Software-Defined Networking (SDN)

Fahmeda Khanom, Touheda Khanom
New York City College of Technology, New York, USA
{Fahmeda.Khanom,Touheda.Khanom}@mail.citytech.cuny.edu

Abstract

Software-Defined Networking (SDN) is a networking architectural model which is programmed to manage network systems. Currently, technology is widely used and lots of hardware components such as routers and switches are used for data communication which overloads the network and slows the process. SDN enables a dynamic network to improve network performance and can monitor the network for any failure and security issues. Different APIs such as OpenFlow are used by SDN for easier communication between controller and switch. Lastly, developers need to have automation, programming, and troubleshooting skills to work on this new networking approach SDN.

Keywords

Software-Defined Network(SDN), OpenFlow, API, SDN Controller, Control Plane, and Topology

I. Introduction

Software-Defined Network (SDN) is a smart network architectural approach to control and set up networks centrally using programming and software applications. Currently, network computer systems are rapidly programmed and controlled using SDN. Application Programming Interfaces (APIs) are being used to allow applications to interact with outside software applications to get access to resources. Data and resources within the network are handled using OpenFlow APIs. This paper will address the importance of SDN in today's fast-evolving technology period and will educate people about the importance of SDN and the tools and software available.

II. Advantages and Disadvantages

According to the Software-Defined Network paper by Keith Kirkpatrick "SDN permits more efficient use of network resources to support virtual

machines, as well as greater flexibility, via OpenFlow virtual switches"(Kirkpatrick, 2013). Virtualization is an effective way to keep the network system under supervision and can be programmed to prevent any kind of security threat by installing security policies on each network. Using software Defined networks we can prioritize the most important traffic by blocking other traffic on the network to make the customer experience better than before. We can set up rules and keep changing the programs based on the necessity and challenges the company faces in the most efficient way in a less costly manner. Due to the automation of configuration and management tasks due to SDN, we do not have to spend more on hardware which can help to reduce the cost to a great extent. Moreover, we do not have to worry about fixing and troubleshooting any hardware devices due to any damage to the devices as SDN reduces the need for hardware devices to a smaller scale. Hardware devices are very costly, and it is very hard to bear the cost after the damage.

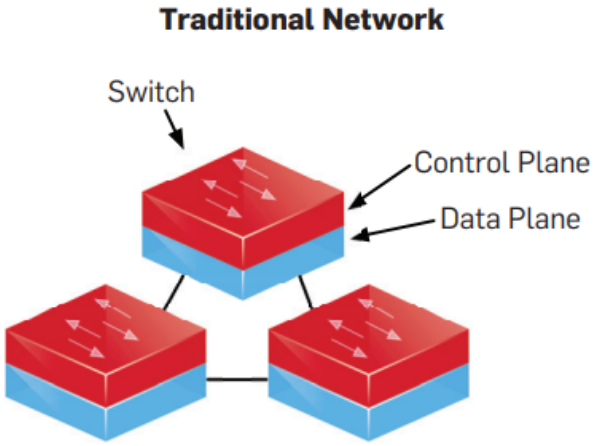

Figure 1a: Traditional Network

Source: Adapted From [3]

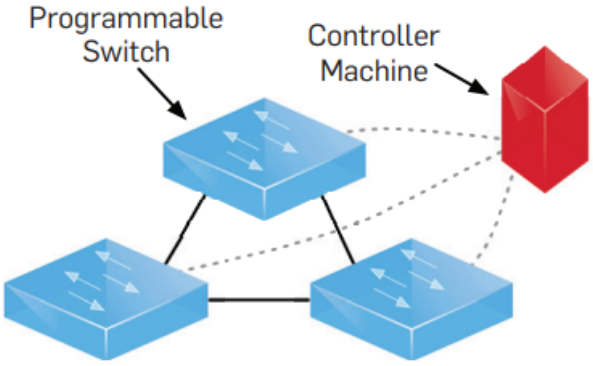

Figure 1b: Software-Defined Network

Source: Adapted From [3]

The architecture of the Software-defined network is much simpler than the traditional network as shown in Figure 1b. SDN uses a few controller machines for multiple programmable switches. As it is programmable it can be reused for various applications. The traditional network can cause

many issues because of the usage of multiple controllers. Because when the load of the controller is high it takes over the other which causes scalability issues (Casado et al, 2014). Moreover, forward is very tight on each as in figure 1a for the traditional network. The major disadvantage of SDN is that there will be a huge loss of vendors who sell hardware products and rapid use of SDN will reduce the need for hardware devices to a smaller scale which will affect the companies that run by selling hardware devices. As mentioned in the paper Software-defined network " many organizations simply do not have the time, expertise, or capital to invest in a completely new networking architecture"(Kirkpatrick, 2013). New networking architecture will require new skills and new labor which is time-consuming and many people will lose jobs due to a lack of skills on SDN. However, companies can save money once they start using SDN and will suffer fewer losses than before along with faster customer service.

III. OpenFlow

SDN technologies are connected with OpenFlow protocols. It is used for remote communications with network elements to find the path of the network packets in different network switches. Because of OpenFlow now network architectures are built very easily which interact with network devices. OpenFlow supports network instrumentation to measure the workload performance and detect any issues for effective network management (Siegel, 2007). It also notifies about any flows, topology, traffic statics.

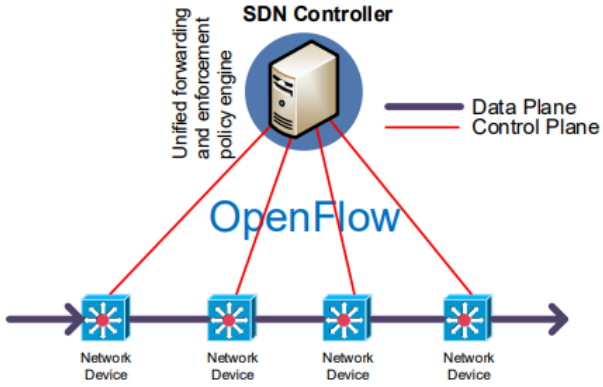

Figure 2: OpenFlow SDN Controller Source: Adapted From [7]

OpenFlow makes it possible to have a simple and clean performance of data, the lines in black on the figure, and the control plane, the separation of the network devices is essential to the success of the SDN. The control plane, the lines in red on the figure, enables data forwarding and flexible, reconfigurable networks as illustrated in Fig. 2.

OpenFlow is used to work with packet-switched networks.

IV. Software & Tools

Software-Defined Networking requires programming, automation, and troubleshooting skills. There are many programming languages used for networking such as Python, C++, and Java. Also, SDN will be required to configure, monitor, and manage networks dynamically which requires automation skills (Kim 2016). Moreover, troubleshooting is one of the most important skills for people working in the networking field. Troubleshooting for an SDN network will be different than troubleshooting a traditional network which requires new skills for SDN. Cisco and TestOut are offering networking courses and give certifications where people will give lab exams to test their excellency in the networking field.

Bibliography

[1] Arpit Gupta Georgia Institute of Technology, Gupta, A., Technology, G. I. of, University, L. V. P., Vanbever, L., University, P., Muhammad Shahbaz Georgia Institute of Technology, Shahbaz, M., Sean P. Donovan Georgia Institute of Technology, Donovan, S. P., Brandon Schlinker University of Southern California, Schlinker, B., California, U. of S., Nick Feamster Georgia Institute of Technology, Feamster, N., University, J. R. P., Rexford, J., Scott Shenker University of California Berkeley, Shenker, S., … Metrics, O. M. V. A. (2014, October 1). *SDX: A software defined internet exchange: ACM SIGCOMM Computer Communication Review: Vol 44, no 4*. ACM SIGCOMM Computer Communication Review. Retrieved March 15, 2022, from https://dl.acm.org/doi/10.1145/2740070.2626300

[2] A. LiMonCELLi, T. (2012). openFlow: A Radical new idea in networking. *Communications of the ACM*, *55*(8), 42-47.

[3] Casado, M., Foster, N., & Guha, A. (2014). Abstractions for software-defined networks. *Communications of the ACM*, *57*(10), 86-95.

[4] Gember, A., Grandl, R., Khalid, J., & Akella, A. (2013). Design and implementation of a framework for software-defined middlebox networking. *ACM SIGCOMM Computer Communication Review*, *43*(4), 467-468.

[5] Kirkpatrick, K. (2013). Software-defined networking. *Communications of the ACM*, *56*(9), 16-19.

[6] Gartner_Inc. (n.d.). *Network Management Instrumentation*. Gartner. Retrieved March 15, 2022,from https://www.gartner.com/en/documents/1405125/network-management-instrumentation

[7] *(PDF) openflow arbitrated programmable network channels ...* (n.d.). Retrieved March 16, 2022, from https://www.researchgate.net/publication/288890479_OpenFlow_ArbitratedProgrammable_Network_Channels_for_Managing_Quantum_Metadata

[8] Certify your sdn skills. Open Networking Foundation. (2017, June 8). Retrieved March 15, 2022, from https://opennetworking.org/news-and-events/blog/certify-your-sdn-skills/