Advanced International Journal for Research (AIJFR)
E-ISSN: 3048-7641 ● Website: www.aijfr.com ● Email: editor@aijfr.com

Software Defined Networking vs Traditional Networks: A Comparative Analysis and Results

Nikhil Pandey[1], Pradeep Kr Sharma[2], Brijesh kumar[3]

[1,2] Department of Computer Science and engineering in Shri Ram murti smarak college of engineering technology, Bareilly, Uttar Pardesh
[3]Department of Computer Science and Application in Manav Rachna International institute of research and studies Faridabad, Haryana

Abstract

SDN is an approach of creating new solutions, innovative applications and deploying them over the application plane rather than providing a hardware-based point solution. In SDN, control plane works as a logically centralized control unit which works like a brain of the network and Application plane is the heart of SDN architecture. This article describes the history of programmable networks and its architecture with the role of different planes like data plane, control plane, and application plane. This paper focused on the qualitative investigation as well as quantitative results and provides a comparative analysis between traditional and SDN on the basis of architecture, protocol and infrastructure layer. It has been proposed that SDN is the new norm of networking and a future technology which provides programmability, flexibility and automation with logically centralized control where TN is lacking behind. Mininet simulator has been used for quantitative results like round-trip time, bitrate, bandwidth on TCP and UDP port. This article provides a fine tune research direction to the researchers and network administrators in field of SDN.

Keywords: network virtualization, round-trip time, bandwidth

1. Introduction

In early days it's like impossible to create the new protocols and deploying over the application plane because of "Internet ossification"[1]. SDN does not challenge any functionality like routing, reliability, mobility and real time communication but only provide the feature of innovating new solutions and deploying it using different APIs [2] (e.g. Open Flow [3][4], etc.). The main concept of SDN is decoupling the control planes (devices that make the decision, where does traffic go?) from the data planes [5] (devices that provide the path to traffic to the destination) via SDN controllers [6] which acts like NOS (as same like server OS that provides the high-level abstraction for the implementation). Software Defined Network is one of the emerging technologies which increased network' s security and as well come up with the affordable chances for dynamic updating and upgrading in response to future [7]. The idea behind of SDN is that control plane should be completely separate from data plane and for increasing the capacity of data plane, add more bare metal switch (hardware devices without any software layers and OS) [8] and for control plane; add more compute servers. Open Networking

Foundation (ONF) [9] and International Telecommunication Union (ITU) have been recently working on the standardization of SDN networks [10].

SDN is not all about the solution; it's an approach for disaggregating (disaggregation means operator purchase the control plane from vendor X and data plane from vendor Y) the control plane from data plane via an open interface. Control plane maintains the routing table known as Routing Information Base (information about the best path) and forwarding table known as Forwarding Information Base (optimized for fast packet processing) maintained by data plane.

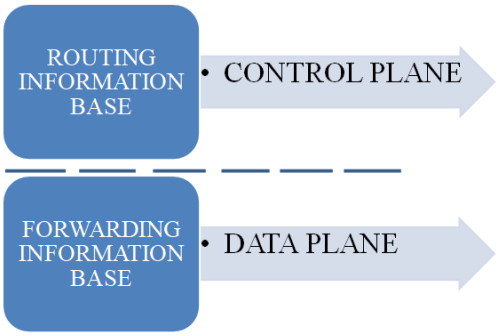

Fig1.2 Disaggregating the control plane from data plane

HISTORY OF PROGAMMABLE NETWORKS

Starting in 2004, **4D Project** was another step towards the distinction between protocols and routing decision logic that are controlled by network components such as early programmable networks like **Open Signaling** (OPENSIG), **GSMP** (a protocol which controls the label switch). **DCAN**, idea worked as an inspiration for the **NOX (**Operating system for networks) [11].

In 2006, IETF was proposed **NETCONF** [11][12] as a new approach for network which can modify the configuration of network devices and as well as expose the APIs. In late 80s, there was a very popular protocol known as structured network management protocol (SNMP) but after sometime operators felt some drawbacks regarding SNMP and one of major shortcoming was lack of security. NETCONF fixed that drawbacks later. NETCONF did not separate control plane from data plan that's why it did not concerned as fully programmable network. After this, another initiative was introduced which known as **Ethane project** [11] which provided a new architecture for the enterprise network and contained two parts: an ethane switch that provides the controller with the flow table and secure channel and a controller that determines whether a packet should be transmitted.

In 2007, A Spanish born and American software Engineer, **Martin Casado** gave a new programmable approach of networking in his research paper "SANE" which described the theory of single protection layer. It governs the all connectivity and routing and all control decisions access by logically centralized server. Ethane project led the creation of the Open Flow and its API was created in 2008. In this journey of research various emulators were introduced like vSDNEmul, Estinet and Mininet [13]. In 2010, SDN is used to create virtual networks by the team NICIRA (a company co-

founded by Martin Casado, focused to SDN and network virtualization) which was described in NSDI [14]. In 2012, NICIRA was acquired by VMware for $1.26 billion but this deal was closed in following months and after that NICIRA was merged with VMware switches and known as NSX in market. In 2012, ONF [9] was introduced which is an organization where multiple vendors are working on software defined networking and programmable interface like OpenFlow.

In 2014, AVAYA used Open Stack and Shortest path bridging to showcase Software defined Networking.

RELATED WORK

Mohammad Nowsin Amin Sheikh et al. proposed a research architecture which shows a quantitative investigation between a centralized network and a distributed network on the basis of different parameters like bandwidth, latency using Mininet emulator and according to this investigation, the OpenDayLight controller does well in terms of throughput, bandwidth and jitter whereas the Ryu controller outperforms it in terms of latency, packet loss and round-trip time [15]. M. Awais et al. proposed a model in which they compare the traditional network from software defined network on the basis of architecture, layers, protocols, infrastructure and different emulators and vendors and it also describes the various SDN controllers and communication between the controllers [16].

This article uses the Raspberry Pi 3 embedded system to compare SDN and Traditional Networks and even SDN offers respectable response times and performance numbers and the traditional network outperforms it with TCP data flows of 96.9 Mbits/sec and UDP data flows of 99.9 Mbits/sec and average response times of 17.48 ms [17]. This research study proposes a new novel model which incorporates the SDN concept to supplant the Traditional Network concept, in order to streamline the network bandwidth management and prioritize Sensitive Traffics to get the network throughput which is also compared with Traditional Network [18]. This model applies test on the basis of RFC 2544 methodology for showing comparison between the performance of traditional networks and Software-defined networks for obtaining the complete performance on every strategy in proposed Topology by applying on each technology, after that analyze the capability of each one and with its advantages and disadvantages [19].

TRADITIONAL NETWORK AND SDN ARCHITECTURE

This section defines the general architecture of traditional networking and SDN and as well as trying to discuss how SDN is different and beneficial than traditional networking and also about multiple components and tools which plays an essential role in the architecture.
The figure shows a comparison between both network architecture is given below [27].

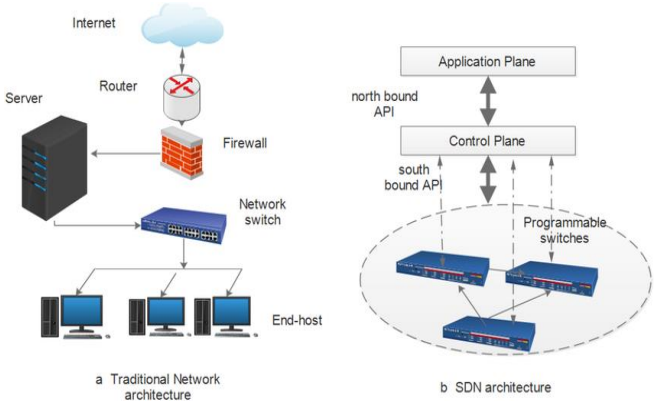

Fig 2.1 comparison between traditional architecture and SDN architecture

In Traditional networking, data flow will be managed by routers and switches and its architecture consist the different planes like data plane, control plane and management plane.
The main task of Data plane is generating the forwarding table and control plane generate the routing table. Data plane is used for forwarding the data packets from source to the destination and control plane managed the data packets over the network and management plane behaves like an administrator [20].

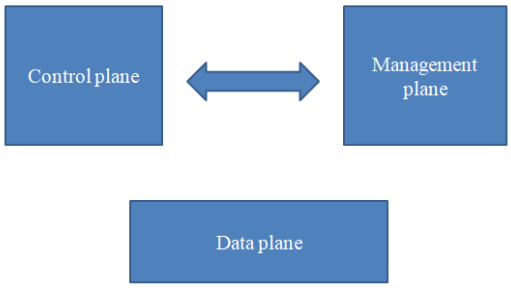

Fig 2.2 Traditional Network Architecture

In the process of decoupling the control plane from data plane where controller works as an actor [21]. Control plane behaves like a centralized unit in software defined networking. Due to the centralized control plane, administrators have to do more work to ensure general network security and regular operations [22].

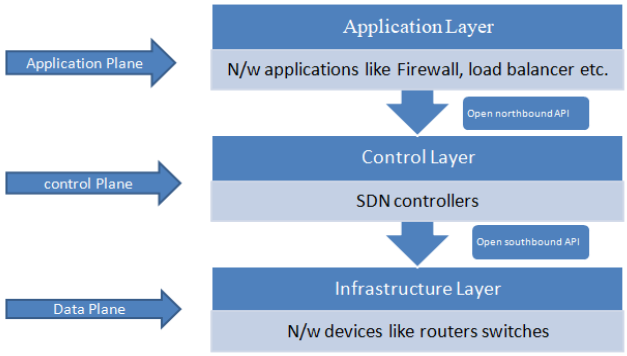

Fig 2.3 Basic Architecture of SDN

- **Application Plane**

  Application plane is called as the heart of this Paradigm in SDN. Northbound API [23] works like a link between the Application and the SDN controllers where application tell the requirements to the network and network delivers that resources.
  It gives a standardized interface for building and implementing smart applications that may improve network performance through using network data and analytics [24].

- **Control Plane**

  Control plane makes forwarding decisions which provides the quality of services. It handles flow control as well as make decisions through control logic and it sets rule to the management of routing devices in the Data plane [25].

  The Southbound API is the medium of communication. This plane consists the SDN Controller which comprised of multiple functional components to manage the behavior of network. Network requirements for application are translated for network element resources by SDN control logic using a controller [26].

- **Data Plane**

  Data plane works on infrastructure layer consisting the devices which forwards the data packets or traffic to the destination. It refers as a forwarding plane or user plane. Basically, we can say it is used to forwarding the data packets from source to the destination. A lot of devices like routers, switches are present in this plane.

TABLE 2.
Comparative analysis of infrastructure layers

Software defined networking	Traditional networking
This model has been proposed by Open networking foundation (ONF)	It has been followed by OSI model.
It comprises three layers: application, control and physical layer.	It follows stratified infrastructure that includes core, distribution and access layer.

TABLE 3.
overall comparison of SDN and traditional

Attributes	Software Defined Networking	Traditional Networking
Control logically Centralized	✓	×
Programmability	✓	×
Flexibility	✓	×
Performance	✓	×

SDN APPLICATIONS

Now this section discusses initial applications of SDN and some creative implementation at advanced phase.

- **SDN WAN**

This article [27] introduces the first and largest software defined network' s deployment in the form of SDN WAN model B4 as shown in figure 3.1 which shows a 3- Tier architecture. switch hardware layer which consists the switches to help the run the OpenFlow and enhance the new solutions for forwarding traffic and the site controllers' layer consisted OFC and NCS.

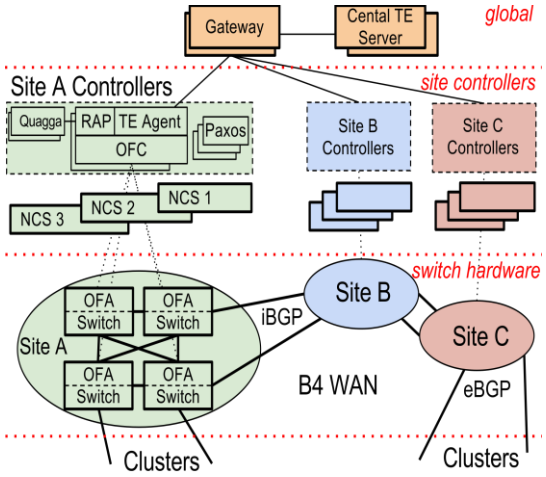

Fig 3.1 SD WAN B4 architecture

In which the OFC takes an eye on the network status through commands from network control applications and switch events [27].

The global layer takes the responsibility for centralized control to whole network with the help of SDN gateway and central traffic engineering and it also receives the link information from OFC and provide the abstracts for the TE server for better understanding of global path information [27] [16].

- **Network virtualization**

It is easy to reveal the single API to create, modify and delete the virtual network by separating the control plane from the data plane Because network virtualization set out to provide a full package of network applications in a programmatic way, its impact went beyond the simplification and automation of network provisioning [14].

In recent years, a more flexible encapsulation called GENEVE (Generic Network Virtualization Encapsulation) has emerged. For better understanding a figure of network virtualization system is given below.

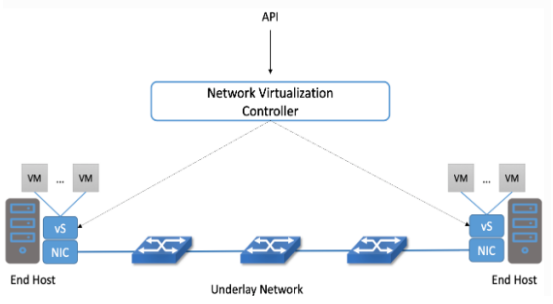

Fig 3.2 An architecture of Network Virtualization

➢ **Access Networks**

Access network is also a part of the network which provides the internet connectivity to the users. There are majorly two types of Access Networks: PON (Passive Optical Networks) which provides the FTTH (fiber to the home) services and RAN (Radio Access Networks) provides the feature of mobile cellular networks. Using SDN, control plane is disaggregated from the physical devices and provides the centralized management, quality of services.

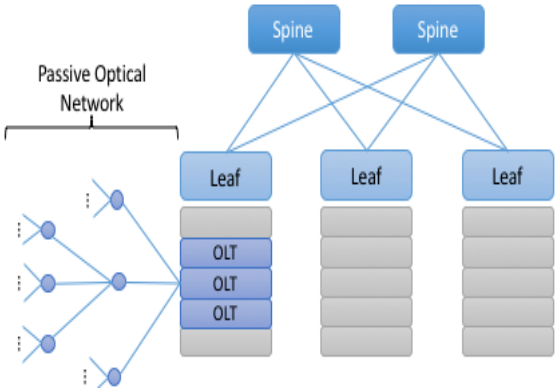

Fig 3.3. General hardware architecture of SEBA: SDN-Enabled Broadband Access

EXPERIMENTAL AND RESULT

In the paradigm of software defined network, this paper reached to the final chapter where it compares both networks; software defined network and traditional network in real time.

the Mininet emulator is used for completing the task where a Software defined Network of two host (h1, h2), controller (c0) and virtual switch(s1) is created by using command **Sudo mn** on emulator.

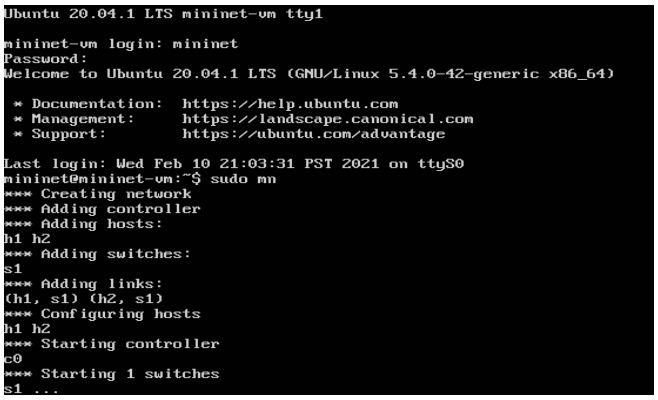

Fig 4.1 creates a software defined network

After that checking the connectivity of all nodes by pinging the network using **pingall** command on Mininet emulator.

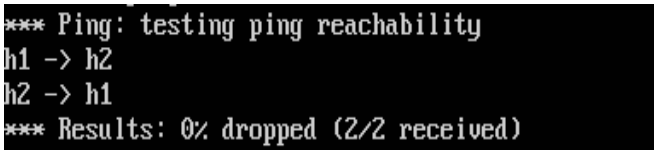

Fig 4.2 checks the connectivity between nodes

Here for checking the connectivity for specific node by using the command **h1 ping h2** for knowing how many packets received or lost into the network.

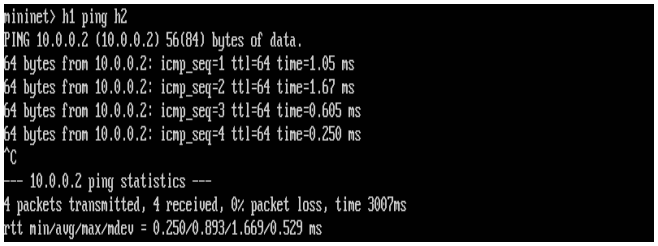

Fig 4.3 ping for the specific node

Here the results are found;
time = 3007 ms
min time =0.250 ms
avg time = 0.893 ms
max time =1.669 ms
mean deviation=0.529 ms

In traditional network, take two PCs of different IP for calculating the round-trip time.

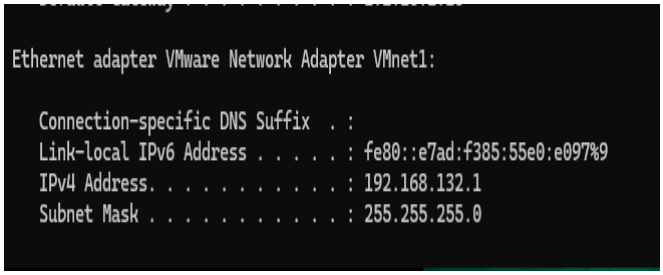

Fig 4.4 PC1 IP address

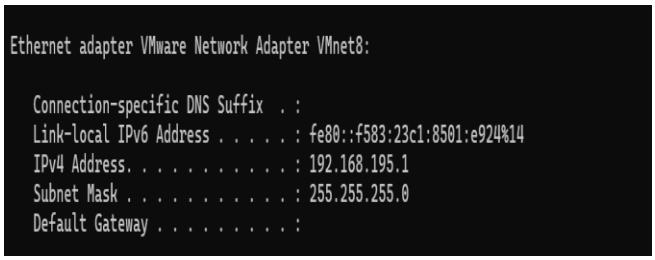

Fig 4.5 PC2 IP address

Now, check the connectivity to one pc to another.

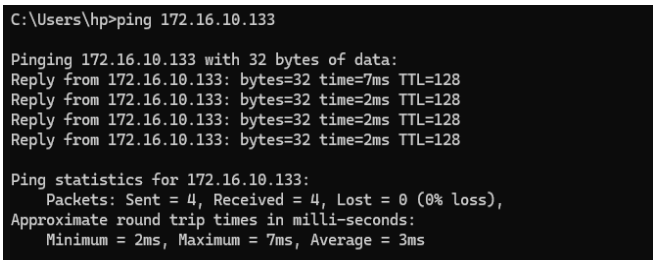

Fig 4.6 calculates the round-trip time

Here the results are;
Minimum= 2ms
Maximum=7ms
Average =3ms

After comparing both the network on the basis of round-trip time. It is very easy to say, the performance of software defined network is far better than traditional network. On the basis of round-trip time, we draw a graph between min, max, avg scenario.

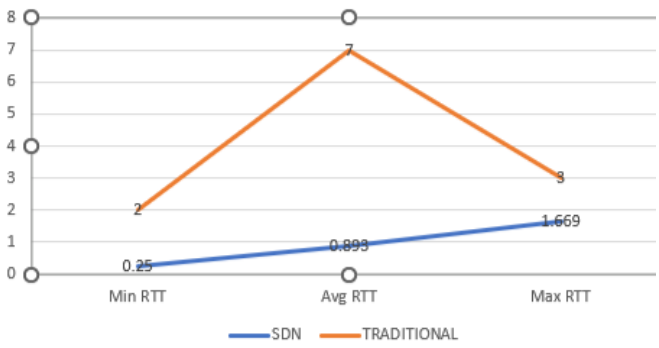

Fig 4.7 Comparison of round-trip time in different networks

For measuring the bandwidth between hosts type a command **iperf** on Mininet. In this paradigm first start the iperf server on one host by using command **h1 iperf -s &** where **-s** refers to the server and **&** helps to run it in the background.

Now run the **iperf** on one host by using command **h2 iperf -c h1** on Mininet.

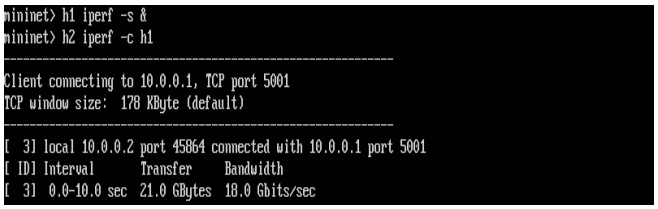

Fig 4.8 bandwidth in TCP Test

Here results on given ports;
Interval= 0.0-10.0 sec
Transfer=21.0 Gbytes
Bandwidth= 18.0 Gbits/sec

Now find the results in UDP test instead of TCP used the command **h2 iperf -c h1 -u** on Mininet emulator.

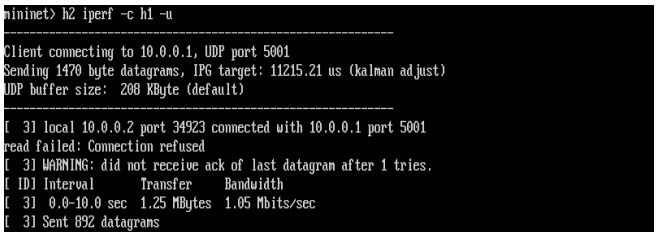

Fig 4.9 bandwidth in UDP Test

Here the results;
Interval= 0.0-10.0 sec
Transfer=1.25 Mbytes
Bandwidth= 1.05 Mbits/sec

Further, find the bitrate (which is the actual rate of data being transmitted at a given time) where windows as client and Mininet virtual machine as server with the help of **iperf3.**

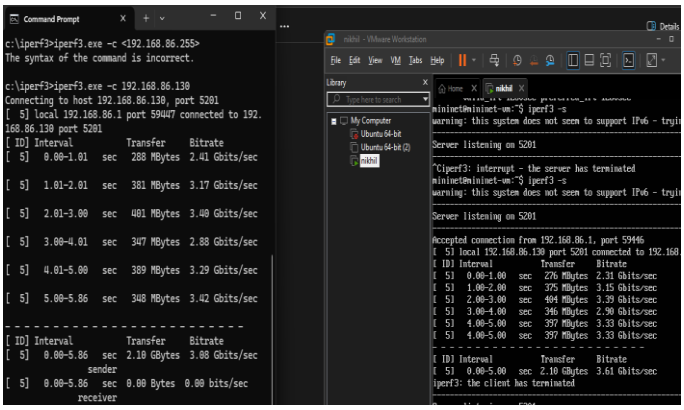

Fig 4.10 bandwidth between the client and server (Mininet)

Here the results as;
Interval = 0.00-5.86 sec
Transfer =2.10 Gbytes
Bitrate = 3.88 gbits/sec

Then perform the reverse process where windows act as server and Mininet virtual machine as client.

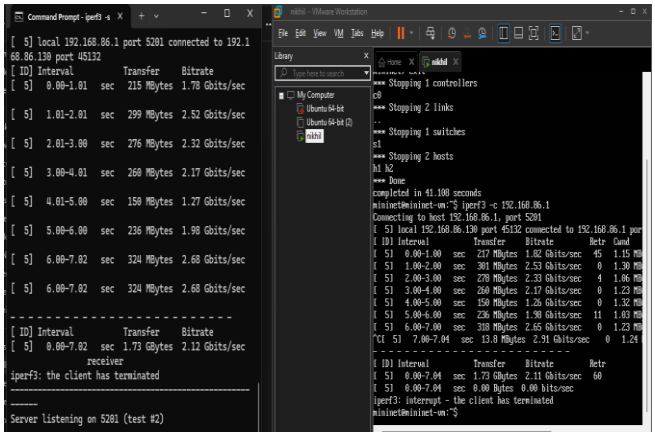

Fig 4.11 bandwidth between the client (Mininet) and server

Here the results are given below;
Interval = 0.00-7.62 sec
Transfer=1.73 Gbytes
Bitrate =2.12 Gbits/sec

Conclusion

This research performs the qualitative investigation and as well as quantitative results. In qualitative investigation, both the networks are compared on the basis of architectures, infrastructure layers. On the

basis of above investigation, software defined network provides a centralized control, programmability and flexibility as compared to traditional network.

The quantitative results provide the comparative analysis between the centralized network and the distributed network on the parameters of round-trip time, bandwidth, bitrates using Mininet emulator. In which software defined network takes a lot time initially but after sending one or two packets it improves the performance but in traditional, round-trip time remained same from starting to the last. The bandwidth is measured on TCP port 18.0 Gbits/sec and in UDP test, it is 1.05 Mbits/sec which is better on TCP port than UDP port. In this process windows machine worked as client and Mininet emulator as server and vice versa. In this process Mininet as server provides the good result as compare to Mininet as client.

References

1. B. A. A. Nunes, M. Mendonca, X. -N. Nguyen, K. Obraczka and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," in IEEE Communications Surveys & Tutorials, vol. 16, no. 3, pp. 1617-1634
2. W. Zhou, L. Li, M. Luo and W. Chou, "REST API Design Patterns for SDN Northbound API," 2014 28th International Conference on Advanced Information Networking and Applications Workshops, Victoria, BC, Canada, 2014, pp. 358-365
3. A. Lara, A. Kolasani, and B. Ramamurthy, “Network innovation using openflow: A survey,” Communications Surveys & Tutorials, IEEE,vol. 16, no. 1, pp. 493–512, 2014.
4. K. Suzuki, K. Sonoda, N. Tomizawa, Y. Yakuwa, T. Uchida, Y. Higuchi,T. Tonouchi, and H. Shimonishi, “A survey on OpenFlow technolo-gies,” IEICE Transactions on Communications, vol. 97, no. 2, pp. 375–386, 2014.
5. Y. Jarraya, T. Madi and M. Debbabi, "A Survey and a Layered Taxonomy of Software-Defined Networking," in IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 1955-1980.
6. M. Paliwal, D. Shrimankar and O. Tembhurne, "Controllers in SDN: A Review Report," in IEEE Access, vol. 6, pp. 36256-36270.
7. T.koponen, “network virtualization n Multi-tenant Datacenters”, NSDI, April 2014.
8. Kurtz, N. Dorsch and C. Wietfeld, "Empirical comparison of virtualized and bare-metal switching for SDN-based 5G communication in critical infrastructures," 2016 IEEE NetSoft Conference and Workshops (NetSoft), Seoul, Korea (South), 2016.
9. C. Trois, M. D. Del Fabro, L. C. E. de Bona and M. Martinello, "A Survey on SDN Programming Languages: Toward a Taxonomy," in IEEE Communications Surveys & Tutorials, vol. 18, no. 4.
10. K. Raghunath and P. Krishnan, "Towards A Secure SDN Architecture," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bengaluru, India, 2018, pp. 1-7
11. A.T. Campbell, I. Katzela, K. Miki, and J. Vicente. Open signaling for atm, internet and mobile networks (opensig’98). ACM SIGCOMM Computer Commun. Review, 29(1):97–108, 1999.
12. R. Enns. NETCONF Configuration Protocol. RFC 4741 (Proposed Standard), December 2006. Obsoleted by RFC 6241.

13. R. L. S. de Oliveira, C. M. Schweitzer, A. A. Shinoda and Ligia Rodrigues Prete, "Using Mininet for emulation and prototyping Software-Defined Networks," 2014 IEEE Colombian Conference on Communications and Computing (COLCOM), Bogota, Colombia, 2014.
14. T.koponen, "network virtualization n Multi-tenant Datacenters", NSDI, April 2014.
15. Sheikh, Mohammad Nowsin Amin, I-Shyan Hwang, Muhammad Saibtain Raza, and Mohammad Syuhaimi Ab-Rahman. 2024. "A Qualitative and Comparative Performance Assessment of Logically Centralized SDN Controllers via Mininet Emulator" Computers 13
16. Sushant Jain, Alok Kumar, Subhasree Mandal, et al, "B4: Experience with a globally-deployed software defined wan," ACM SIGCOMM,vol. 43,issue 4,pp. 3–14,2013.
17. A. C. Jaramillo, R. Alcivar, J. Pesantez and R. Ponguillo, "Cost Effective test-bed for Comparison of SDN Network and Traditional Network," 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC), Orlando, FL, USA, 2018, pp. 1-2
18. E. R. Jimson, K. Nisar and M. H. bin Ahmad Hijazi, "Bandwidth management using software defined network and comparison of the throughput performance with traditional network," 2017 International Conference on Computer and Drone Applications (IConDA), Kuching, Malaysia, 2017, pp. 71-76
19. A. Almazan, N.soriano, "A comparison of Traditional Network and Software-defined Network schemes using Open Flow protocol" in WSEAS TRANSACTIONS on COMPUTERS Volume 18, 2019.
20. F.Alam, I.Katib and A.Alzahrani, " New networking era: software defined networking" in International Journal of Advanced Research in Computer Science and software engineering , vol. 3, November 2023.
21. BusinessWire. (2020). Global Software-Defined Networking Market (2020 to 2025)—Software-Defined Networking for 5G Presents Opportunities.
22. M. B. Jiménez, D. Fernández, J. E. Rivadeneira, L. Bellido and A. Cárdenas, "A Survey of the Main Security Issues and Solutions for the SDN Architecture," in IEEE Access, vol. 9, pp. 122016-122038, 2021.
23. F. Bannour, S. Dumbrava and D. Lu, "A Flexible GraphQL Northbound API for Intent-based SDN Applications," NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, 2022.
24. T.Aditya, A.David and G.Thipanna, "SDN Application Plane Where Intelligence Meets Networking" in International Journal of Advanced Research in Science, Communication and Technology, Volume 3, pp. 512-519, February 2023.
25. A. Abuarqoub , "A Review of the Control Plane Scalability Approaches in Software Defined Networking" in future internet in 2020.
26. M.Karakus, A.Durresi, "A Survey: Control Plane Scalability Issues and Approaches in Software-Defined Networking (SDN)" in computer networks pp.279-293, November 23, 2016.
27. Tong Li. Jinqiang chen,hongyong Fu, "Application Scenarios based on SDN: An Overview" in IOP Conf. Series: Journal of Physics: Conf. Series in IOP Conf. Series: Journal of Physics 2019.