SDN Ecosystem Implementation

B. N. MOHAN KUMAR
Deartment of ECE
RRIT,Bangalore-560010,
INDIA

Abstract: With its increased flexibility, agility, and control over network resources, software-defined networking (SDN) has become a paradigm shift in the field of network architecture. The OpenFlow protocol, a core component of SDN that enables centralized and programmable network administration, is at the heart of the progress. The existing and future state of software-defined networks using OpenFlow is thoroughly examined. It examines the historical development of SDN, clarifies the core ideas behind OpenFlow, and evaluates the influence of all three on modern networking. In-depth examinations of current developments, market leaders, and enduring difficulties are presented in the paper's analysis of the condition of SDN today. It also takes a forward-looking look at the coming years, seeing new trends, chances, and potential challenges. The article also looks into the various ways that SDN is being used in different industries and highlights the numerous benefits that these applications offer. It provides helpful insights into securing SDN and OpenFlow environments while addressing a crucial security issue. With its insightful analysis of the constantly changing world of software-defined networks and OpenFlow technology, It is a essential tool for network specialists, researchers, and decision-makers.

Keywords: software-defined networking, OpenFlow, network automation, security, network architecture, centralized network management.

Received: April 16, 2024. Revised: November 9, 2024. Accepted: December 8, 2024. Published: February 24, 2025.

1. Introduction

In the area of computer networking, Software-Defined Networking (SDN) has become a disruptive force that is changing how networks are built, managed, and used. OpenFlow, a crucial technology that gives SDN programmability and centralized network management, is the driving force behind the paradigm shift. In order to provide insight on the emergence of software-defined networks and their significant consequences for the current networking landscape, It does a thorough examination of their current and future environments using OpenFlow. The ever-increasing needs for agility, scalability, and adaptability in today's digital world have greatly challenged the traditional, rigid networking architecture, which is built on distributed and static configurations. By separating the control plane from the data plane, software-defined networking (SDN) overcomes these difficulties by enabling network managers to define network behavior and policies independently of the physical infrastructure. For contemporary applications like cloud computing, the Internet of Things, and edge computing, the architectural shift offers unprecedented flexibility and dynamic management capabilities. A key SDN enabler is OpenFlow, which was first released in 2008 [3]. It outlines the protocol for communication between the network devices that forward data packets and the SDN controller, which manages and specifies network policies.

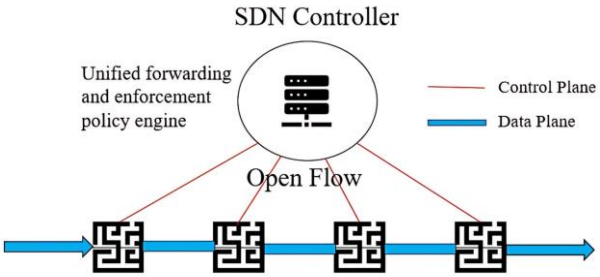

Fig.1: Representation of SDN using OpenFlow

The separation of the control and data planes, mediated by OpenFlow, enables dynamic network setup, in-flight adaptation to shifting traffic patterns, and the deployment of novel services without the need to upgrade or reconfigure network hardware [4]. Examining SDN and OpenFlow in their current condition and comprehending the trends, difficulties, and important industry actors is crucial as the networking ecosystem develops further. Examining the horizon, spotting new trends, and realizing the possibilities that SDN and OpenFlow hold are equally crucial. It is also examining the various uses of SDN across industries, emphasizing the practical advantages they bring. As networks grow more dynamic and vulnerable to cyber threats,[1] security considerations are also taken into consideration. Understanding the dynamic environment of software-defined networks utilizing OpenFlow is of utmost relevance in an era where digital connectivity is at the center of business and everyday life. For network experts, researchers, and decision-makers, It intends to be a thorough

resource that offers insightful information about the ground-breaking technology and its developing future.

2. Background Study

OpenFlow is a widely used protocol for interacting with the forwarding plane of network switches and routers. By allowing a centralized controller to direct how packets are forwarded,[2] it enables network behavior to be controlled at a higher degree of abstraction [6]. An OpenFlow-based SDN typically has three layers: application, control, and infrastructure. Switches and routers are examples of network hardware from the infrastructure layer that forwards packets in response to instructions from the control layer. One element of the control layer is the SDN controller, which oversees network devices and carries out network regulations. Controller-based applications that interact with network components to deliver network services to users are the last tier of the application layer [2]. Figure 1 depicts the SDN's architecture.

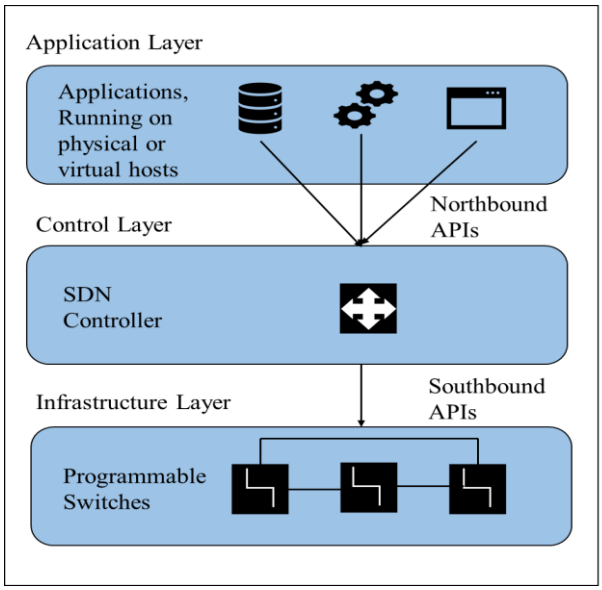

Fig.2: OpenFlow of SDN Architecture

Let's explore the various OpenFlow-based SDN architecture layers in further detail:

1) *Infrastructure Layer:* The physical network elements, such as switches, routers, and access points, are included in the infrastructure layer, which is the lowest layer of a network. According to their routing setups and tables, these components forward packets. In an OpenFlow-based SDN, these devices are coupled with a switch or OpenFlow agent, which communicates with the SDN controller to get packet routing instructions [6].

2) *Control Layer:* The control layer, which is the second layer, plays a crucial role in controlling the network and enforcing network policies. The layer includes the SDN controller, a software-based central entity that communicates with network devices via the OpenFlow protocol. Its primary responsibility is to collect and analyze network status information so that choices may be made regarding the optimal way to configure network devices to adhere to desired network policies. A more effective and efficient administration of network resources is ultimately made possible by the control layer, which serves as the network's primary management hub [7].

3) *Aapplication Layer:* The application layer is the topmost layer. It includes a number of apps that are intended to interact with network devices and run on the SDN controller in order to provide end users with network services.

The main software application for the network is the SDN controller, which functions either independently or as part of a network operating system. Its main job is to create a reliable network representation that various network services and applications can use. SDN applications like load balancing, network security, traffic engineering, and network monitoring employ network programmability to automate procedures and increase overall network effectiveness.

TABLE I: KEY CONCEPTS OF OPENFLOW PROTOCOL

Concept	Description
Control Plane	Separation from Data Plane
Flow Tables	Match-action processing
Flow Table Entry	Match fields, instructions, and counters
Controller	Centralized network control
Southbound APIs	Communication between controller and switches
Northbound APIs	Interfaces for applications and services

The OpenFlow-based (SDN) architecture enables network operators to segregate the control plane from the data plane [9]. The division makes it possible to create network topologies that are more versatile and flexible. It also makes network management simpler by centralizing network control and offering a consistent view of a network that can be used by many network applications and services [10].

3. Present Landscape of Software-defined Networks Using Openflow

3.1 Current Trends and Technologies in SDN:

1) *Network Virtualization:* Network virtualization is one of the main trends in SDN. SDN facilitates the establishment of virtual networks by separating network functions from actual hardware, improving scalability and resource efficiency. Technologies like Cisco's Application Centric Infrastructure (ACI) and VMware NSX are becoming more popular.
2) *Intent-Based Networking (IBN):* More and more networks are utilizing intent-based networking, which enables network administrators to specify the desired network behavior and let the network automatically arrange itself to fulfill those goals. One of the best IBN solutions is Cisco's DNA Center.

3) *SD-WAN Adoption:* A fast-expanding market segment is software-defined wide area networking (SD-WAN). SD-WAN solutions make use of SDN concepts to enhance WAN management and performance, especially in multi-site businesses. Leaders in the field include businesses like Cisco, VMware, and Silver Peak.
4) *Containerization and Microservices:* The development of containerization and microservices designs is having an impact on SDN. The dynamic and distributed nature of containers is changing the requirements for networks, thanks to technologies like Kubernetes and Docker. To deal with these issues, SDN solutions like Calico and Cilium are emerging.
5) *5G Integration:* The development of 5G networks is driving the demand for networking solutions that are more flexible and agile. SDN is essential to the integration of 5G technologies because it makes it possible to slice the network and allocate resources effectively.

3.2 Profiles of Major Players

TABLE II: KEY PLAYERS IN THE SDN INDUSTRY

Company Name	Description	Notable SDN Solutions
Cisco Systems	Global networking technology giant	Cisco ACI, Cisco SD-WAN
VMware	Leading virtualization and cloud computing firm	VMware NSX
Juniper Networks	Provider of high-performance networking	Juniper Contrail
Arista Networks	Specializes in cloud networking solutions	Arista EOS
Hewlett Packard Enterprise (HPE)	Technology solutions provider	HPE SDN Controller
Extreme Networks	Provides software-driven networking solutions	ExtremeXOS
Huawei	Global information and communications technology provider	Huawei CloudFabric, Huawei Agile Controller
Cumulus Networks	Open networking provider	Cumulus Linux

3.3 Existing Challenges

1) *Interoperability:* It is still difficult to achieve interoperability between various SDN technologies and conventional network architectures. The problem is being addressed by industry standards and open-source initiatives.
2) *Scalability:* Scalable and high-performance controllers and switches are required as SDN networks expand. In order to address scalability issues, hardware improvements and improved control plane software are being made.
3) *Security*: SDN creates additional security issues, such as controller weaknesses and expanded attack surfaces. Identity and access management is one emerging solution for SDN security; however, it still needs to be developed.
4) *Operational Complexity:* Operationally, moving to SDN can be difficult. Organizations frequently need to retrain employees and modify current workflows and procedures. For adoption to be effective, the obstacle must be cleared.

The current environment of OpenFlow-based software-defined networks is characterized by shifting trends, the involvement of significant industry actors, and persistent difficulties. As SDN develops, it offers solutions for 5G integration, SD-WAN, intent-based networking, network virtualization, and SD-WAN. The widespread acceptance and future success of SDN and OpenFlow, however, depend on addressing interoperability, scalability, security, and operational challenges.

4. Future Landscape of Software-Defined Networks Using OpenFlow

Technology breakthroughs and the ever-increasing demands of contemporary network settings are what are driving the dynamic and ongoing landscape of software-defined networks (SDN). The future of networking will be largely shaped by OpenFlow, which is a key protocol under SDN. The section explores the newest advances and prospective trends in SDN using OpenFlow, along with the difficulties and possibilities they bring.

4.1 Emerging Trends:

1) *5G Integration:* The need for more adaptable and agile network management is growing along with the deployment of 5G networks. SDN and OpenFlow will make 5G network orchestration more effective and enable operators to dynamically distribute resources to meet the various demands of apps and services.
2) *Multi-Cloud Environments:* Multi-cloud methods are being adopted by businesses more frequently. Network provisioning will be made easier and application performance will be improved thanks to SDN with OpenFlow, which enables seamless network management across numerous cloud providers and data centers.
3) *Intent-Based Networking (IBN):* A developing concept called "intent-based networking" tries to improve the clarity of network settings. To automate network operations and minimize human interaction, OpenFlow can be used to translate high-level network regulations into particular network settings.
4) *AI and Machine Learning Integration:* Predictive network optimization and maintenance will result from the combination of AI and machine learning with SDN and OpenFlow. Networks will anticipate traffic patterns, adjust to changing conditions, and proactively counter security threats.

4.2 Potential Developments:

1) *OpenFlow Enhancements:* The OpenFlow protocol is probably going to become more versatile and adaptable as it develops. Future iterations might include stronger security features, better support for advanced features, and more effective network flow management techniques.
2) *Convergence with Network Functions Virtualization (NFV):* The convergence of SDN and Network Functions Virtualization (NFV) will result in a more comprehensive approach to network management. Increased flexibility and resource efficiency will emerge from the integration's ability to dynamically deploy virtualized network functionalities through OpenFlow.
3) *Standardization and Interoperability:* SDN and OpenFlow protocol standardization efforts will probably carry on. Increased interoperability across various suppliers and SDN controllers may be seen in the landscape of the future, encouraging vendor-neutral, open solutions.
4) *Edge Computing Integration:* SDN and OpenFlow will be crucial in controlling the intricate and scattered networks at the network's edge as edge computing becomes more widespread. For edge devices, the integration will guarantee a low-latency, high-performance, and secure connection.

4.3 Anticipated Challenges:

TABLE III: CHALLENGES IN FUTURE LANDSCAPE OF SDN NETWORK USING OPENFLOW

Challenges	Description
Scalability	Ensuring that OpenFlow and SDN solutions scale effectively to meet the needs of expansive and complicated networks
Interoperability	overcoming compatibility problems across diverse switches, SDN controllers, and other network components from various vendors.
Security And Privacy	Addressing the ever-evolving vulnerabilities and attacks specific to SDN, as well as privacy issues and security threats in SDN systems
Regulatory Compliance	Complying with evolving regulations and standards, which may vary by region and impact SDN implementations.
Operational Complexity	Managing SDN environments' growing complexity, which can call for specialist knowledge and powerful management tools.

Addressing these issues will be essential to ensuring the successful adoption and use of SDN and OpenFlow because they are anticipated to influence their future landscape.

The OpenFlow-based SDN landscape of the future is both exciting and difficult. SDN and OpenFlow will play a bigger part in enabling agility, efficiency, and scalability as networks become more sophisticated and dynamic. However, to fully utilize these technologies while resolving the difficulties they provide, concentrated efforts in research, development, and teaching are required. Networking is expected to undergo a major shift in the upcoming years, and SDN with OpenFlow will surely be at the forefront of change.

5. Security Implications of Software-Defined Networks Using OpenFlow:

In the world of Software-Defined Networks (SDN), security is of utmost importance, especially when OpenFlow is used to govern and regulate network resources. New security issues arise as the traditional network paradigm transitions to a more dynamic and programmable architecture. The section is emphasizes special security issues and offers information on securing SDN and OpenFlow setups.

5.1 Centralized Control:

SDN's dynamic nature has important security implications. Security measures in a conventional network frequently rely on static configurations. The programmability of SDN, on the other hand, enables quick and in-the-moment changes to network policy and routing. The flexibility can be both a strength and a weakness. To avoid vulnerabilities or incorrect configurations, security experts must rapidly and effectively respond to these changes. SDN's centralized network control can have both positive and negative effects. While it gives the network complete visibility and control, it also turns into a single point of failure. Attackers who gain access to the SDN controller may be able to control the entire network. To safeguard the controller and its communications, strong security measures are essential, such as access controls, authentication, and encryption.

5.2 Southbound and Northbound Interfaces:

The southbound interface, which connects network devices to the SDN controller, and the northbound interface, which connects applications to the controller, make up the two main interfaces of SDN networks. Both interfaces require security precautions. In order to stop illegal access or device manipulation, the southbound interface needs to be protected. Contrarily, the northbound interface must make sure that only authorized applications communicate with the controller.

5.3 OpenFlow Protocol Vulnerabilities:

The OpenFlow protocol, which is the foundation of SDN, presents a unique set of security issues. Some of these are as follows:

1) *Denial of Service (DoS) Attacks:* Attackers may flood the controller with nefarious requests, causing the network to go down. Effective filtering and rate limitation are crucial defenses.
2) *Man-in-the-Middle (MitM) Attacks:* Interception or manipulation of OpenFlow messages between the controller and switches may result in unauthorized control of network resources. OpenFlow communication encryption reduces danger.
3) *Flow Table Poisoning:* Data leakage or network instability may result from unauthorized additions or updates to flow entries on network devices. Effective procedures for authentication and authorization are essential.

5.4 Flow Table Security and Network Segmentation:

Vulnerabilities may arise due to the dynamic nature of SDN flow tables. There is a chance that unauthorized flows will be added, clogging the network or enabling hostile activity. Flow table security measures must be put into place to ensure that only authorized parties can install flows and that they are regularly validated and cleaned away. The basic security principle of segmentation. Network segments must be adequately segregated in SDN to prevent lateral attacker movement. It can be accomplished via network access controls and micro-segmentation methods.

5.5 Security Monitoring and Intrusion Detection:

Systems for continuous monitoring and intrusion detection are crucial to SDN security. Malicious acts, irregular traffic patterns, and policy infractions must all be quickly discovered and stopped. The efficiency of these systems can be increased by using machine learning and AI-based techniques.

5.6 Disaster Recovery and Redundancy:

Solid disaster recovery strategies and network redundancy are crucial for ensuring the availability of SDN services in the event of security events or controller failures. Redundant controllers and failover techniques can reduce service interruptions brought on by security issues.

OpenFlow-dependent Software-Defined Network security requires a comprehensive approach. Understanding these security implications and putting proper procedures in place are essential to maintaining network integrity and safeguarding sensitive data as more businesses embrace SDN. Given the particular difficulties of programmable and centralized network control, security in SDN settings should be flexible, proactive, and all-encompassing.

6. Applications of OpenFlow-Based SDN

There are several uses for OpenFlow-based SDN, including network virtualization (NV), network slicing, network automation, and network security. Through network virtualization, several virtual networks may be created from a single physical network infrastructure, enhancing flexibility and scalability. By using network slicing, virtual networks may be built to accommodate multiple applications' and users' varying bandwidth, latency, and security demands. The complexity and cost of network operations are reduced thanks to network automation, which makes it possible to automate procedures like setup and provisioning. Network security enables the deployment of complex security policies and facilitates the detection and correction of security flaws. [11].

A. Traffic Engineering: By dynamically configuring the network devices in line with user demands and traffic patterns, OpenFlow-based SDN may be utilized to optimize network traffic. In order to enhance network performance and reduce congestion, the SDN controller may gather data on network traffic and make real-time routing decisions.

B. Quality of Service (QoS): Network management makes use of the idea of giving some forms of network traffic precedence over others [12]. Software-Defined Networking (SDN) based on OpenFlow may be used to implement QoS rules by setting network devices to provide precedence to particular types of traffic, such as audio or video traffic, over others. By ensuring that important traffic is delivered successfully, It improves both network performance and the user experience.

C. Cloud Computing: Cloud computing can also be made possible with OpenFlow-based SDN because of its more flexible and dynamic network infrastructure. SDN may also be used to create virtual networks for each cloud tenant, providing isolation and security across tenants [13–14].

D. Internet of Things (IoT): OpenFlow-based SDN may be used to facilitate IoT by creating a flexible and scalable network architecture that can manage the various devices and data created by IoT devices. IoT devices may benefit from security and privacy thanks to SDN.

E. Network Monitoring: SDN that is based on OpenFlow may be used to analyze problems and monitor network performance. The SDN controller may track network performance and offer network managers quick notifications when issues are discovered [13].

F. Network function virtualization (NFV): Network function virtualization (NFV) includes the virtualization of several network functions using software like firewalls and load balancers. By utilizing an OpenFlow-based Software Defined Network (SDN), NFV may provide a more adaptable and scalable network design that can dynamically distribute resources to different network functions based on user demands. It increases the flexibility and efficiency of managing network resources [11].

SDN, which is based on OpenFlow technology, offers a flexible and scalable network architecture that can be adapted to meet various user and application needs [14]. Due to its programmable and automated characteristics, network administration is simplified, performance optimization is improved, and complexity is decreased.

7. Challenges and Future Directions

OpenFlow-based SDN provides a lot of advantages, but there are still certain problems that need to be fixed. Interoperability, security, scalability, dependability, and performance are some of these challenges. Scalability is a major issue since SDN controllers may end up being a bottleneck when managing large networks. An issue with the SDN controller's dependability might interrupt the entire network. Performance is also a challenge, as SDN controllers may contribute more latency and overhead than traditional networking techniques. Security is still another major concern due to the possibility of new attack vectors that the network's central control may create. Interoperability is also important since it's possible that SDN deployments will lead to a lack of compatibility between the products of various providers. [15]. To address the issues that OpenFlow-based SDN confronts, future research must look at the use of AI and ML learning techniques. The effectiveness and security of networks are significantly impacted by these tactics [16]. Network traffic patterns may be predicted using machine learning techniques, which can also lead to better routing decisions. Artificial intelligence systems can be used to promptly recognize potential security risks [17] and take the necessary countermeasures. The use of distributed SDN systems may also be investigated by researchers as a way to improve scalability and reliability. Standardizing APIs and protocols can also improve interoperability across various SDN systems. It would make it straightforward for different SDN architectures to integrate and communicate with one another. By investigating these techniques, OpenFlow-based SDN's overall performance and security may be enhanced [16–18].

8. Conclusion

In conclusion, it has shown the dynamic environment of software-defined networks (SDN) and the crucial function of OpenFlow within it. SDN has developed from a promising idea to a revolutionary force in networking, providing unheard-of flexibility and control over network resources. The study has shown the present condition of SDN, highlighting ongoing developments, significant market participants, and the difficulties encountered in its deployment. Furthermore, by examining the probable future environment, we can discover new trends, possibilities, and difficulties that are expected to influence the development of SDN and OpenFlow. Applications of SDN in many sectors have been highlighted, demonstrating the versatility and advantages of the technology in numerous situations. Additionally, the study addressed the crucial problem of security in SDN and offered suggestions for securing these adaptable and programmable network environments. It is becoming more and more obvious that SDN has a significant influence. It offers a thorough overview of SDN and OpenFlow, making it an invaluable resource for network experts, researchers, and decision-makers. The aim is to give a road map for those navigating and transforming environment as they go from the present to the future of SDN, which is one of promise, innovation, and adaptability.

References

[1] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, ... and J. Turner, "OpenFlow: enabling innovation in campus networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, 2008, pp. 69-74.

[2] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," Proceedings of the IEEE, vol. 103, no. 1, 2015, pp. 14-76.

[3] Y. J. Kim, H. J. Kim, Y. H. Lee, and K. H. Kim, "Software-defined networking (SDN): A reference architecture and open APIs," Journal of Communications and Networks, vol. 15, no. 6, 2013, pp. 593-604.

[4] C. Y. Hong, and K. Kim, "SDN-based virtualization for future internet architecture," Journal of Communications and Networks, vol. 15, no. 6, 2013, pp. 580-592.

[5] M. A. Nascimento, L. C. Mendes, and F. L. D. Souza, "Machine learning applied to software-defined networks: a survey," Journal of Network and Computer Applications, vol. 137, 2019, pp. 1-22.

[6] Y. Zhang, Y. Chen, H. Xiong, and Y. Xiang, "Machine learning for network anomaly detection: A survey," IEEE Communications Surveys & Tutorials, vol. 20, no. 2, 2018, pp. 1045-1061.

[7] J. Yi, S. Li, and Z. Li, "A survey on artificial intelligence for network security," Journal of Network and Computer Applications, vol. 167, 2020, pp. 102739.

[8] P. Costa, P. Silva, and S. Fernandes, "Toward an open standard for SDN: OpenFlow, OVSDB, and YANG," IEEE Communications Magazine, vol. 58, no. 7, 2020, pp. 90-97.

[9] S. S. Kozat, and J. Shin, "Toward interoperable software-defined networking (SDN): A survey," IEEE Communications Magazine, vol. 53, no. 2, 2015, pp. 109-115.

[10] Q. Zhou, and X. Mao, "Distributed software-defined networking: A survey," IEEE Communications Surveys & Tutorials, vol. 19, no. 4, 2017, pp. 2294-2324.

[11] H. Kim, N. Feamster, & J. Rexford, "Improving network management with software defined networking," IEEE Communications Magazine, vol. 51, no. 2, 2013, pp. 114-119.

[12] S. Jain, and S. Paul, "Network virtualization and software defined networking for cloud computing: A survey," IEEE Communications Magazine, vol. 51, no. 11, 2013, pp. 24-31.

[13] G. Wang, Y. Liu, and Y. Yang, "A survey on software-defined networking," IEEE Communications Surveys & Tutorials, vol. 17, no. 1, 2015, pp. 27-51.

[14] R. Wu, S. Hariri, and I. L. Yen, "SDN-based network virtualization: A comprehensive survey," IEEE Communications Surveys & Tutorials, vol. 17, no. 1, 2015, pp. 485-512.

[15] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," Proceedings of the IEEE, vol. 103, no. 1, 2015, pp. 14-76.

[16] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, ... and S. Shenker, "OpenFlow: enabling innovation in campus networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, 2008, pp. 69-74.

[17] M. Casado, M. J. Freedman, J. Pettit, J. Luo, and N. McKeown, "Fabric: A retrospective on evolving SDN," ACM SIGCOMM Computer Communication Review, vol. 42, no. 4, 2012, pp. 7-13.

[18] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat, "Hedera: Dynamic flow scheduling for data center networks," Proceedings of the 7th USENIX conference on Networked systems design and implementation, 2010, pp. 19-19.

[19] M. Moshref, M. Yu, M. E. Moghaddam, and A. Gupta, "Towards achieving operational efficiency in software-defined data centers," Proceedings of the 15th ACM Workshop on Hot Topics in Networks, 2016, pp. 38-44.

[20] H. Deng, W. Cai, S. Guo, and J. Luo, "A survey of research on security in software-defined networks," IEEE Communications Surveys & Tutorials, vol. 19, no. 1, 2017, pp. 114-137.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The author contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The author has no conflict of interest to declare that is relevant to the content of this article.