Design And Development of a Network Mechanism for The Issues and Challenges of MPLS, BGP and SDN

HAFSA FATIMA UMAMA[1], Dr. SAMEENA BANU[2]
[1, 2] Research Scholar, Career Point University

Abstract— Since its emergence a few years ago, the software-defined networking paradigm has attracted the interest of both business and academia, and it continues to do so today. It is possible to design software-defined networking architecture because of the decoupling between the network control plane and the data plane. With Software Defined Networks (SDN), operators may decrease deployment time, increase flexibility, and adjust network resources on-the-fly based on changing consumer demands. Though SDN's concentration of intelligence has exposed vulnerabilities, research on forwarding traffic and reconfiguration problems has mostly ignored the control plane's fault management elements, despite this. An improved BGP, MPLS, and SDN method will be developed by reviewing different aspects of these protocols.

Indexed Terms- BGP, MPLS and SDN, Software Defined Networks, Programmable networks, etc.

I. INTRODUCTION

Fast networks that can handle a lot of traffic are required for today's Internet applications, as well as the ability to install many different, dynamic apps and services. With the advent of "inter-connected data centers" and "server virtualization," network demand has skyrocketed. Besides proprietary network hardware, distributed protocols and software components, legacy networks are flooded with switching devices that decide on the route taken by each packet individually. Furthermore, data paths and decision-making processes for switching or routing are co-located on the same device.

Given the widespread use of distributed systems, system management knowledge is both costly and hard to come by.. As a result, when faced with numerous failures in a short period of time, such as minutes, experts may not be able to respond quickly enough. For example, a command typed improperly deleted a significant number of servers, causing problems in other critical subsystems in Amazon s3 service in 2017. In order to allow system self-healing, quick automated solutions are required. SDN, which stands for Software-Defined Networking, has emerged as a major issue solution. With this method, the control plane and data plane are separated into a distinct network architecture, and greater flexibility is added to the central network controller side. A network or a commercial application is included in the application layer. To make network administration easier and foster creativity, numerous novel networking ideas have been developed as a result of the rise of cloud computing. One of the cloud model's accepted ideas is the rise of the software-defined networking (SDN) paradigm, which eliminates network infrastructure maintenance procedures and guarantees simple administration. So, SDN provides real-time performance while still meeting high availability needs. This new emergent paradigm, however, has been confronted with many technical challenges, some of which are intrinsic and others which have been passed down from previously accepted technology.

II. ISSUES AND CHALLENGES

This section discusses the features, advantages, disadvantages, Issues and Challenges of above-discussed mechanisms: -

MPLS/SDN-Cost: In contrast to transmitting traffic via the public Internet, MPLS requires the purchase of a carrier's service and therefore is much more costly. However, SDN allows utilizing multiple, high bandwidth and inexpensive internet connections.

MPLS-Speed and Security: When it comes to MPLS, the benefits include scalability, flexibility, greater use of transmission capacity, decreased congestion in the network, and a better end-user experience. MPLS is a virtual private network (VPN) rather than an encryption scheme, and as such is isolated from the rest of the Internet. As a result, MPLS is regarded as a secure method of transport. Denial-of-service attacks on pure-IP networks aren't a problem since it's immune to them.

BGP/SDN-Convergence Speed: Furthermore, the research explored that the BGP possess many advantages along with certain problems like routing table growth, instability, slower convergence, and security. These problems can also be solved by appending the network architecture with SDN. Due to the presence of centralized controller in SDN, the state propagation process can be accelerated and due to global overview present at the controller, the decision for the alternate path will be based upon the current updates.

MPLS/SDN-Performance: Because MPLS preset routes are fixed circuits rather than dynamic ones like SDN, making modifications to them is more difficult after they have been put in place. However, as soon as an MPLS network is put in place, assured real-time performance may be expected. On the other edge, SDN uses the public internet and public internet is more vulnerable to packet loss, latency, and Jitter and hence no performance guarantees. Hence MPLS is better for real-time applications over the internet and SDN vouches for better performance in terms of scalability, availability along with resource utilization.

2.1 SDN/Existing Network-Comparison

The advantage of SDN is that an enterprise network traffic architect can sit at a central point and easily apply policies across all WAN devices. Table 1 summarizes the comparison between traditional and Software Defined Networking.

Table 1: Comparing Traditional/Existing and Software Defined Networks

	SDN	Existing Network
Configuration Control	User	Hardware Vendor
Network Perspective	Software Dominated	Hardware Dominated
Interlock Compatibility	Standardized Protocol	Independent Protocol
Technology Openness	Open Structure	Closed Structure
Market Fairness	Fair Competition	Monopoly
New Technological Adoption	Acc to users' needs	Acc to the vendor needs
Managerial efficiency	High-efficiency / Logical Operation	Low-efficiency/ high-cost operation

2.2 SDN Controllers Comparison

Furthermore, the SDN is equipped with various controllers. Table 2 shows a comparison of the three controllers on 'Mininet' a network emulator for SDN, controllers available in SDN and it also depicts that the Network Performance with 'Iperf' tool. It depicts that we can implement MPLS as well as BGP only through RYU Controller.

Table 2: Comparison Summary of SDN Controllers

	RYU	NOX	POX
Performance	Slow	Fast	Slow
Language	Python	C++	Python
MPLS Library	Yes	No	Yes
BGP Library	Yes	No	No
OpenFlow	1.0 -1.4	1.0/1.3	1.0

For further clarification, the topology shown in Fig. 11 has been implemented with POX as well with RYU controller on 'Mininet' a network emulator for SDN and recorded the Network Performance with 'Iperf ' tool.

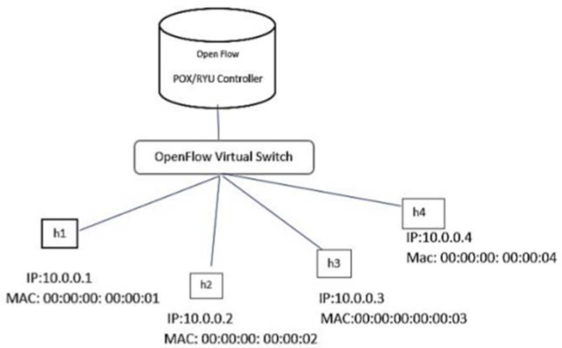

Figure 1: Single Switch 4 hosts topology

Controllers Comparison-Parameters

Mininet has been used as the network emulator in this study because of its ease of use. The virtualization feature of Mininet allows researchers to create a bespoke virtual network testbed on a single Linux kernel for their research purposes. Mininet is used to test the experimental topology shown in Figure 2. Using this architecture, you can see four different hosts, each with its own IP and MAC addresses. They all connect to an OpenFlow Virtual Switch (version 2.0.2). The Switch is further controlled by the POX (version 0.2.4) first and then with Ryu (version 4.10) controller. Table 3 summarizes the values of various parameters for POX and RYU, which have been received through Iperf-a network monitoring tool.

Table 3: Comparison of POX and RYU Controller

Parameters	RYU	POX
Throughput	5.81Gbit/sec	5.01Gbit/sec
Round Trip Time (RTT)	0.036(ms)	3.012(ms)
MPLS Library	Yes	Yes
BGP Library	Yes	No
Web Server latency	3.5(ms)	4.6(ms)

From the above study, it has been cleared that for designing and developing an Enhanced Network Path Restoration Mechanism, the RYU controller possesses the potential of libraries of BGP and MPLS, appended with less RTT and Web Server Latency value.

III. SIMULATION RESULTS

The discussion above mentioned vouched for an integrated approach of MPLS, BGP and SDN. Fig. 2 depicts the proposed architecture to be implemented for achieving results.

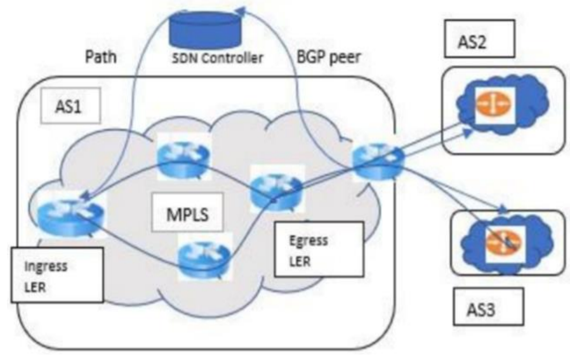

Figure 2: Proposed architecture

To implement the above architecture, Figure 3 shows the scenario designed and configured in GNS3-a network emulator and packets are captured and analyzed with the Wireshark-a network protocol analyzer.

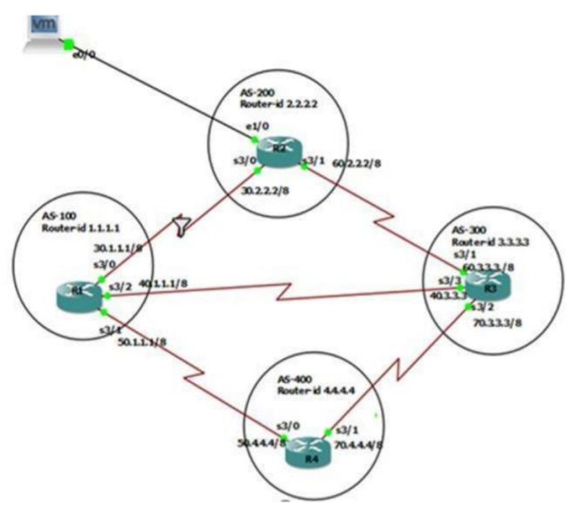

Figure 3: Topology in GNS3

After the detailed study and experiments, various mechanisms are compared on the basis of some crucial parameters such as Round-Trip Time, Throughput and network lifetime etc. Fig. 4 exhibits the comparison of Round-Trip Time (RTT) between BGP, MPLS+BGP and MPLS+BGP+SDN networks. The analysis is as follows:

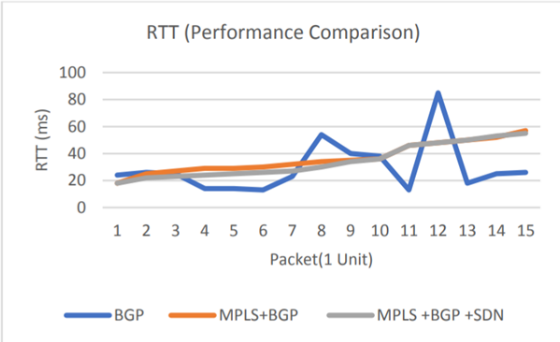

Figure 4: Round Trip Time

It can be clearly observed that the assimilated approach of BGP, MPLS and SDN take less Round-Trip time. The graphs are shown in Fig. 5 and Fig. 6 for delay and throughput which exhibits, that assimilated approach of various mechanisms are not showing much better performance as compared with BGP alone. The reason for this is that the SDN controller may add extra delay for the communication between the client node and an SDN controller which further affect throughput also. However, the SDN controller provides flexibility for customization of routing mechanisms through their built-in libraries.

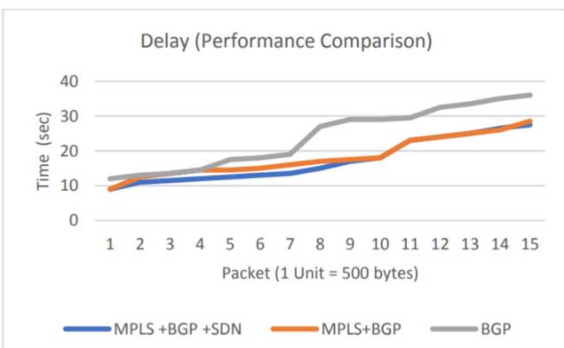

Figure 5: Delay (Performance Comparison)

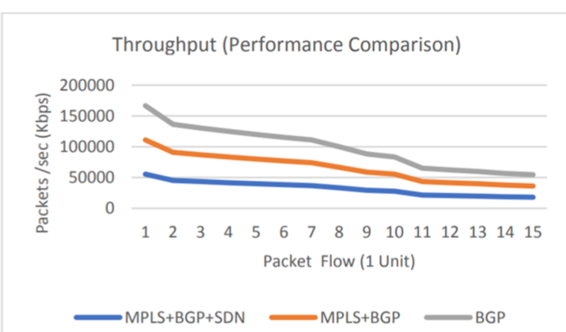

Figure 6: Throughput (Performance Comparison)

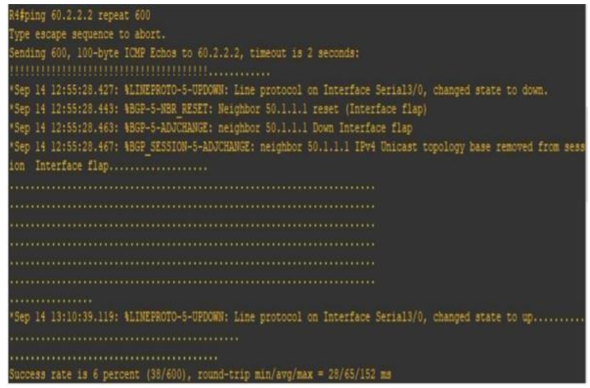

Figure 7. Network Path Restoration with BGP\

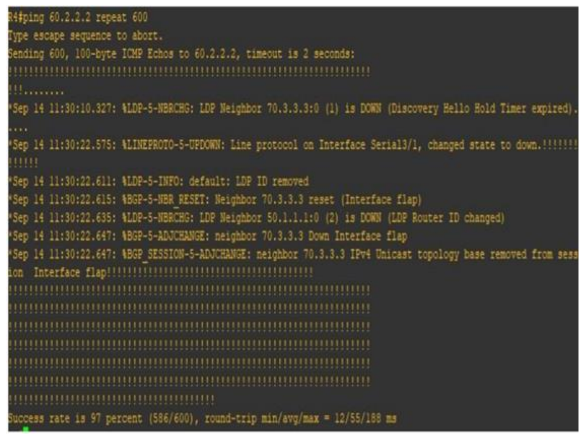

Figure 8. Network Path Restoration with MPLS+BGP+SDN

Fig. 7 and Fig. 8 shows the time taken by routers for network path restoration if one node/interface on the path towards destination fails. This clearly shows that the router configured with BGP alone takes much time for path restoration with much loss in packets as compared to the integrated approach.

CONCLUSION

Software-defined networking (SDN) has several advantages, and one of them is the availability of network services. Any SDN application may make use of these services since they are reusable parts of the network. Topology discovery is one of the most important SDN functions, aided by the logically centralized SDN controllers' global view. In this paper, a detailed discussion is to design a Network Mechanism, where the architecture must be designed with SDN and the routing support must be given by

BGP Routing Protocol with the underlying mechanism of MPLS. This implementation turns out to be significant for decreasing network convergence delay when the network failures are frequent and link reconnection delays are higher. It has been found that assimilated approach of mechanisms has much potential as an enhanced Network mechanism with much throughput, minimal packet loss and least restoration time.

REFERENCES

[1] A. C. Baktir, C. Tunca, A. Ozgovde, G. Salur, C. Ersoy, SDN-based multi-tier computing and communication architecture for pervasive healthcare, IEEE Access, 6 (2018), 56765–56781.

[2] A. Hussein, A. Chehab, A. Kayssi, I. H. Elhajj, Machine learning for network resilience: The start of a journey, in 2018 Fifth International Conference on Software Defined Systems (SDS), (2018), 59–66.

[3] A. Khamparia, D. Gupta, V. H. C. De Albuquerque, Internet of health things-driven deep learning system for detection and classification of cervical cells using transfer learning, J. Supercomput, 76 (2020), 8590–8608.

[4] A. Mauthe, D. Hutchison, E. K. Cetinkaya, I. Ganchev, J. Rak, James P. G. Sterbenz, et al., Disaster-resilient communication networks: Principles and best practices, in 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM), (2016), 1–10.

[5] A. Modarresi, S. Gangadhar, J. P. G. Sterbenz, A framework for improving network resilience using SDN and fog nodes, in 2017 9th International Workshop on Resilient Networks Design and Modeling (RNDM), (2017), 1–7.

[6] C. Iwendi, S. Khan, J. H. Anajemba, M. Mittal, M. Alenezi, M. Alazab, the use of ensemble models for multiple class and binary class classification for improving intrusion detection systems, Sensors, 20 (2020), 2559.

[7] E. Rojas, G. Ibáñez, J. M. Gimenez-Guzman, J. A. Carral, A. GarciaMartinez, I. Martinez-Yelmo, and J. M. Arco, “All-Path bridging: Path exploration protocols for data center and campus networks,” Computer Networks, vol. 79, pp. 120 – 132, 2015.

[8] F. Pakzad, M. Portmann, W. L. Tan, and J. Indulska, “Efficient Topology Discovery in OpenFlow-based Software Defined Networks,” Comput. Commun., vol. 77, no. C, pp. 52–61, Mar. 2016.

[9] Hyojoon Kim; Feamster, N., "Improving network management with software defined networking," Communications Magazine, IEEE, vol.51, no.2, pp.114,119, February 2013

[10] Kempf, J.; Bellagamba, E.; Kern, A.; Jocha, D.; Takacs, A.; Skoldstrom, P., "Scalable fault management for OpenFlow," Communications (ICC), 2012 IEEE International Conference on, vol., no., pp.6606,6610, 10-15 June 2012

[11] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O’Connor, P. Radoslavov, W. Snow, and G. Parulkar, “ONOS: Towards an Open, Distributed SDN OS,” in Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, 2014.

[12] Ryu. (2012) Ryu SDN framework. [Online]. Available: http://osrg.github.com/ryu/

[13] S. Bhattacharya, P. K. R. Maddikunta, Q. V. Pham, T. R. Gadekallu, C. L. Chowdhary, M. Alazab, et al., Deep learning and medical image processing for coronavirus (COVID-19) pandemic: A survey, Sustainable Cities Soc., 65 (2021), 102589.

[14] S. Wallin, “Rethinking Network Management: Models, Data-Mining and Self-Learning”, Ph.D dissertation, Dept. Comp. Sciences, Luleå, Sweden, 2012.

[15] T. R. Gadekallu, N. Khare, S. Bhattacharya, S. Singh, P. K. Maddikunta, I. Ra, et al., Early detection of diabetic retinopathy using PCA-firefly based deep learning model, Electronics, 9 (2020), 274.