5G and beyond-enabled applications: Opportunities and challenges,” *Network*, vol. 1, no. 1, pp. 28–49, Jun. 2021, doi: 10.3390/network1010004.

[20] D. E. Sarmiento, A. Lebre, L. Nussbaum, and A. Chari, “Decentralized SDN control plane for a distributed cloud-edge infrastructure: A survey,” *IEEE Communications Surveys and Tutorials*, vol. 23, no. 1, pp. 256–281, 2021, doi: 10.1109/COMST.2021.3050297.

[21] Q. Waseem, S. S. Alshamrani, K. Nisar, W. I. S. Wan Din, and A. S. Alghamdi, “Future technology: software-defined network (SDN) forensic,” *Symmetry*, vol. 13, no. 5, Apr. 2021, doi: 10.3390/sym13050767.

[22] I. T. Aziz and I. H. Abdulqadder, “An overview on SDN and NFV security orchestration in cloud network environment,” *Cihan University-Erbil Scientific Journal*, vol. 5, no. 1, pp. 20–27, Jun. 2021, doi: 10.24086/cuesj.v5n1y2021.pp20-27.

[23] T. Hu *et al.*, “SEAPP: A secure application management framework based on REST API access control in SDN-enabled cloud environment,” *Journal of Parallel and Distributed Computing*, vol. 147, pp. 108–123, Jan. 2021, doi: 10.1016/j.jpdc.2020.09.006.

[24] S. Badotra *et al.*, “A DDoS vulnerability analysis system against distributed SDN controllers in a cloud computing environment,” *Electronics*, vol. 11, no. 19, Sep. 2022, doi: 10.3390/electronics11193120.

[25] Y. Xu, Y. Yu, H. Hong, and Z. Sun, “DDoS detection using a cloud-edge collaboration method based on entropy-measuring SOM and KD-tree in SDN,” *Security and Communication Networks*, vol. 2021, pp. 1–16, Apr. 2021, doi: 10.1155/2021/5594468.

[26] A. Lakhan, M. A. Mohammed, O. I. Obaid, C. Chakraborty, K. H. Abdulkareem, and S. Kadry, “Efficient deep-reinforcement learning aware resource allocation in SDN-enabled fog paradigm,” *Automated Software Engineering*, vol. 29, no. 1, May 2022, doi: 10.1007/s10515-021-00318-6.

[27] A. lakhan, M. A. Mohammed, D. A. Ibrahim, and K. H. Abdulkareem, “Bio-inspired robotics enabled schemes in blockchain-fog-cloud assisted IoMT environment,” *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 1, pp. 1–12, Jan. 2023, doi: 10.1016/j.jksuci.2021.11.009.

[28] A. Lakhan, M. A. Mohammed, S. Kadry, K. H. Abdulkareem, F. T. Al-Dhief, and C.-H. Hsu, “Federated learning enables intelligent reflecting surface in fog-cloud enabled cellular network,” *PeerJ Computer Science*, vol. 7, Nov. 2021, doi: 10.7717/peerj-cs.758.

[29] M. Kamal *et al.*, “Privacy-aware genetic algorithm based data security framework for distributed cloud storage,” *Microprocessors and Microsystems*, vol. 94, Oct. 2022, doi: 10.1016/j.micpro.2022.104673.

BIOGRAPHIES OF AUTHORS

Venkata Nagaraju Thatha Associate Professor, Department of Information Technology, MLR Institute of Technology, Hyderabad, Telangana State, India. He was awarded Ph.D, Computer Science and Engineering from JNTUK in the year 2022. He has published 9 research articles in reputed international journals and conferences, 7 patents. His research interests include machine learning, artificial intelligence, and natural language processing. He can be contacted at email: nagaraju.thatha@gmail.com.

Swapna Donepudi received her M.Tech degree in Computer Science and Engineering from JNTU Kakinada. She is currently pursuing a Ph.D. from Gitam Institute of Technology, Visakhapatanam, Andhra Pradesh. She has 15 Scopus Indexed Journals and Conferences. She can be contacted at email: dswapna@pvpsiddhartha.ac.in

Miriyala Aruna Safali is a Professor of CSE at the Dhanekula Institute of Engineering and Technology in Vijayawada. She has two patents, more than 22 papers published in National and International Journals, is a fellow member of APAS and I2OR, and has published a textbook on machine learning for All. Data science, machine learning, and IoT are area of specializations. Her current research interests include deep learning for scientific cognition, domain sensitive large-scale frameworks, the internet of things, and sentiment analysis. She can be contacted at email: arunasafali.m@gmail.com.

Surapaneni Phani Praveen received his Ph.D. degree from the Department of Computer Science at Bharathiar University, Coimbatore, India, in 2020. He is currently Associate professor in the Department of Computer Science and Engineering at PVPSIT, Vijayawada, AP, India. His research interest includes cloud computing, data mining, mobile computing, wireless networks, and blockchain. He can be contacted at email: sppraveen@pvpsiddhartha.ac.in.

Nguyen Trong Tung graduated from the University of Danang in 2010 with a master's degree in computer science. He has 20 years of experience working in education, with special interests in computer networking, cloud computing, and machine learning. He has published numerous papers on resource allocation in distributed systems. Now he is a lecturer at Dong-A University, Da Nang, Vietnam. He has participated in many cooperation programs in the field of training, international conferences. Currently, he is a Ph.D. student at Ho Chi Minh City University of Technology, Vietnam. He can be contacted at email: tungqn@donga.edu.vn.

Nguyen Ha Huy Cuong obtained his doctorate in Computer Science/Resource Allocation Cloud Computing in 2017 from the University of Danang. He has published over 50 research papers. His main research interests include the resource allocation, detection, prevention, and avoidance of cloud computing and distributed systems. He serves as a technical committee program member, track chair, session chair and reviewer of many international conferences and journals. He is a guest editor of "International Journal of Information Technology Project Management (IJITPM)" with Special Issue On: Recent Works on Management and Technological Advancement. Currently, he is working at Software Development Centre, The University of Danang. He can be contacted at email: nhhcuong@vku.udn.vn.