ISSN 2348 – 7968

Open Flow-Based Software-Defined Network: A Survey

Abhishek Sanjay Patil[1] Prof. A.S. Gundale[2],

[1,2] Dept. of Electronics Engineering,
Walchand Institute of Technology, Solapur - 413 006,
Maharashtra, India

Abstract

The network in the current state is becoming very complex with more devices attached to it; these devices require more bandwidth and more network nodes. Due this the task of managing the network is becoming very difficult in terms of defining the connectivity and managing the traffic. There are added complexities due to geographical locations of the nodes and updating a repaired node with previous hardware configuration. To address this issue networking industry is looking for solution which can addresses the above issues in a smarter way. The step taken is to separate out management alias control plane and traffic alias data plane. In such network control plane can anywhere centralized in the network and manage the nodes from single window. This is referred as SOFTWARE-DEFINED NETWORK (SDN). SDN is an emerging network architecture where network control is decoupled from forwarding and is directly programmable. This migration of control, formerly tightly bound in individual network devices, into accessible computing devices enables the underlying infrastructure to be abstracted for applications and network services, which can treat the network as a logical or virtual entity. OpenFlow is considered the first Software-Defined Networking standard. It defines the open communications protocol in SDNs that enables the Controller to interact with the forwarding plane and make adjustments to the network, so it can better adapt to changing business requirements [1].

Keywords: *Software-Defined Network (SDN), OpenFlow*

1. Introduction

Traditional network architecture are ill-suited to meet the requirement of today’s enterprises. The current network architecture is shown in figure 1. There are so many applications are connected to the single monitor or network. Any changes to network are still manual process. When an application is rolled out, have to manually figure each and every device on the network recommend line interface. At the same time new setup application is unique experience from the network and manually figuration can accommodate the data. Not only does manual configuration at hours, sometime days to may task but we can’t make dynamic changes on a network to accommodate the new application of wires and that can strengths poor potentiality new application brings to business [1].

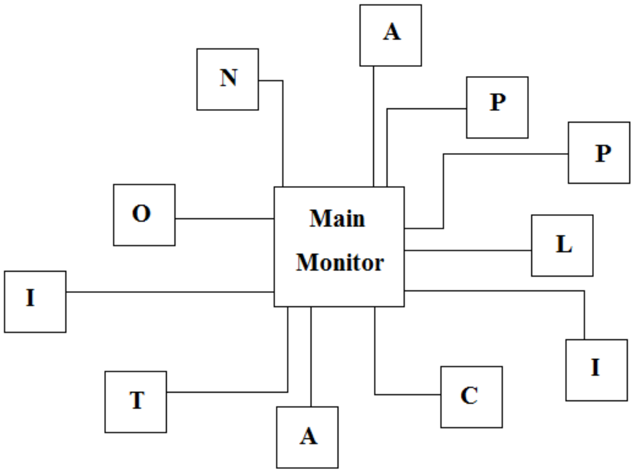

Fig. 1: Current Network Architecture

The Software-Defined Network (SDN) architecture decouples the data plane and control plane. In SDN network intelligence and states are logically centralized As a result, enterprises and carriers gain unprecedented programmability, automation, and network control, enabling them to build highly scalable, flexible networks that readily adapt to changing business needs. With SDN, enterprises and carriers gain vendor-independent control over the entire network from a single logical point, which greatly simplifies the network design and operation. SDN also greatly simplifies the network devices themselves, since they no longer need to understand and process thousands of protocol standards but merely accept instructions from the SDN controllers [2].

OpenFlow is considered the first software-defined networking (SDN) standard. It defines the open communications protocol in SDNs that enables the Controller to interact with the forwarding plane and make adjustments to the network, so it can better adapt to changing business requirements. OpenFlow protocol, which structures communication between the control and data planes of supported network devices. OpenFlow is the first standard interface designed specifically for SDN, providing high-performance, granular traffic control across multiple vendors’ network devices [3].

2. Necessity of New Network Architecture

As we know todays network is very complicated to the network and as number of devices increases in the network so requiring service also increasing, the requirements of network bandwidth and data is also increasing. As per the different surveys the number of devices by 2020 on internet will cross 20 billion, also in large industry there are so many devices are connected in single network, so changing or add device in the network is very much difficult in today's time scheduled. Managing of such network is very difficult in terms of defining the connectivity and managing the traffic. The things become more complex due to geographical locations as well. The other part of the problem is to integrate smoothly the newer node for expanding the capacity. Another reason is many conventional networks are hierarchical, built with tiers of Ethernet switches arranged in a tree structure. This design made sense when client-server computing was dominant, but such a static architecture is ill-suited to the dynamic computing and storage needs of today's enterprise data centers, campuses, and carrier environments [4].

2.1 Changing Traffic Patterns:

Within the network, data center and traffic patterns have changed significantly. The bulk of communication occurs between client and server, today's applications access having different databases and servers, creating a flurry of "east-west" machine-to-machine traffic before returning data to the end user device in the classic "north-south" traffic pattern. At the same time, users are changing network traffic patterns as they push for access to corporate content and applications from any type of device (including their own), connecting from anywhere, at any time. Finally, many enterprise data centers managers are contemplating a utility computing model, which might include a private cloud, public cloud, or some mix of both, resulting in additional traffic across the wide area network.

2.2 Rise of Cloud Service:

Enterprise business units now want the agility to access applications, infrastructure, and other IT resources on demand. To add to the complexity, IT's planning for cloud services must be done in an environment of increased security, compliance, and auditing requirements, along with business reorganizations, consolidations, and mergers that can change assumptions overnight.

2.3 More Bandwidth:

While performing in the todays network, required more bandwidth to transfer the data. As we discussed earlier in today's network lots of application is added in the network, having more data with the main monitor so while transferring all the data over the network require more bandwidth. While talk is going on between client and server, everyone wants direct communication between each other. The rise of mega datasets is fueling a constant demand for additional network capacity in the data center.

3. Limitation of Current Network Architecture

The today's requirement of application is virtually impossible with traditional network architecture. Many industrial are trying to quite the most from their network using device-level management tools and process which is manual. The demand for the mobility and bandwidth explodes are the same challenges for the carriers [5]. Today's users, carriers and enterprises could not use the existing network architecture; rather network designers are constrained by limitations of existing network, which includes:

3.1 Complexity that Leads to Stasis:

Networking topology to date has consisted largely of discrete sets of protocols connect hosts reliably over arbitrary distances, link speeds, and topologies. To meet business and technical requirement over the last few years, the industry has evolved networking protocols to deliver higher performance and reliability, broader connectivity, and more stringent security. Protocols tend to be defied in isolation, however, with each solving a specific problem and without the benefit of any fundamental abstractions. This has resulted in one of the primary limitations of today's networks: complexity. For example, to add or move any device, IT must touch multiple switches, routers, firewalls, Web authentication portals, etc. and update ACLs, VLANs, quality of services (QoS), and other protocol-based mechanisms using device-level management tools. In addition, network topology, vendor switch model, and software version all must be taken into account. Due to this complexity, today's networks are relatively static as IT seeks to minimize the risk of service disruption.

3.2 Inconsistent policies:

To implement a network-wide policy, IT may have to configure thousands of devices and mechanisms. For example, every time a new virtual machine is brought up, it can take hours, in some cases days, for IT to reconfigure ACLs across the entire network. The complexity of today's networks makes it very difficult for IT to apply a consistent set of access, security, QoS, and other policies to increasingly mobile users, which leaves the enterprise vulnerable to security breaches, noncompliance with regulations, and other negative consequences.

3.3 Vendor Dependence:

Carriers and enterprises seek to deploy new capabilities and services in rapid response to changing business needs or user demands. However, their ability to respond is hindered by vendors' equipment product cycles, which can range to three years or more. Lack of standard, open interfaces limits the ability of network operators to tailor the network to their individual environments. This mismatch between market requirements and network capabilities has brought the industry to a tipping point. In response, the industry has created the Software-Defied Networking (SDN) architecture and is developing associated standards.

4. Software-Defined Network

Software Defied Networking (SDN) is an emerging network architecture where network control is decoupled from forwarding and is directly programmable. This migration of control, formerly tightly bound in individual network devices, into accessible computing devices enables the underlying infrastructure to be abstracted for applications and network services, which can treat the network as a logical or virtual entity [6].
Figure 2 is the logical view of SDN. Network intelligence is (logically) centralized in software-based SDN controllers, which maintain a global view of the network. With SDN, enterprises and carriers gain vendor-independent control over the entire network from a single logical point, which greatly simplifies the network design and operation [4]. SDN also greatly simplifies the network devices themselves, since they no longer need to understand and process thousands of protocol standards but merely accept instructions from the SDN controllers.

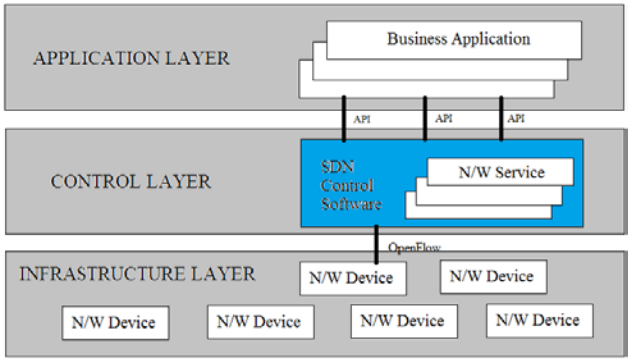

Fig. 2: Software-Defined Network Architecture

By centralizing network state in the control layer, SDN gives network managers the flexibility to configure, manage, secure, and optimize network resources via dynamic, automated SDN programs. Moreover, they can write these programs themselves. SDN architectures support a set of APIs that make it possible to implement common network services, including routing, security, multicast, access control, quality of service, bandwidth management, processor and storage optimization, traffic engineering and custom tailored to meet business objectives, example SDN architecture makes it easy to define and enforce consistent policies across both wired and wireless connections on a campus.

SDN makes it possible to manage the entire network through intelligent orchestration and provisioning systems. The Open Networking Foundation is studying open APIs to promote multi-vendor management, which opens the door for on-demand resource allocation, self-service provisioning, truly virtualized networking, and secure cloud services. Thus, with open APIs between the SDN control and applications layers, business applications can operate on an abstraction of the network, leveraging network services and capabilities without being tied to the details of their implementation. SDN makes the network not so much "application-aware" as "application-customized" and applications not so much "network-aware" as "network-capability-aware". As a result, computing, storage, and network resources can be optimized.

5. Openflow

OpenFlow is the first standard communications interface defied between the forwarding layers and controls of an SDN architecture. OpenFlow allows direct access to the forwarding plane of network devices such as switches and routers, both physical and virtual based. It is the absence of an open interface to the forwarding plane that has led to the characterization of today’s networking devices as monolithic, closed, and mainframe-like. No other standard protocol can’t do what OpenFlow can do, and a protocol like OpenFlow is needed to move network control out of the networking switches to logically centralized control software [7]. The thing is OpenFlow protocol is implemented on both side network infrastructure device and SDN control software. The OpenFlow protocol is a key enabler for software-defied networks and currently is the only standardized SDN protocol that allows direct manipulation of the forwarding plane of network devices. While initially applied to Ethernet-based networks, OpenFlow switching can extend to a much broader set of use cases [8]. OpenFlow-based SDNs can be deployed on existing networks, both physical and virtual. Network devices can support OpenFlow-based forwarding as well as traditional forwarding, which makes it very easy for enterprises and carriers to progressively introduce OpenFlow-based SDN technologies, even in multi-vendor network environments. So here also have to go through the benefits of OpenFlow based SDN network.

5.1 Centralized Control of Multi-Vendor Environment:

OpenFlow enabled network device from vendors, virtual switches and routers can be controlled by SDN control software. So because of this now no need to manage individual group of vendors, industry or IT can use SDN based management tools to quickly deploy, configure and update device across the entire network.

5.2 Reduced Complexity through Automation:

OpenFlow based SDN offers management framework and flexible network which makes it possible to develop tools that automate many management takes that are done manually today. In addition, with SDN, cloud-based applications can be managed through intelligent orchestration and provisioning systems, further reducing operational overhead while increasing business agility.

5.3 Increased Network Reliability and Security:

An OpenFlow-based SDN architecture eliminates the need to individually configure network devices each time an end point, service, or application is added or moved, or a policy changes, which reduces the likelihood of network failures due to configuration or policy inconsistencies.

5.4 More Granular Network Control:

OpenFlow‘s flow-based control model allows IT to apply policies at a very granular level, including the session, user, device, and application levels, in a highly abstracted, automated fashion. This control enables cloud operators to support multi tenancy while maintaining traffic isolation, security, and elastic resource management when customers share the same infrastructure [4].

6. Conclusions

In today’s network as user mobility, server virtualization, IT as a Service, and the need rapidly to respond to changing business conditions place significant demands on the network demands that today’s conventional network architectures can’t handle. Software-Defied Networking provides a new, dynamic network architecture that transforms traditional network backbones into rich service-delivery platforms. OpenFlow based SDN architecture allowing the network to become as programmable and manageable at scale as the computer infrastructure that it increasing similarly. To create network virtualization, enabling IT staff manage their applications, servers, networks and common tool set the SDN approach fosters. An SDN going to improve network manageability, scalability and agility. It transform today’s static networks into flexible, programmable platform with the intelligence to allocate resources dynamically, highly automated, dynamic virtual support and security. With all this advantages, SDN is the new typical networks.

Acknowledgments

We would like to take this opportunity to thank one and all who have provided their valuable advice, without their guidance this work would not have been a success, we have to thank who have helped us directly or indirectly since they have given us more than just guidance.
Our profound thanks to **Dr. S R Gengaje**, Head of the Department, Department of Electronics Engineering, Walchand Institute of Technology, Solapur, for his invaluable advice and constant encouragement to complete this work in a successful manner.
I would like to convey our sincear thanks to management and supportive staff of Walchand Institute of Technology, Solapur, for encouraging us to come up with this paper work

References

[1] Wikipedia- www.google.co.in
[2]Marby Tyson, “A Security Enforcement Kernel for OpenFlow Networks”HotSDN’12,August 13, 2012, Helsinki, Finland.

[3]E. Al-Shaer and S. Al-Haj. FlowChecker: "Configuration Analysis and Verification of Federated Openflow Infrastructures." In proceedings of the 3rd ACM SafeConfig Workshop, 2008.

[4]Software-Defined Networking: The New Norm for Networks (White paper- CISCO)

[5] S. Sezer, S. Scott-hayward, P.K Fraser, D. Lake, J. Finnegan, N. Vijoen, M. Miller, and N. Rao, "Are we ready for SDN? Implementation challenges for software-defined networks." Communications Magazine, IEEE Vol. 51 (7), 2013.

[6] OpenFlow Switch Speci_cation, Version 1.2 (Wire Protocol 0x03). http://tinyurl.com/84kelcj.

[7] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," ACM SIGCOMM Comput. Commun. Rev. **38**(2), 69–74 (2008).

[8] Kim, P. Sharma, J. Lee, S. Banerjee, J. Tourrilhes, S. Lee, and P. Yalagandula, "Automated and Scalable QoS Control for Network Convergence," Proceedings of the INM/WREN'10, Apr. 2010