Asian Journal of Research in Computer Science

9(2): 1-18, 2021; Article no.AJRCOS.68725
ISSN: 2581-8260

Comparison of Software Defined Networking with Traditional Networking

Saad H. Haji[1*], Subhi R. M. Zeebaree[1], Rezgar Hasan Saeed[2], Siddeeq Y. Ameen[1], Hanan M. Shukur[3], Naaman Omar[1], Mohammed A. M.Sadeeq[1], Zainab Salih Ageed[4], Ibrahim Mahmood Ibrahim[1] and Hajar Maseeh Yasin[1]

[1]Duhok Polytechnic University, Duhok, Kurdistan Region, Iraq.
[2]Near East University, Cyprus.
[3]Al-Kitab University, Kirkuk, Iraq.
[4]Nawroz University, Duhok, Kurdistan Region, Iraq.

Authors' contributions

This work was carried out in collaboration among all authors. All authors read and approved the final manuscript.

Article Information

DOI: 10.9734/AJRCOS/2021/v9i230216
Editor(s):
(1) Dr. Xiao-Guang Lyu, Huaihai Institute of Technology, P. R. China.
Reviewers:
(1) Fatima Faydhe Al-Azzawi, Middle Technical University MTU, Iraq.
(2) Rositsa Velichkova, Technical University of Sofia, Bulgaria.
(3) Muhammad Akmal Bin, University Tun Husein Onn Malaysia (UTHM), Malaysia.
Complete Peer review History: http://www.sdiarticle4.com/review-history/68725

Review Article

Received 20 March 2021
Accepted 24 May 2021
Published 27 May 2021

ABSTRACT

The Internet has caused the advent of a digital society; wherein almost everything is connected and available from any place. Thus, regardless of their extensive adoption, traditional IP networks are yet complicated and arduous to operate. Therefore, there is difficulty in configuring the network in line with the predefined procedures and responding to the load modifications and faults through network reconfiguring. The current networks are likewise vertically incorporated to make matters far more complicated: the control and data planes are bundled collectively. Software-Defined Networking (SDN) is an emerging concept which aims to change this situation by breaking vertical incorporation, promoting the logical centralization of the network control, separating the network control logic from the basic switches and routers, and enabling the network programming. The segregation of concerns identified between the policies concept of network, their implementation in hardware switching and data forwarding is essential to the flexibility required: SDN makes it less

**Corresponding author: E-mail: saad.hikmat91@gmail.com;*

complicated and facilitates to make and introduce new concepts in networking through breaking the issue of the network control into tractable parts, simplifies the network management and facilitate the development of the network. In this paper, the SDN is reviewed; it introduces SDN, explaining its core concepts, how it varies from traditional networking, and its architecture principles. Furthermore, we presented the crucial advantages and challenges of SDN, focusing on scalability, security, flexibility, and performance. Finally, a brief conclusion of SDN is revised.

Keywords: SDN; control plane; open flow; traditional networking.

1. INTRODUCTION

As networks are increasingly growing in size and requirements, navigating hardware switches has become a challenge. Setting up individual network software switches manually has been very complicated and time-consuming for businesses running highly virtual systems alongside large networks. This is where SDN comes into the game [1,2].

SDN can be described as a network approach that enables network operators to programmatically set up, track, change and control network operation through open interfaces such as the OpenFlow protocol [3]. The SDN transforms the operation, management, and configuration of the network infrastructures. The SDN's view is based on separating the data plane from the control plane [4]. SDN proposes to concentrate network intelligence on a single network component by distinguishing the data packet forwarding mechanism (data plane) from the routing process (control plane), as seen in Fig.1 [5-7].

The remaining parts of this paper describe SDN's comparison with traditional networking, the need for SDN, the architecture of SDN, the benefits of SDN, and explaining the tools used in SDN. Then the paper will be ended with a discussion and conclusion.

2. BACKGROUND THEORY

2.1 Traditional Networking vs. SDN

For the control plane, traditional networking implements a distributed paradigm. For each network device, protocols such as ARP, STP, OSPF, EIGRP, BGP, and others operate independently [8]. These network devices connect, but no centralized machine manages the whole network or summarizes [9,10]. The most critical difference between conventional networking and SDN is that traditional networking is hardware-based, whereas SDN is usually software-based [11,12]. SDN is more versatile since it is software-based, helping users better control and ease handling resources remotely in the control plane [13,14].

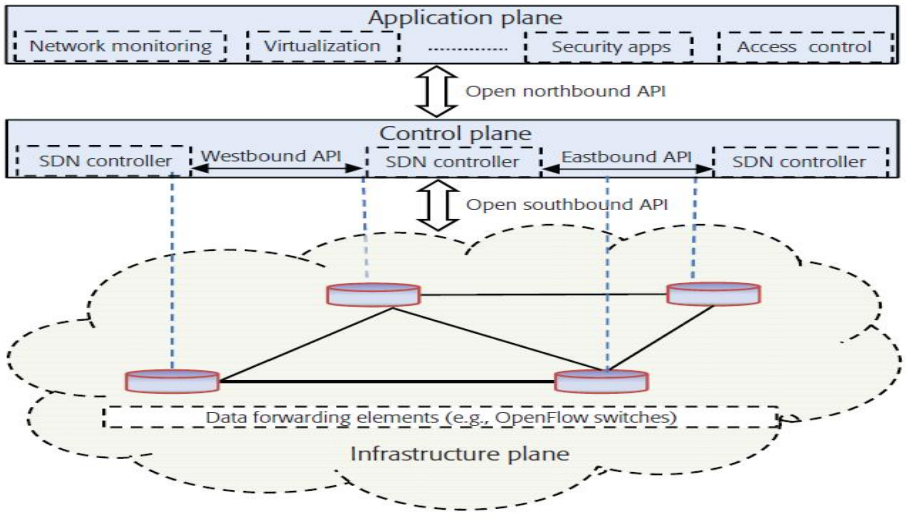

Fig. 1. SDN architecture [4]

Traditional networks utilize switches, routers, and other physical hardware to produce connections and operate the network [15-17]. A northbound interface that communicates with Application Programming Interfaces (APIs) is used in SDN controllers [18,19]. Because of this connectivity, device developers, as opposed to using the protocols required for conventional networking, can explicitly program the network [9,20]. Conventional networks are used to mount all data planes and control aircraft in one physical unit and then to share their capacity, increase the traffic load and the burden on the CPU and memory in two processes [21-23]. Detachments of control planes and data planes in SDN can be easily monitored and managed by the controller and network to take the right ride decisions and thus enable the network to better configure with a less traffic load, by separating these processes and having a dedicated server [9,24].

SDN is considered a popular alternative to traditional networking because it allows IT managers to provide extra physical infrastructure services and bandwidths without requiring an investment [25]. In order to expand the network power, traditional networking requires new hardware [26,27]. Fig. 2 shows the traditional network and SDN.

The main differences between the traditional networking architecture and SDN architecture as clarified in Table 1.

2.2 Need for SDN

SDN is defined as a modern paradigm that is rapidly becoming the alternative for networks that are unable to solve the shortages of traditional networking via isolating software from the hardware [28,29]. In SDN, management/control is provided for the hardware from a centralized software program. This software program is isolated from the hardware itself [30,31]. The prime focused need of SDN is an open source framework standard and layered architecture. Because software can be produced via different vendors easily, it is more effective, more flexible programmability, and more facilitating creativity in computer networking [7,32]. In SDN, several issues need to be addressed, such as scalability problems, virtualization, continuity of connectivity, location of controllers, and so on [33,34]. Reliability is one of the serious SDN difficulties. Reliability is an especially important issue for large-scale networks [22,35]. As the SDN controller tends to be a single point of failure, it is a technically unified control feature in the SDN. Accordingly, steps need to be taken to ensure that the reliability of modern technological solutions is at least as high as or better than before [36,37]. SDN is one of the most important innovations for developing the new economy's network infrastructure. However, unreliable networks cannot be the basis of the digital economy [38,39].

2.3 Architecture of SDN

SDN Architecture explains how SDN operates at its different stages and ensures the stability and reliability of software. For software-defined networking, there are primarily three layers: Application plane, Data plane, and Control plane [7,28,40]. SDN consists of 2 interfaces, one between the southbound APIs (e.g., OpenFlow) and the other between the API's application layer and the Northbound API's control layer. The SDN consists of 2 interfaces [41]. As shown in Fig.3.

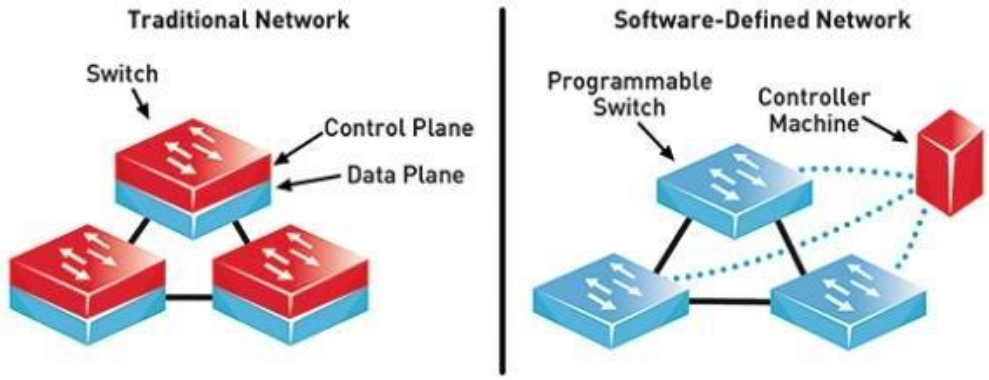

Fig. 2. The architecture of SDN Vs Traditional Network [9]

Table 1. The characteristics of SDN and traditional networking architecture

Characteristics	SDN	Traditional Network
Network Control Centralized	✓	×
Programmability	✓	×
Flexibility of Network	✓	×
Complex Control Network	×	✓
Performance Improved	✓	×
Configuration of Error-Prone	×	✓
Management Enhanced	✓	×
Configuration Efficiency	✓	×
Easy to use and implement	✓	×

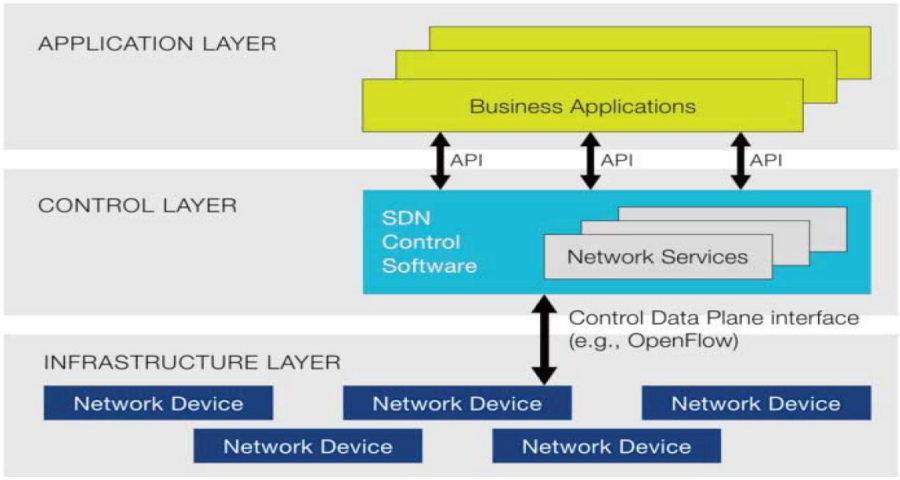

Fig. 3. Centralizing control plane [41]

2.3.1 Control plane

It can be defined as a control layer. It encompasses a series of software-based SDN controllers that provide a centralized control mechanism by a well-defined API to oversee network forwarding actions through an open interface [42,43]. Generally, The control plane consists of three primary layers: the device layer, the network operating system layer, and the network abstraction layer [44,45].

2.3.2 Southbound APIs

To connect with the SDN controller and network switches and routers, SDN southbound APIs are used. In this interface, the most common protocol is the OpenFlow protocol [10,41].

2.3.3 Application plane

The application layer consists of one or more programs, each of which has exclusive power over one or more SDN controllers exposed to a collection of resources—part of the SDN architecture, which consists of software implementing network services delivered to users/devices [46,47]. In order to achieve an abstract global view of the network they are using and to express the network activity they require at the moment, applications connect with the SDN controller by APIs (northbound interface) [44].

2.3.4 Northbound APIs

The relation between the applications and the SDN controller is the northbound APIs. The applications should inform the network what they need, and those services can be given by the network or convey what it has [41,48].

2.3.5 Infrastructure plane

The infrastructure plane is also known as the data layer or data plane [44]. Like the OSI model's physical layer, it comprises network components that interact with data traffic, such

as physical and virtual machines. It is an SDN forwarding plane and responsible for forwarding packet frames physically via the protocols used by the control plane from its entrance to the exit interface [7,49].

2.4 Benefits of SDN

One of SDN's key benefits is that it provides a platform for promoting more data-intensive software, for instance, virtualization and big data [50,51].

2.4.1 Centralized networking management

SDN can control the whole network from a centralized unit called a central node to automate network administration and security and ensure that security and policy knowledge is reliably communicated across the organization [7,52].

2.4.2 Reduced hardware costs

SDN uses the software principle to create a network with the minimal hardware available, removing the need for manual assistance and the expense of setup by leveraging the organizational performance and improving network usage by utilizing the virtualization concept [16, 26].

2.4.3 Cloud abstraction

Cloud computing is here to remain, and a unified infrastructure is emerging. It is easier to unify cloud services by abstracting cloud infrastructure using SDN. The networking elements that make up large data center systems can all be controlled [53,54].

2.4.4 Security approach

It gets easier to track and control the security features when there is a single management console for networking [54,55]. It may not have to deal with several applications around the system or dependent on them. It operates from one central point easily and provides a better security strategy [56]. When there is a security-related alarm, the same console may also be used to disperse information. In order to keep up with network management, virtualization has made it more complex for IT administrators [57]. Applying filtering rules and firewalls can be challenging for many virtual devices connected to the physical networks [58]. With SDN, it is possible to monitor and spread all information and safety measures consistently within the organization [59-61].

2.4.5 Automation

Today's network does not have to deal with internet access, unlike before. With SDN, it is also possible to adjust the cloud's automatic responses. In environments like enterprise-wide SD-WAN networks, the process works well [38].

2.5 Challenges of SDN

Even though SDN is identified as the basic solution to the problems that the infrastructure of the expanding network is facing major, it is still in its infancy phase. In addition to many others, advantages such as better functionality, lower cost, and higher efficiency have been laid out, but different challenges also demand attention. Challenges arise as SDN is generally accepted and new alternatives are being suggested [62].

2.5.1 Scalability

The main problems faced by SDN are scalability. From this single problem, two sub-issues can be extracted: (a) scalability of the controller (b) scalability of the network node. A single controller can handle up to 6 million flows per second [63]. Therefore, this demonstrates that for a large number of data forwarding nodes, only one controller or several controllers can manage control plane services needed [64,65]. To enhance scalability, rather than functioning on a peer-to-peer basis, the logically centralized controller should be physically distributed [66]. However, the problems faced by the controller when interaction happens will be shared between network nodes, whether it be a distributed or peer-to-peer controller network [38]. Hyper Flow and Onix are known as efficient means of achieving scalability. Through allocating and partitioning network status to separate physically dispersed controllers, Onix runs. HyperFlow is an application that allows for the interconnection of OpenFlow networks that are individually controlled [67]. Specifically, the events that allow changes to the network condition will be distributed by HyperFlow program, then all the distributed events will be replayed by the other controllers to reproduce the situation. As such, with the same homogeneous network topology, any controller will operate [64,68].

2.5.2 Flexibility and performance

How to deal with high-level packet processing flows proficiently is a fundamental problem of SDN. There are two main factors to be considered in this regard: flexibility and performance [69]. Flexibility refers to the ability of

networks to respond to modern and unprecedented functionality, such as software and facilities for the network. The performance deals with the speed at which information is transmitted from the control plane via network nodes in the data plane [70].

2.5.3 Security challenges

In software-defined networking, security is a very critical feature [71]. In order to provide usability, integrity, and protection to all elements and info, SDN protection needs to be integrated into the architecture [72]. You will have to secure and defend the device, rely on the SDN of each component, make sure the controller does what you want, and when a malfunction occurs, the architecture should be able to detect, fix and expose the problem [41]. The division of the data and control aircraft allows for security breaches and SDN safety issues. The optimal location of SDN controllers, switches, and other devices is an open challenge in SDN, which affects overall network security and performance [73,74]. Its integration is another security problem because of the design of SDN as it is flat, Where monitoring systems and defense solutions need to be compliant to improve overall performance, energy savings, and network security [5,75]. Fig.4 shows the potential SDN architecture attacks.

2.5.3.1 Data plane layer security challenge

The flood tables in the data plane lack space and flow tables' storage flow entries generate overhead on flow tables, leading to high cost and low performance [75,76]. Using intelligent flow table control techniques to store many low-cost and high-performance rules will overcome this problem [41]. Switches or access points can interrupt network activity, which results from malicious users initiating a Denial of Service (DoS) attack resulting in the interruption or network loss [70].

2.5.3.2 Control plane layer security challenge

Controllers are fundamental to SDN, but because of their centralized decision-making that can trigger networking in a security breach, it becomes a single weakness [77]. The control layer is an attractive function for security attacks due to its transparent environment. Another problem is how many switches to the controller are attached, and requests are sent to the controller, waiting for a response. If you add many switches to your controller's response time, your controller can crash due to the load on the controller [41,78].

2.5.3.3 Application plane layer security challenge

The hacker can flood malicious data into the application layer to monitor a network node that can infect other connected network nodes [79]. By inserting malicious code to monitor network packets' flow and steal valuable information, the attacker may obtain unauthorized access to the network node [1].

2.6 Implementation Tools for SDN

So many simulation tools have been developed to test SDN performance, such as OMNET++ and Mininet. Ns-3 and Estinet are the other modeling instruments. These methods have their capabilities. The comparison between the various simulation tools is seen in the Table.2 [9]. This paper presented a review of SDN, its definition, architecture, benefits, and challenges. We also reviewed the SDN networking paradigm design with the related open study challenges and revised some of the work performed with each challenge, including scalability, security, reliability, and performance. Moreover, several certain issues in SDN still require additional study attention to prevent inherited issues from the legacy networks, like standardizing the SDN modules and introducing new unique procedures developed for SDN.

3. LITERATURE REVIEW

Software-defined networks are a sophisticated network structure that detaches the network control plane from the forwarding plane (Data plane); SDN frees network devices from a range of detailed properties, responsibilities and provides a flexible model that can be managed through a global central controller. This idea is meant to enhance the infrastructure of integrated and programmable networks [80]. Due to the SDN mentioned above, different researchers concentrated on studying SDN.

Rahman, Islam, Montieri, Nasir, Reza, Band, Pescape, Hasan, Sookhak and Mosavi [81] presented a secure and optimized effective energy framework of Blockchain-enabled software-defined IoT for smart networks. In order to deploy a distributed efficient Blockchain-based SDN-IoT framework, they proposed a layered architecture that ensures secure network communication and efficient cluster-head

Table. 2. The Simulation Tools Comparison [9]

Feature \ Tool	OMNET ++	NS-3	Estinet	Mininet
Simulation Support	✓	✓	✓	×
Emulation Support	×	×	✓	✓
Capability to use an actual controller	×	×	✓	✓
Repeatable Outcomes	✓	✓	✓	×
Correctness of results outcome	No Real Controller	No Real Controller	✓	Performance relies on resources
Supporting GUI	Only for monitoring	only Monitoring, C++	Only for monitoring	only Monitoring, Python

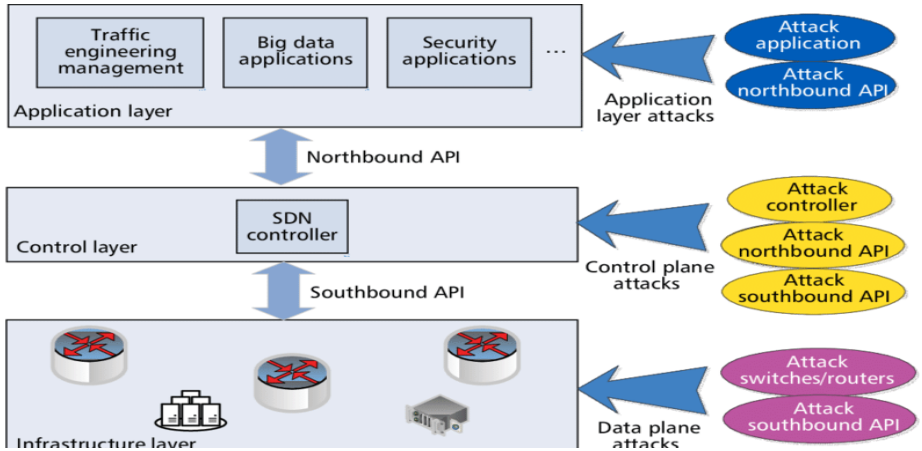

Fig. 4. Probable Attacks on SDN Architecture [1]

selection. Finally, they evaluated the performance of the suggested framework within a simulation environment. The results showed that it could obtain optimized end-to-end delay, energy-utilization, and throughput compared to classical Blockchain, i.e., capable of achieving security and efficiency in the smart network.

Vishnevsky, Pham, Kirichek, Elagin, Vladyko and Shestakov [82] discussed applying SDN to Unmanned Aerial System (UAS) to monitor sensor data, which provides abilities to manage sensor networks and UAV by a centralized SDN controller. They presented the SD-UAV architecture framework and sensor networking. They performed the comparison between the networks without SDN and networks using the SDN approach. The simulation findings showed that using the SDN approach within networks decreases packet loss and increases the bandwidth as transmitted datagrams were lost in SDN-assisted networks at 3.3%. In comparison, losses in SDN network datagrams were 16.6%. The result also showed that the proposed system could be scalable, flexible, and reusable for different applications.

Ruaro, Caimi and Moraes [83] Proposed Stable and Systematic SDN Architecture outlining necessary measures to support SDN in Many-Core Systems on Chip (MCSoC), wherein only trustworthy SDN Controllers identify the contact route. This work will help the MCSoCs designers incorporate stable SDN management for communications tools with the structural information presented. Furthermore, the proposed framework phases cover the functionality specifics from hardware modules to the OS and even analyze the impact on the user's role. Due to its co-design hardware/software, the techniques seen were low overheads and are viable in the MCSoC design sense. The experimental results demonstrated the capacity of the proposed architecture to prevent spoofing and DoS attacks with a low overhead SDN router system.

Ren, Bai, Wang and Li [84] proposed a formulation to minimize the maximum link utilization as the Traffic Engineering (TE) objective. They complied with TCAM (Ternary Content Addressable Memory) resource limitation and SDN waypoint enforcement. They solved the TE problem in a centralized manner by formulating it as an integer linear programming model. In order to solve the TE problem effectively, they developed a distributed algorithm derived from Lagrangian decomposition theory. The simulation findings explored that the proposed TE-aware distributed routing (TEDR) algorithm can obtain maximum link utilization when 30% of the SDN nodes are deployed comparable to full SDN. Also, it has a limited impact on routing efficiency.

Xu, Wang and Xu [85] explained that bringing several possible bottlenecks that attackers can leverage to reduce network efficiency or even interrupt the availability of networks. In addition, a more powerful and cost-effective saturation strike, a table miss attack, is examined. Their results were more than standard saturation. As protocol-independent security and efficient platform for SDN/OpenFlow networks, they proposed SDN Guardian to prevent missing table attacks. They also proposed SDN Guardian. It is located between the router and other controller deployments and protects the network using four functional modules: The packet sensitive fields preprocessor module, which causes the controller's flow rules to emit; the threat detector module to warn of the attack signal; a module of traffic filter which classifies the targeted ports; and frequency-based filtering of traffic; the law sweeper in the turn flow table for deleting malicious rules. All SDN Guardian designs comply with OpenFlow, requiring no alteration of the protocol or external equipment. The assessment showed that, in terms of control channel bandwidth, machine CPU usage, and transfer flow table with minimal device overhead, SDN Guardian could effectively ease the table-miss attack and protect network infrastructure resources.

Almohaimeed and Asaduzzaman [86] explained a new architecture for linking edge computing to software-based networking and showing improved performance in dealing with big data processing in SDN. The issue that has been reduced by SDN's creation of high pressure on the main controller affects the overall network output, leading to longer latency as the data size increases. They used a new model of SDN Edge Controlling that, by utilizing edge computing technologies, overcomes the limitations of the performance. In order to reduce the burden on the main SDN controller and decrease the delay between the control plane and forward plane, the goal is to get the computer and computing facilities close to the network equipment. The experimental results have shown that the main controller's overall response time is reduced by almost 62 percent per 10,000 requests, and bandwidth is reduced by almost 45 percent.

Al-Tam and Correia [87] presented a study about migrating devices from overloaded to underloaded controllers promotes network reliability and adaptability. However, at the same time, it is a difficult challenge to determine which switches can be transferred to which controllers while retaining a balanced load on the network. A local search algorithm which is Migration Competency-Based Load Balancing (MCBLB), is presented that takes a shift and swap movements into account and implements a managed solution shaking scheme. The results revealed that the proposed algorithm could raise the load balance by up to 14% relative to the latest work.

Y. I. Khalid, M. Ismael and Baheej Al-Khalil [5] showed that there are many security challenges in traditional networks, some of them ended by SDN and some others remain, like Address Resolution Protocol (ARP) spoofing. The author discussed the solution to prevent ARP spoofing without using any additional hardware and software but only by extending the SDN controller by a module, this scans each ARP packet in the network to identify and avoid potentially spoofed packets. The results of the simulation showed that the suggested mechanism is stable against the attack of ARP spoofing.

Lawal and Nuray [4] presented a real-time solution to detect and reduce Distributed Denial of Service (DDoS) attacks on the SDN network. DDoS aims to overwhelm the network traffic and stop the servers from being available all the time [88]. The Flow real-time analyzer added to the main controller, and the findings showed that the suggested approach detects and mitigates DDoS attacks effectively.

Achleitner, Bartolini, He, Porta and Tootaghaj [89] discussed that SDN provides a mechanism allowing the use of flow rules to modify and re-program the data plane easily. The realization of highly adaptive SDNs with the potential to respond to evolving requirements or recover in a short period after a network outage depends on successful flow rules updates. To support fast-changing flow specifications in SDNs, the optimization architecture and associated flow configuration algorithms have been developed, considering calculating the current flow configuration on the controller and the time of execution of this configuration on the switches. Via detailed simulations. The proposed algorithms have shown that they outperform existing, shortest path-based solutions in the considered scenarios by reducing the overall network initialization time by up to 55 percent while providing comparable packet loss. They also revealed that algorithms would reduce the average time to restore broken streams by 40 percent in a networked system with a fraction of a failed link.

Gao, Li, Xiao and Wei [90] discussed that attackers might initiate different attacks from data planes against SDN, such as attacks by DoS, topology attacks by poisoning, and side-channel attacks. Flow Keeper, a standard system for creating a stable data plane against multiple attacks, is proposed. Flow Keeper enforces the data plane's port control and lowers the control plane's workload by screening out unauthorized packets. Experimental studies indicated that Flow Keeper could be used to counteract various kinds of attacks effectively.

Chin, Xiong and Hu [91] clarified that a phishing attack is a very popular approach to manipulating an enterprise and end-users in social engineering. Nowadays, it has been one of the most dangerous threats. As a new approach to foil phishing threats, the author has suggested Phish Limiter. Can cope with network traffic dynamics to contain phishing threats and improve traffic management as SDN has a global networking view. The result showed that, with its accuracy of 98.39 percent, Phish Limiter is an efficient and effective solution for detecting and preventing phishing attacks.

Karakus and Durresi [26] Descript the unit costs for a service with QoS criteria is specified, and the unit cost for the service was characterized by CAPEX (capital expenses), OPEX (operating expenses), and the network workload for a certain duration. The operational costs are determined. The authors also studied the relation between the unit cost of service and the scalability of a network. Experiments showed that the unit cost of service and the scalability of an architectural control plane are interrelated: more compact architectures lead to the lower unit cost of service.

Dridi and Zhani [92] explained an application to protect the SDN network upon DoS attack. DoS attacks are a considerable threat to such networks where the communication and processing ability of the controller and flood switch CAM tables can be overwhelmed by DOS attacks quickly. Furthermore, this will reduce the

general performance of the network. To protect or solve this issue, the author proposed the SDN-Guard application by rerouting possible malicious traffic, changing timeouts for flow, and aggregating flow rules. The tests carried out revealed that the SDN-Guard could decrease the DoS effect by significantly decreasing the incoming output of the controller and the bandwidth of the control plane by up to 32 percent and reducing transfer memory by up to 26 percent.

Cui, Yu and Yan [53] clarified that SDN's positive features would significantly facilitate collecting, delivering, retrieving, and analyzing big data. Big data, on the other hand, would have significant implications on SDN architecture and operation. The authors showed that SDN could benefit from big data, including traffic modeling, cross-layer architecture, security threats defeat, and SDN-based intra-and inter-data center networks. With big data, a promising approach for networking will be big data and SDN joint architecture.

3.1 Survey Discussion and Analysis

Traditional networks are complicated and difficult to control. Most of the reasons for this are that data and control planes are vertically integrated and specific to the manufacturer. SDN provided an opportunity to resolve these long-standing issues by decoupling the Data plane and Control plane, making the network more flexible and centralized the control network. For this reason, many studies focused on SDN and its utilization instead of traditional networking. Based on the literature review, each research studied SDN because of different features. Table 3 shows a comparison among the researches mentioned in section 2. From the comparison table, it is obvious that reference [81] showed that it could To guarantees the security and consistency to the network using a secure and optimized effective energy framework of Blockchain-enabled software-defined IoT compared to classical Blockchain. The authors [82] showed that using the SDN approach within networks decreases packet loss and increases the bandwidth. The study [83] presented a secure and systemic SDN framework capable of avoiding spoofing attacks and DoS with a common SDN router configuration overhead. The reference [84] proposed TEDR algorithms that can achieve optimum connection use if the SDN nodes are deployed as 30 percent as complete SDN and have a small effect on routing efficiency. The research [85] explored that using SD Guardian can reduce table-miss special attacks. The authors [86] showed that edge-based control on the centralized SDN controller could significantly handle higher network load while maintaining lower latency. The reference [87] found that by using the MCBLB algorithm in the SDN network, the load balancing is increased by up to 14 %. The authors [5] discussed how to prevent ARP spoofing without using any additional hardware and software but only by extending the SDN controller by a module. The results of the simulation showed that the suggested mechanism is stable against the attack of ARP spoofing. The [4] clarified that using sFlow technology embedded in the controller shows that the method can detect and reduce DDoS attacks. The study [89] discussed that SDN provides a mechanism allowing the use of flow rules to modify and re-program the data plane easily. They developed optimization architecture and associated flow configuration algorithms that reduce the configuration time by 55% and average time to recover interrupted flows by 40 %. The [90] showed that under DoS attacks, Flow Keeper maintains more than 80 percent bandwidth and can prevent unauthorized topology changes by screening out forged LLDP packets. The authors [91] explored that Phish Limiter is an efficient and effective solution for detecting and preventing phishing attacks within SDN networks. The reference [26] demonstrated an inverse relation between the unit cost of the service and the control scalability of the architecture where more scalable architecture contributes to lower unit cost of service. The research [92] found that using SDN-Guard the DoS attacks on the performance of SDN controller decreased by up to 32%. The authors [53] showed that SDN can benefit from big data, where big data and SDN joint design will become a promising approach for networking big data.

In the last decade, IT has improved considerably. The growth of cloud, social networking and other developments such as the internet of things has made IT a server center. Therefore, the network should be considered a competitive tool for IT and corporate leaders. Any issue that affects the network would thus have a direct effect on the enterprise, which will cost the company money and/or resources. It is essential for companies to address today's top issues for the networks.

Table 3. Critical Analysis of Existing Studies

Ref.	Year	Achieved Objectives	Significant Results	Tool/Technique
[42]	2021	To guarantees security and consistency to the network.	The results showed that it could obtain optimized end-to-end delay, energy-utilization, and throughput compared to classical Blockchain, i.e., capable of achieving security and efficiency in the smart network.	MininetWi-Fi emulator.
[43]	2020	To provide abilities to manage sensor networks and UAV by a centralized SDN controller.	The simulation findings showed that using the SDN approach within networks decreases packet loss and increases the bandwidth.	Mininet Wi-Fi emulation platform.
[44]	2020	To present a secure and systemic SDN framework describing the required steps to support SDN in MCSoC.	The experimental findings showed the capability of the proposed framework to avoid spoofing attacks and DoS with a common SDN router configuration overhead.	RTL (VHDL and System C) , C code (maps-GCC cross-compiler)
[45]	2020	To minimize the maximum link utilization as the Traffic Engineering (TE) objective.	The simulation findings showed that the proposed TEDR algorithm could obtain maximum link utilization when 30% of the SDN nodes are deployed, comparable to full SDN. Also, it has a limited impact on routing efficiency.	TEDR algorithm.
[46]	2020	To reduce table-miss special attacks.	By using SDN Guardian, the table-miss special attack was mitigated.	Testbed
[47]	2019	To retrieve processing and storage resources near network devices using the edge control system, the burden on the centralized SDN controller can be minimized.	The results showed that higher network load cab is handled significantly by using edge-based control while maintaining latency lower.	Python.
[48]	2019	To solve a Switch Migration Problem for time computation, load balancing, and robustness in SDN network via applying MCBLB algorithm.	The results showed an increase in load balancing by up to 14 %.	MATLAB
[5]	2019	To prevent SDN Network against Address Resolution Protocol ARP spoofing Attack	The suggested mechanism proved its robustness against ARP attacks and is very easy to detect and avoid.	Mininet

Ref.	Year	Achieved Objectives	Significant Results	Tool/Technique
[3]	2018	To protect the SDN network against DDoS attacks.	The sFlow technology embedded in the controller showed that the method could detect and reduce DDoS attacks.	Mininet
[50]	2018	To reduce the setup time of the SDN network in response to the changing requirements.	They showed reducing in configuration time by 55% and average time to recover interrupted flows by 40 %	Python
[51]	2018	To build a stable data plane against multiple attacks.	Under DoS attacks, Flow Keeper maintains more than 80 percent bandwidth and can prevent unauthorized topology changes by screening out forged LLDP packets.	Mininet, Polaris switch, Python
[52]	2018	To protect the SDN network against a Phishing attack.	The Results shows that Phish Limiter offers an effective and effective approach with an accuracy of 98.39%	GENI
[12]	2017	Describe unit price assessment for a QoS-parameter service and define the unit cost of a service concerning CAPEX, OPEX, and network workload for a certain period.	Experiments showed that the unit cost of a service is linked in reverse to the scalability of the control plane: more scalable architectures lead to the lower unit service cost.	--------
[53]	2016	Mitigate DoS attacks and protect SDN network.	Using SDN- Guard the DoS attacks on the performance of SDN controller decreased by up to 32%.	Mininet
[24]	2016	Using the advantages of SDN to boost the efficiency of large data systems and how to use big data to make SDN operate quicker and more efficiently are urgent issues that need to be tackled.	Big data and SDN joint design will become a promising approach for networking big data.	Python.

Today, the network has the top five problems:

- Network issues troubleshooting: The challenge of wireless divided tunnel WAN connections has always been challenging but has been especially difficult.
- To ensure that the network is used appropriately. The main challenge is to ensure that the traffic on a network is business-related, in particular with WAN links. Consumer apps are becoming more smart and there is increasing video traffic in real time and on demand.
- Ensure main applications tools. Often network operations are a compromise game. Prioritize this service application. Deploy performance protection. Spend more budget than is required for the network to be over-supplied for peak use.
- Reduce expenditure on broad area network. Nearly all CIOs have been responsible for reducing the costs of IT management. Given the high price that it makes sense to look at the WAN, given MPLS and other private network networks.
- Vital IT projects support. For the very life of certain organisations, business resilience is crucial. Companies who have the potential to rapidly introduce new services will be frozen.

The following are number of the newest research in progress in the SDN field:

- SmartBlock-SDN: An Optimized Blockchain-SDN Framework for Resource Management in IoT, by Rahman et al., at [81].
- BDF-SDN: A Big Data Framework for DDoS Attack Detection in Large-Scale SDN-Based Cloud, by Dinh et al. [93].
- SmartBlock-SDN: An Optimized Blockchain-SDN Framework for Resource Management in IoT, by Haque et al. [94].
- Networks Modernization Using SDN and NFV Technologies, by Kundimana et al. [95].
- DSF: A Distributed SDN Control Plane Framework for the East/West Interface, by Almadani et al. [96].

4. CONCLUSION

SDN is an evolving networking paradigm that enables a standardized programming capability to control network behavior. Since SDN is a modern approach to networking, this architecture has been used to redesign various solutions to classical network problems, while several issues remain challenging. SDN provides efficient and automatic control of the network that meets the need for increased complexity of the network and many other software domains. This paper reviewed the SDN networking paradigm design with the related open study challenges and revised some of the work performed with each challenge, including scalability, security, reliability, and performance. Moreover, several certain issues in SDN still require additional study attention to prevent inherited issues from the legacy networks, like standardizing the SDN modules and introducing new unique procedures developed for SDN. To develop innovative ideas for controllers that are the brains of the SDN design, the study needs to concentrate more on the control plane. As the control plane is a point of failure for the entire network, several security measures should be considered. As a result, SDN plays a vital role in redesigning various solutions to classical network problems, while several issues remain challenging. It also provides efficient and automatic control of the network that meets the need for increased complexity of the network and many other software domains.

The question is "while we're building it, can you (customers) come up?" A unsuccessful attempt to invest in a new deal left a start-up cautious. We helped them to work out what the consumer actually needs to buy with Service Design.

The emphasis on customer travel actually dominates the service architecture, such that the increasingly diverse problems facing public institutions and industries are not enough in itself. The combination of structural architecture capability and an interdisciplinary approach is vital for tackling diverse problems in the public sector.

In the field of New York City, a good step passed towards helping to construct the New York Chapter of the SDN, along with other leaders of service architecture. In the years that followed, it been recognized that the related staff and chapter with honors for their chapter work and took part in SDN's global campaigns in diversity, equity, and inclusion as part of a 2020 taskforce. And they found time for the day's work: managing service architecture at Capital One, the US banking giant.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Lotlikar T, Shah D. A Defense Mechanism for DoS Attacks in SDN (Software Defined Network). In 2019 International Conference on Nascent Technologies in Engineering (ICNTE). 2019;1-7.
2. Karmakar KK, Varadharajan V, Tupakula U. Mitigating attacks in Software Defined Network (SDN). in 2017 Fourth International Conference on Software Defined Systems (SDS). 2017;112-117.
3. Zeebaree SR, Shukur HM, Hussan BK. Human resource management systems for enterprise organizations: A review. Periodicals of Engineering and Natural Sciences (PEN). 2019;7:660-669,.
4. Lawal BH, Nuray AT. Real-time detection and mitigation of distributed denial of service (DDoS) attacks in software defined networking (SDN). In 2018 26th Signal Processing and Communications Applications Conference (SIU). 2018;1-4.
5. Khalid HYI, Ismael PM, Baheej Al-Khalil A. Efficient mechanism for securing software defined network against arp spoofing attack. Journal of Duhok University. 2019;22:124-131.
6. Mohammed AH, Hussein KRMMk, Abdulateef IA. A review software defined networking for internet of things. In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). 2020;1-8.
7. Deepak Singh Rana SAD, Sushil Kumar Chamoli. Software defined networking (SDN) challenges, issues and Solution. International Journal of Computer Sciences and Engineering. 2019;7:884-889.
8. Zebari RR, Zeebaree S, Jacksi K, Shukur HM. E-business requirements for flexibility and implementation enterprise system: A review. International Journal of Scientific & Technology Research. 2019;8:655-660.
9. Prajapati A, Sakadasariya A, Patel J. Software defined network: Future of networking. In 2018 2nd International Conference on Inventive Systems and Control (ICISC). 2018;1351-1354.
10. Zeebaree S, Ameen S, Sadeeq M. Social media networks security threats, risks and recommendation: A case study in the kurdistan region. International Journal of Innovation, Creativity and Change. 2020;13:349-365.
11. Alzakholi O, Shukur H, Zebari R, Abas S, Sadeeq M. Comparison among cloud technologies and cloud performance. Journal of Applied Science and Technology Trends. 2020;1:40-47.
12. Zebari SR, Yaseen NO. Effects of parallel processing implementation on balanced load-division depending on distributed memory systems. J. Univ. Anbar Pure Sci. 2011;5:50-56.
13. Mousa M, Bahaa-Eldin AM, Sobh M. Software defined networking concepts and challenges. in 2016 11th International Conference on Computer Engineering & Systems (ICCES). 2016;79-90.
14. Xu H, Huang H, Chen S, Zhao G, Huang L. Achieving high scalability through hybrid switching in software-defined networking. IEEE/ACM Transactions on Networking. 2018;26:618-632.
15. Sufiev H, Haddad Y. A dynamic load balancing architecture for SDN. In 2016 IEEE International Conference on the Science of Electrical Engineering (ICSEE). 2016;1-3.
16. Kareem FQ, Zeebaree SR, Dino HI, Sadeeq MA, Rashid ZN, Hasan DA*, et al.* A survey of optical fiber communications: challenges and processing time influences. Asian Journal of Research in Computer Science. 2021;48-58.
17. Yazdeen AA, Zeebaree SR, Sadeeq MM, Kak SF, Ahmed OM, Zebari RR. FPGA implementations for data encryption and decryption via concurrent and parallel computation: A review. Qubahan Academic Journal. 2021;1:8-16.
18. Zeebaree S, Zebari I. Multilevel client/server peer-to-peer video broadcasting system. International Journal of Scientific & Engineering Research. 2014;5.
19. Zebari IM, Zeebaree SR, Yasin HM. Real Time Video Streaming From Multi-Source Using Client-Server for Video Distribution. In 2019 4th Scientific International Conference Najaf (SICN). 2019;109-114.
20. Zeebaree SR, Shukur HM, Haji LM, Zebari RR, Jacksi K, Abas SM. Characteristics and analysis of hadoop distributed systems. Technology Reports of Kansai University. 2020;62:1555-1564.
21. Shukur H, Zeebaree S, Zebari R, Ahmed O, Haji L, Abdulqader D. Cache coherence

protocols in distributed systems. Journal of Applied Science and Technology Trends. 2020;1:92-97.

22. Ageed ZS, Zeebaree SR, Sadeeq MA, Abdulrazzaq MB, Salim BW, Salih AA*, et al.* A state of art survey for intelligent energy monitoring systems. Asian Journal of Research in Computer Science. 2021;46-61.
23. Ageed ZS, Zeebaree SR, Sadeeq MM, Kak SF, Yahia HS, Mahmood MR*, et al.* Comprehensive survey of big data mining approaches in cloud systems. Qubahan Academic Journal. 2021;1:29-38.
24. Perepelkin D, Tsyganov I. SDN Cluster Constructor: Software Toolkit for Structures Segmentation of Software Defined Networks. In 2019 XVI International Symposium "Problems of Redundancy in Information and Control Systems (REDUNDANCY). 2019;195-198.
25. Abdulqadir HR, Zeebaree SR, Shukur HM, Sadeeq MM, Salim BW, Salih AA*, et al.* A study of moving from cloud computing to fog computing. Qubahan Academic Journal. 2021;1:60-70.
26. Karakus M, Durresi A. Service cost in software defined networking (SDN). in 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA). 2017;468-475.
27. Shukur H, Zeebaree SR, Ahmed AJ, Zebari RR, Ahmed O, Tahir BSA*, et al.* A state of art survey for concurrent computation and clustering of parallel computing for distributed systems. *Journal of Applied Science and Technology Trends.* 2020;1:148-154.
28. Mubarakali A, Alqahtani AS. A Survey: security threats and countermeasures in software defined networking. In 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT). 2019;180-185.
29. Abdullah PY, Zeebaree SR, Shukur HM, Jacksi K. HRM system using cloud computing for small and medium enterprises (SMEs). Technology Reports of Kansai University. 2020;62:04.
30. Dino HI, Zeebaree SR, Ahmad OM, Shukur HM, Zebari RR, Haji LM. Impact of load sharing on performance of distributed systems computations. International Journal of Multidisciplinary Research and Publications (IJMRAP). 2020;3:30-37.
31. Zeebaree SR, Rajab H. Design and implement a proposed multi-sources to multi-destinations broadcasting video-signals. in 2019 4th Scientific International Conference Najaf (SICN). 2019;103-108.
32. AlShehri SMaMAR. Software defined networking: research issues, challenges and opportunities. Indian Journal of Science and Technology. 2017;10:1-9.
33. Abdulraheem AS, Abdulla AI, Mohammed SM. Enterprise resource planning systems and challenges.
34. Ageed ZS, Zeebaree SR, Sadeeq MM, Kak SF, Rashid ZN, Salih AA*, et al.* A survey of data mining implementation in smart city applications. Qubahan Academic Journal. 2021;1:91-99.
35. Fonseca PC, Mota ES. A survey on fault management in software-defined networks. IEEE Communications Surveys & Tutorials. 2017;19:2284-2321.
36. Salih AA, Zeebaree SR, Abdulraheem AS, Zebari RR, Sadeeq MA, Ahmed OM. Evolution of mobile wireless communication to 5g revolution. Technology Reports of Kansai University. 2020;62:2139-2151.
37. Hassan RJ, Zeebaree SR, Ameen SY, Kak SF, Sadeeq MA, Ageed ZS*, et al.* State of art survey for iot effects on smart city technology: challenges, opportunities, and solutions. Asian Journal of Research in Computer Science. 2021;32-48.
38. Netes V, Kusakina M. Reliability Challenges in Software Defined Networking," presented at the Proceedings of the 24th conference of open innovations association fruct, Moscow, Russia; 2019.
39. Yahia HS, Zeebaree SR, Sadeeq MA, Salim NO, Kak SF, Adel AZ*, et al.* Comprehensive survey for cloud computing based nature-inspired algorithms optimization scheduling. Asian Journal of Research in Computer Science. 2021;1-16.
40. Rawat DB, Reddy SR. Software defined networking architecture, security and energy efficiency: A survey. IEEE Communications Surveys & Tutorials. 2017;19:325-346.
41. Elazim NMA, Sobh MA, Bahaa-Eldin AM. Software defined networking: attacks and countermeasures. in 2018 13th International Conference on Computer Engineering and Systems (ICCES). 2018;555-567.
42. Zhong W, Yu R, Xie S, Zhang Y, Tsang DHK. Software defined networking for flexible and green energy internet. IEEE

Communications Magazine. 2016;54:68-75.
43. Sallow AB, Sadeeq M, Zebari RR, Abdulrazzaq MB, Mahmood MR, Shukur HM, *et al.* An investigation for mobile malware behavioral and detection techniques based on android platform. IOSR Journal of Computer Engineering (IOSR-JCE). 22;14-20.
44. Huang H, Yin H, Min G, Jiang H, Zhang J, Wu Y. Data-driven information plane in software-defined networking. IEEE Communications Magazine. 2017;55:218-224.
45. Akhunzada A, Ahmed E, Gani A, Khan M, Imran M, Guizani S. Securing the Software Defined Networks: Taxonomy, Requirements, and Open Issues. IEEE Communications Magazine. 2014;53.
46. Hasan DA, Hussan BK, Zeebaree SR, Ahmed DM, Kareem OS, Sadeeq MA. The impact of test case generation methods on the software performance: A review. International Journal of Science and Business. 2021;5:33-44.
47. Sadeeq MA, Zeebaree S. Energy management for internet of things via distributed systems. Journal of Applied Science and Technology Trends. 2021;2:59-71.
48. Abdulraheem AS, Salih AA, Abdulla AI, Sadeeq MA, Salim NO, Abdullah H, *et al.* Home automation system based on IoT; 2020.
49. Abdulrahman LM, Zeebaree SR, Kak SF, Sadeeq MA, Adel AZ, Salim BW, *et al.* A state of art for smart gateways issues and modification. Asian Journal of Research in Computer Science. 2021;1-13.
50. Shin S, Xu L, Hong S, Gu G. Enhancing network security through software defined networking (SDN). in 2016 25th International Conference on Computer Communication and Networks (ICCCN). 2016;1-9.
51. Cox JH, Chung J, Donovan S, Ivey J, Clark RJ, Riley G, *et al.* Advancing software-defined networks: a survey. IEEE Access. 2017;5:25487-25526.
52. Ibrahim IM. Task scheduling algorithms in cloud computing: A review. Turkish Journal of Computer and Mathematics Education (TURCOMAT). 2021;12:1041-1053.
53. Cui L, Yu FR, Yan Q. When big data meets software-defined networking: SDN for big data and big data for SDN. IEEE Network. 2016;30:58-65.
54. Sulaiman MA, Sadeeq M, Abdulraheem AS, Abdulla AI. Analyzation Study for Gamification Examination Fields. Technol. Rep. Kansai Univ. 2020;62:2319-2328.
55. Swami R, Dave M, Ranga V. Software-defined networking-based DDoS defense mechanisms. ACM Comput. Surv.2019;52.
56. Jijo BT, Zeebaree SR, Zebari RR, Sadeeq MA, Sallow AB, Mohsin S, *et al.* A comprehensive survey of 5g mm-wave technology design challenges. Asian Journal of Research in Computer Science. 2021;1-20.
57. Sallow A, Zeebaree S, Zebari R, Mahmood M, Abdulrazzaq M, Sadeeq M. Vaccine tracker," SMS reminder system: Design and implementation; 2020.
58. Ageed ZS, Ibrahim RK, Sadeeq MA. Unified ontology implementation of cloud computing for distributed systems. Current Journal of Applied Science and Technology. 2020;82-97.
59. Dacier MC, König H, Cwalinski R, Kargl F, Dietrich S. Security challenges and opportunities of software-defined networking. IEEE Security & Privacy. 2017;15:96-100.
60. D'Cruze H, Wang P, Sbeit R, Ray A. A software-defined networking (sdn) approach to mitigating ddos attacks. ed, 2018;141-145.
61. Raghunath K, Krishnan P. Towards A Secure SDN Architecture. in 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT). 2018;1-7.
62. Ejaz S, Iqbal Z, Shah PA, Bukhari BH, Ali A, Aadil F. Traffic Load Balancing Using Software Defined Networking (SDN) Controller as Virtualized Network function. IEEE Access. 2019;7:46646-46658,.
63. Maulud DH, Zeebaree SR, Jacksi K, Sadeeq MAM, Sharif KH. State of art for semantic analysis of natural language processing. Qubahan Academic Journal. 2021;1:21-28.
64. Jefia A, Popoola S, Atayero A. Software-Defined Networking: Current Trends , Challenges , and Future Directions; 2018.
65. Chippalkatti O, Nimbhorkar SU. An approach for detection of attacks in software defined networks. in 2017 International Conference on Innovations in Information. Embedded and Communication Systems (ICIIECS). 2017;1-3.

66. Zeebaree S, Yasin HM. Arduino based remote controlling for home: power saving, security and protection. International Journal of Scientific & Engineering Research. 2014;5:266-272.
67. Yasin HM, Zeebaree SR, Zebari IM. Arduino based automatic irrigation system: monitoring and sms controlling. in 2019 4th Scientific International Conference Najaf (SICN). 2019;109-114.
68. Kalghoum A, Gammar SM, Saidane LA. Towards a novel cache replacement strategy for named data networking based on software defined networking. Computers & Electrical Engineering. 2018;66:98-113.
69. Priyadarsini M, Bera P, Bampal R. Performance analysis of software defined network controller architecture—A simulation based survey. in 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET). 2017:1929-1935.
70. Iqbal M, Iqbal F, Mohsin F, Rizwan M, Ahamd F. Security issues in software defined networking (sdn): risks, challenges and potential solutions;2019.
71. Liu Y, Zhao B, Zhao P, Fan P, Liu H. A survey: Typical security issues of software-defined networking. *China* Communications. 2019;16:13-31.
72. Abdullah SMSA, Ameen SYA, Sadeeq MA, Zeebaree S. Multimodal emotion recognition using deep learning. Journal of Applied Science and Technology Trends. 2021;2:52-58.
73. Sadeeq M, Abdulla AI, Abdulraheem AS, Ageed ZS. Impact of electronic commerce on enterprise business. Technol. Rep. Kansai Univ. 2020;62:2365-2378.
74. Abdulla AI, Abdulraheem AS, Salih AA, Sadeeq MA, Ahmed AJ, Ferzor BM*, et al.* Internet of things and smart home security. Technol. Rep. Kansai Univ. 2020;62:2465-2476.
75. Parashar M, Poonia A, Satish K. A survey of attacks and their mitigations in software defined networks. in 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT). 2019;1-8.
76. Kaljic E, Maric A, Begovic P, Hadzialic M. A survey on data plane flexibility and programmability in software-defined networking. IEEE Access. 2019;7:47804-47840.
77. Gelberger A, Yemini N, Giladi R. Performance analysis of software-defined networking (SDN). In 2013 IEEE 21st International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems. 2013;389-393.
78. Dargahi T, Caponi A, Ambrosin M, Bianchi G, Conti M. A survey on the security of stateful sdn data planes. IEEE Communications Surveys & Tutorials. 2017;19:1701-1725.
79. Kalkan K, Gur G, Alagoz F. Defense Mechanisms against DDoS Attacks in SDN Environment. IEEE Communications Magazine. 2017;55:175-179.
80. Ujcich BE, Sanders WH. Data protection intents for software-defined networking. in 2019 IEEE Conference on Network Softwarization (NetSoft). 2019;271-275.
81. Rahman A, Islam MJ, Montieri A, Nasir MK, Reza MM, Band SS, *et al.* Smart block-sdn: an optimized blockchain-sdn framework for resource management in IoT. IEEE Access. 2021;9:28361-28376.
82. Vishnevsky V, Pham VD, Kirichek R, Elagin V, Vladyko A, Shestakov A. SDN-assisted unmanned aerial system for monitoring sensor data. in 2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). 2020:313-317.
83. Ruaro M, Caimi LL, Moraes FG. A systemic and secure sdn framework for noc-based many-cores. IEEE Access. 2020;8:105997-106008.
84. Ren C, Bai S, Wang Y, Li Y. Achieving near-optimal traffic engineering using a distributed algorithm in hybrid SDN. IEEE Access. 2020;8:29111-29124.
85. Xu J, Wang L, Xu Z. An enhanced saturation attack and its mitigation mechanism in software-defined networking. Computer Networks. 2019;169:107092.
86. Almohaimeed A, Asaduzzaman A. Introducing edge controlling to software defined networking to reduce processing time. in 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA;2019.
87. Al-Tam F, Correia N. On Load Balancing via Switch Migration in Software-Defined Networking. IEEE Access. 2019;7:95998-96010.

88. Aleroud A, Alsmadi I. Identifying DoS attacks on software defined networks: A relation context approach. In NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium. 2016;853-857.
89. Achleitner S, Bartolini N, He T, Porta TL, Tootaghaj DZ. Fast network configuration in software defined networking. IEEE Transactions on Network and Service Management. 2018;15:1249-1263,.
90. Gao S, Li Z, Xiao B, Wei G. Security threats in the data plane of software-defined networks. IEEE Network. 2018;32:108-113.
91. Chin T, Xiong K, Hu C. Phishlimiter: A phishing detection and mitigation approach using software-defined networking. IEEE Access. 2018;6:42516-42531.
92. Dridi L, Zhani MF. SDN-Guard: DoS attacks mitigation in SDN network. in 2016 5th IEEE International Conference on Cloud Networking (Cloudnet). 2016;212-217.
93. Dinh PT, Park M. BDF-SDN: A big data framework for ddos attack detection in large-scale sdn-based cloud. in 2021 IEEE Conference on Dependable and Secure Computing (DSC). 2021;1-8.
94. Haque MR, Tan SC, Yusoff Z, Nisar K, Lee CK, Chowdhry B*, et al.* SDN architecture for UAVs and EVs using Satellite: A hypothetical model and new challenges for future. in 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC). 2021;1-6.
95. Kundimana G, Vyukusenge A, Tsym A. Networks modernization using sdn and nfv technologies. in 2021 Systems of Signals Generating and Processing in the Field of on Board Communications. 2021;1-5.
96. Almadani B, Beg A, Mahmoud A. DSF: A distributed sdn control plane framework for the east/west interface. IEEE Access. 2021;9:26735-26754.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/68725