Software-Defined Networking: Abstraction of the Network Control from the Hardware

Courage Osasumwen Oghogho[1]
B.C.A. Student,
Department of Computer Science & Engineering,
Ganga Institute of Technology and Management

Narender Narwal[2]
Assistant Professor,
Department of Computer Science & Engineering,
Ganga Institute of Technology and Management

Software Defined Networking (SDN) is enabling organizations to accelerate application deployment and delivery, dramatically reducing IT costs through policy-enabled workflow automation. SDN technology enables cloud architectures by delivering automated, on-demand application delivery and mobility at scale. SDN enhances the benefits of data center virtualization, increasing resource flexibility and utilization and reducing infrastructure costs and overhead. Since the advent of OpenFlow, researchers have been working to improve and of course facilitating the abstraction of network control plane from the Networking devices. This paper takes a lot at the inside-out of SDN, taking OpenFlow into effective use. This paper also details the architecture of SDN and OpenFlow, implementation, cost effectiveness and why organization should take advance of this cutting-edge technology to enhance productivity in their business while promising short and long cost worthiness and easy maintainability.

I. INTRODUCTION

Software Defined Networking (SDN) is allowing organizations to fast-track application deployment and delivery, intensely reducing IT costs through policy-enabled workflow automation. SDN technology enables cloud architectures by delivering automated, on-demand application delivery and mobility at scale. SDN augments the benefits of data center virtualization, increasing resource flexibility and utilization and reducing infrastructure costs and overhead.

SDN accomplishes these business objectives by linking the management of network and application services into centralized, extensible orchestration platforms that can automate the provisioning and configuration of the entire infrastructure. Common centralized IT policies bring together distinct IT groups and workflows. The result is a modern infrastructure that can deliver innovative applications and services in minutes, rather than days or weeks required in the past.

SDN delivers speed and agility when deploying new applications and business services. Flexibility, policy, and programmability are the hallmarks of SDN solutions, with a platform capable of handling the most demanding networking needs of today and tomorrow.

The concept of OpenFlow protocol forms the basis for Software Defined Networking. OpenFlow (OF) networks distinguish themselves from legacy network infrastructures by dramatically rethinking the relationship between the data and control planes of the network device. OpenFlow embraces the paradigm of highly programmable switch infrastructures, enabling software to compute an optimal flow routing decision on demand. For modern networks, which must increasingly deal with host virtualization and dynamic application migration, OpenFlow may offer the agility needed to handle dynamic network orchestration beyond that which traditional networks can achieve.

For an OpenFlow switch, the data plane is made programmable, where flows are dynamically specified within a flow table. The flow table contains a set of flow rules, which specify how the data plane should process all active network flows. In short, Open Flow's flow rules provide the basic instructions that govern how to forward, modify, or drop each packet that traverses the OF-enabled switch. The switch's control plane is simplified to support the OpenFlow protocol, which allows the switch to communicate statistics and new flow requests to an external OpenFlow network controller. In return, it receives flow rules that extend its flow table rule set.

Software-defined networking (SDN) is an architecture purporting to be dynamic, manageable, cost-effective, and adaptable, seeking to be suitable for the high-bandwidth, dynamic nature of today's applications. SDN architectures decouple network control and forwarding functions, enabling network control to become directly programmable and the underlying infrastructure to be abstracted from applications and network services.

II. BACKGROUND AND MOTIVATION

We have lost our way. Networking devices we use today are ramparted with highly sophisticated functionalities. As a result, the price of these devices is on a constant rise. We therefore need a network design that allows routers and switches to focus on data path. To do this, we are abstracting the intelligence and the entire control plane of our network devices to a separate software system, forming a centralized system to management. We intent to build a network design that allow centralized management of data. This initiative dramatically reduces cost of implementation, maintenance and even network up gradation.

III. DEFINITION OF TERMS

SDN: The physical separation of the network control plane from the forwarding plane, and where a control plane controls several devices.

OpenFlow Protocol: OpenFlow is a communications protocol that gives access to the forwarding plane of a network switch or router over the network.

Virtualization: virtualization refers to the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, storage devices, and computer network resources.

Orchestration: Orchestration is the automated arrangement, coordination, and management of computer systems, middleware, and services.

IV. BASIC SDN MODEL

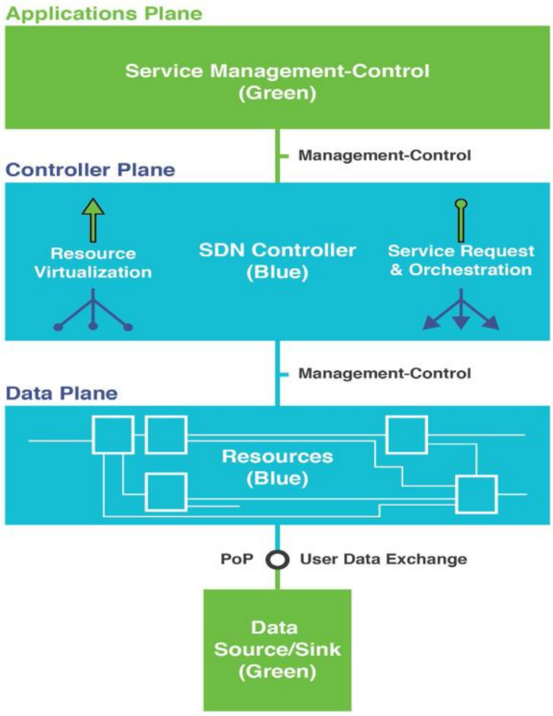
SDN is modeled as a set of client-server relationships with the SDN controller at its core.

Service Customer

• Manage-control network services via SDN Controller
• Send/receive data using network resources

Network/Service Provider

• Map customer service intent onto resources
• Resources include forwarding, processing, and storage domains
• Recursively map services and resources to scale or to span multiple administrative domains
• Virtualize resources to views for customers
• Orchestrate resources needed for service fulfillment

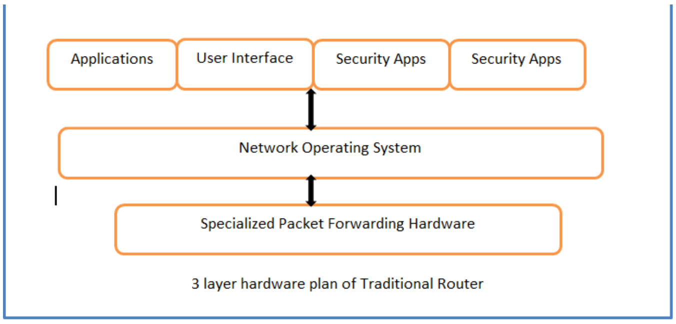


.”

Fig. 4.1 Basic view of SDN layers

V. TRADITIONAL NETWORKING

Traditional network devices are very sophisticated devices. They are made up of specialized packet forwarding hardware on which a sophisticated Operating System run. These Operating Systems are dedicated OSs perform all the intelligent decision of the devices. These generalizations of the devices make them a super expensive material to purchase. Furthermore, the cost of acquiring networking devices is on a dramatic rise. Atypical Cisco Router (networking device) used for internetworking would cost not less than USD 23,000.


Applications
User Interface
Security Apps
Security Apps
Network Operating System
Specialized Packet Forwarding Hardware
3 layer hardware plan of Traditional Router

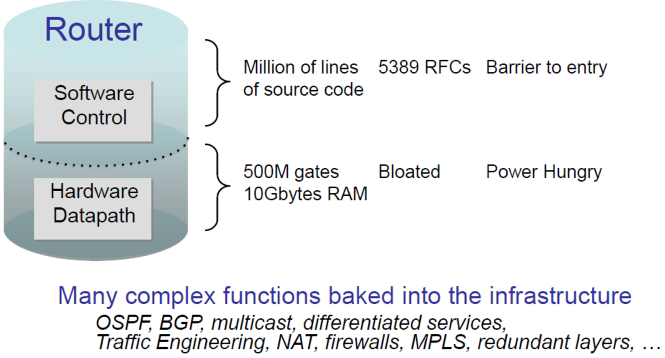

Router
Software Control
Hardware Datapath
Million of lines of source code
5389 RFCs
Barrier to entry
500M gates 10Gbytes RAM
Bloated
Power Hungry
Many complex functions baked into the infrastructure
OSPF, BGP, multicast, differentiated services, Traffic Engineering, NAT, firewalls, MPLS, redundant layers, …

Fig. 5.1 and 5.2 showing traditional router overview

VI. THE ARCHITECTURE OF SOFTWARE DEFINED NETWORK

Software-Defined Networking (SDN) is an emerging architecture that is dynamic, manageable, cost-effective, and adaptable, making it ideal for the high-bandwidth, dynamic nature of today's applications. This architecture decouples the network control and forwarding functions enabling the network control to become directly programmable and the underlying infrastructure to be abstracted for applications and network services. The OpenFlow protocol is a foundational element for building SDN solutions. The SDN architecture is:

Directly programmable: Network control is directly programmable because it is decoupled from forwarding functions.

Agile: Abstracting control from forwarding lets administrators dynamically adjust network-wide traffic flow to meet changing needs.

Centralized Management: Network intelligence is (logically) centralized in software-based SDN controllers that maintain a global view of the networks, which appears to applications and policy engines as a single, logical switch.

Programmatically Configured: SDN lets network managers configure, manage, secure, and optimize network resources very quickly via dynamic automated SDN programs, which they can write themselves, because the programs do not depend on proprietary software.

Open standards-based and vendor-neutral: When implemented through open standards, SDN simplifies network design and operation because instructions are provided by SDN controllers instead of multiple, vendor-specific devices and protocols.

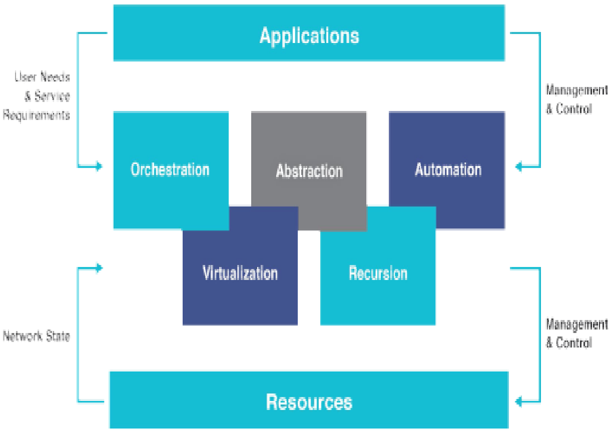


Fig. 6.1 Abstract view of SDN

The purpose of SDN is to reduce cost and improve user experience by automating the full range of network services, from end user to network element.

Principles that promote this include the decoupling of control from traffic processing and forwarding, centralization of control, and the ability of customers

and applications to interact directly with network control.

6.1) *The Central Entity in the SDN Architecture is the SDN Controller*

The essence of the SDN controller is its real-time feedback control capability, expressed in the orchestration and virtualization functions.

The architecture usually portrays client and server as existing in separate business domains, illustrated with separate colors.

Virtualization: is the process of abstracting, partitioning and aggregating underlying resources into virtual resources, each of which is dedicated to some particular client.

Orchestration: is the process of using the SDN controller's resources to simultaneously satisfy service demands from all of its clients according to an optimization policy.

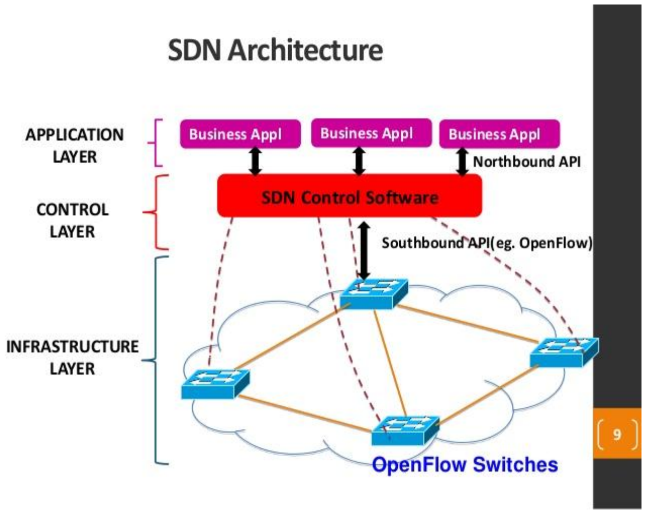


Fig. 6.2 SDN Architecture

6.2 The Open Flow Protocol

Open Flow is an open standard that has gained tremendous interest in the last few years within the network community. It is an embodiment of the software defined networking paradigm, in which higher-level flow routing decisions are derived from a control layer that, unlike classic network switch implementations, is separated from the data handling layer. The central attraction to this paradigm is that by decoupling the control logic from the closed and proprietary implementations of traditional network switch infrastructure, researchers can more easily design and distribute innovative flow handling and network control algorithms. Indeed, we also believe that OpenFlow can, in time, prove to be one of the more impactful technologies to drive a variety of innovations in network security. Open-Flow could offer a dramatic simplification to the way we design and integrate complex network security applications into large networks. However, to date there remains a stark paucity of compelling OpenFlow security applications.

"OpenFlow is the first standard communications interface defined between the controls and forwarding layers of an SDN architecture. OpenFlow allows direct access to and manipulation of the forwarding plane of network devices such as switches and routers, both physical and virtual (hypervisor-based).

OpenFlow-based SDN technologies enable IT to address the high-bandwidth, dynamic natures of today's applications, adapt the network to ever-changing business needs, and significantly reduce operations and management complexity." Open Networking Foundation ONF.

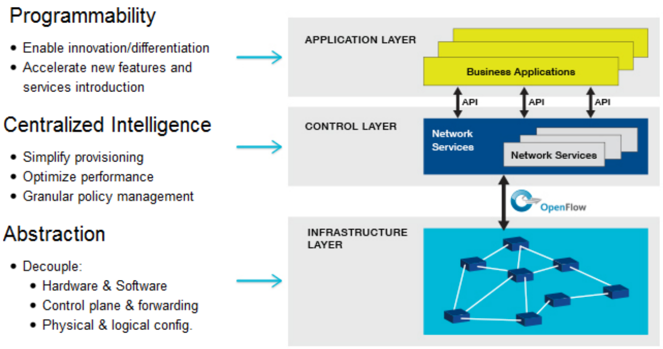


Fig. 6.3 The Open Flow protocol

VII. MIGRATING TO SDN AND WHY

Some companies are hesitant to deploy new technologies until they have seen what they can do in real-world situations. Software-defined networking is no different. And while at first glance the technology may seem like a daunting task to implement, the advantages of SDN, namely flexibility and long-term cost savings, are what make it an attractive option.

The biggest task is making management realize the inherent advantages of SDN, especially as standards continue to develop. Have management imagine the ability to turn new policies into network configuration at the push of a button, with no additional engineering time needed. How much cost savings would the company realize in a network of any appreciable size? Granted, for much smaller businesses with a single closet that makes up their total network, the advantages of SDN doesn't make much sense, yet.

In the future, however, once the technology matures and comes at a lower cost of ownership, smaller companies will be willing to jump in. Unfortunately, there is also the life cycle of the current equipment to take into consideration. Some companies are perfectly happy to use their equipment for seven to 10 years until it is fully depreciated. They see no reason to jump into new technology until they get what they consider their money's worth out of their legacy network equipment.

What's necessary is for you, the engineer, to take a good honest look at your network and see where it could benefit from an SDN deployment. Make a case, based on cost analysis, functionality and needs. Once compiled, you can make an honest self-determination of whether your company will gain the advantages of SDN. The keyword here is honest. If you just want to make the jump to SDN because it's the latest and greatest, you may want to reassess your position. If you can honestly say your company will save time and money with an SDN deployment, take your case to management and present it. While it won't always work, having an analysis and a plan can help move your case from fantasy to reality.

VIII. ADVANTAGES OF SDN

Cost Reduction: First, SDN does not require a huge investment. There are even a few SDN products that are free. And while you'll need to pay a license fee for some SDN solutions such as VMware's NSX, there are a few that ship with the operating system itself, including Microsoft's Hyper-V Network Virtualization.

And since SDN supports Layer 1 through Layer 3 networking models, there's no need to buy expensive networking devices. In other words, the use of SDN in a production environment can help reduce the costs involved in purchasing expensive hardware.

Overhead Reduction: In a physical environment, the isolation for the customer workloads requires configuring VLANs on separate networking devices, including routers, switches, etc. Since most of the networking is done at the SDN, it is easy for service providers to isolate the customer virtual machines from other customers by using various isolation methods available in the SDN.

Physical vs. Virtual Networking Management: Physical environments necessitate collaboration among different teams to get a task done. For example, if you require some modification at a physical networking device, it would often take a considerable amount of time and teamwork in most organizations before the task can be accomplished.

SDN provides you the ability to control the virtual and physical networking by using a central management tool. A virtual administrator can process the necessary changes without needing to collaborate with different teams.

Managing Virtual Packet Forwarding: SDN can help you forward the virtual packets to a software or physical device running on the network. For example, if a virtual machine needs to access the internet, it becomes easy for virtual administrators to provide the necessary configuration to the virtual machine with minimal effort.

Reduced Downtime: Since SDN helps in virtualizing most of the physical networking devices, it becomes easy to perform an upgrade for one piece rather than needing to do it for several devices. SDN also supports snapshotting the configuration, which helps you quickly recover from any failures caused by the upgrades.

Isolation and Traffic Control: Cloud service providers can benefit from centralizing the networking control using a central management tool. At the same time, SDN provides several isolation mechanisms such as configuring ACLs and firewalls at the virtual machine NIC level. You can also define the traffic rules using the SDN management console, which helps in providing full control over the network traffic.

Extensibility: Since SDN is software-based, it is easy to use SDN API references for vendors to extend the capabilities of an SDN solution by developing applications to control the behavior of networking traffic.

Central Networking Management Tool: SDN can deliver all your networking needs in one product, enabling you to control every piece of an organization's network using a central management tool.

Network administrators often find it difficult to manage a physical router's configuration, and it quickly becomes time consuming and tedious when more than one physical router needs to be managed. SDN simplifies the management of physical routers by providing the management APIs in the SDN console.

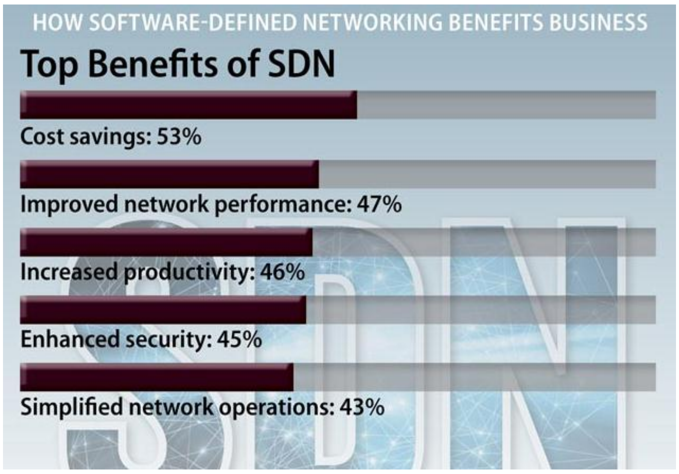


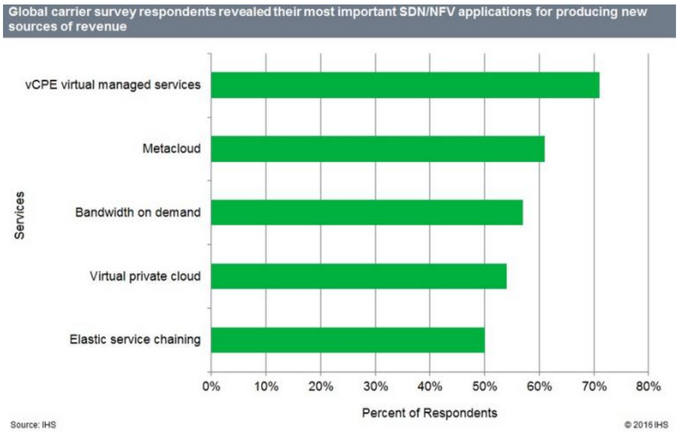
Fig. 7.1 (Benefits of SDN)

IX. THE FUTURE OF SDN

Since the emergence of the OpenFlow in Stanford University, many researchers have been working on the advancement of this topic. The Open Network Foundation is a key player of SDN. Network vendors like Cisco has referred to SDN as the future of networking. Since the SDN paradigm is a hybrid of Networking and software programming, the concept has attracted professionals and researchers from both sides.

It is clear that SDN has come to stay; it is indeed the future of networking. What Open flow is today is a just a little of millions of possibilities and potential of the concept. Virtualization is a key feature of SDN and forms the basis of abstraction of control from the hardware devices.

APPENDIX



APPENDIX 1.1 (SDN DEPLOYMENT)

X. ACKNOWLEDGMENT

My thanks goes to god almighty for his grace and of course provisions. I humbly appreciate my family for their support and guidance. While it is the responsibility of every parent to support his/her children, mine are outstandingly special. My warm appreciation goes to my elder sister, ms. Victory oghogho. I cannot thank you enough. You are indeed a blessing. I also want you to know that i am and will ever be indebted. I also extend my ward regards to mrs. Anita dhankar dagar for her motivation. Dr. Ahman agarwal is also an icon of support for my stay and focus here in ganga institute of technology.

XI. REFERENCES

[1] Seugwon Shin, Phillip Porras, Vinod Yegneswaran, Martin Fong, on certern information regarding OpenFlow technology: “FRESCO: Modular Composable Security Services for Software-Defined Networks”.

[2] Open Network Foundation (ONF) on basis of SDN

[3] Cisco Corporation new letters