A SURVEY ON SOFTWARE DEFINED NETWORKS

Karthiga S[1] , Abinaya C[2] , Murugavalli E[3] , Uma Maheswari G[4]
[1, 3, 4] Department of ECE, Thiagarajar college of Engineering, Madurai, India
[2] Department of Information Technology, Thiagarajar college of Engineering, Madurai, India

Contact author: karthiga227@gmail.com , cabinaya03@gmail.com

Abstract*—SDN is a type of computer networking which allows administrators to initialize, control, and change network via open interfaces. SDN separates the control plane and data plane. Data plane is used for processing and delivery of packets based on the state in routers and endpoints. Control plane determines how and where packets are forwarded. In SDN, networking devices need to separate the infrastructure layer from the control layer and the application layer. When a single controller is given then it's capable of controlling multiple devices. It is cost effective, and suitable for high bandwidth. In these paper we discussed about SDN's Architecture applications and research areas including hot topics such as Information Centric Networks, Cloud and data centres, multimedia, wireless and mobile networks over SDN are reviewed.

***Keywords*— Control Plane, Data Plane, Programmable Network, OpenFlow, Software Defined Network.**

I.

II. INTRODUCTION

In recent years use of mobile phones has been increasing day by day which results in emerging technologies such as cloud computing and virtualization etc… The rise of Big Data in data centers arise the need for high network capacity and network scaling. To support these needs, network devices become more complex. It would be hard and time consuming for administrators to configure individual devices due to even little changes in network, such as adding or omitting a device. They should reconfigure many multivendor switches and routers which may cause inconsistency and errors. The Idea of programmable networks was introduced to meet these challenges and facilitate network evolution. As a result, Software Defined Networking (SDN) is a new paradigm, which revolutionized traditional network architecture. SDN effectively separates the control plane from data plane. Data plane and network devices, makes control plane to be directly programmable and manageable in a centralized manner. The SDN architecture provides programmability, flexibility and reliability over networks. Network operators can implement their own protocols, rules and policies with common programming languages. They can achieve flexible control over network services such as routing, traffic engineering, QOS and security. Network can adapt itself depends on users' requirements. Network management and configurations can be automated through the centralized controller and standard open API, making the network scale easily. By using SDN, administrators are able to add features to control plane without changing data plane or enhance devices in data plane without changing control plane. Decoupling control plane from infrastructure is also important because it reduces costs and inconvenience of testing new ideas and strategies in network or deploying new architectures.

In this paper, we present difference between Traditional networking and SDN, SDN architecture, applications and challenges to give a broader view for those who are interested in this area.

III. TRADITIONAL NETWORKING

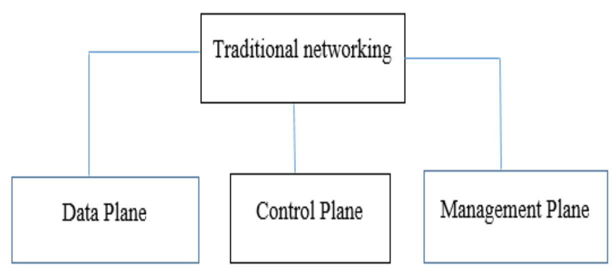


Fig.1 Types of plane in Traditional networking

Computer networks can be divided in three planes of functionality: the data, control and management planes as shown in fig.1 the data plane corresponds to the networking devices, which are responsible for forwarding data. The control plane represents the protocols used to populate the forwarding tables of the data plane elements. The management plane includes the software services such as remotely monitor and configure the control functionality. Network policy is defined in the management plane, the control plane enforces the policy, and the data plane executes it by forwarding data.

In traditional networks, the control and data planes are tightly coupled, embedded in the same networking devices, and the whole structure is highly decentralized. This was considered important for the design of the Internet in the early days. Traditional networks are rigid, and complex to manage and control Network misconfigurations and related errors are extremely common in today's networks. For instance, more than 1000 configuration errors have been observed in routers from a single misconfigured device may result much undesired network behavior. To overcome all these issues we are going for SDN.

IV. Early programmable networks

A. Open Signaling

The Open Signaling (OPENSIG) working group began in 1995 with a series of workshops dedicated to "making ATM, Internet and mobile networks more open, extensible, and programmable" They believed that a separation between the communication hardware and control software was necessary but challenging to realize; this is mainly due to vertically integrated switches and routers, whose closed nature made the rapid deployment of new network services and environments impossible. The core of their proposal was to provide access to the network hardware via open, programmable network interfaces; this would allow the deployment of new services through a distributed programming environment by these ideas, which led to the specification of the General Switch Management Protocol a general purpose protocol to control a label switch[1][2]. GSMP allows a controller to establish and release connections across the switch, add and delete leaves on a multicast connection, manage switch ports, request configuration information, request and delete reservation of switch resources, and request statistics.

B. Active Networking

In 1990s, the Active Networking the idea of a network infrastructure that would be programmable for customized services. There were two main approaches being considered, namely: (1) user-programmable switches, with in band data transfer and out-of-band management channels; and (2) capsules, which were program fragments that could be carried in user messages; program fragments would then be interpreted and executed by routers [1][4] Active Networking never gathered critical mass and transferred to widespread use and industry deployment, mainly due to practical security and performance concerns

C. DCAN

In the mid-1990s is the Devolved Control of ATM Networks (DCAN). The aim of this project was to design and develop the necessary infrastructure for scalable control and management of ATM networks. Control and management functions of the many devices should be decoupled from the devices themselves and delegated to external entities dedicated to that purpose, which is basically the concept behind SDNs. DCAN assumes a minimalist protocol between the manager and the network, in the lines of what happens today in proposals such as OpenFlow. Still in the lines of SDNs and the proposed decoupling of control and data plane over ATM networks, amongst others, in the work proposed in multiple heterogeneous control architectures are allowed to run simultaneously over single physical ATM network by partitioning the resources of that switch between those controllers.

D. 4D Project

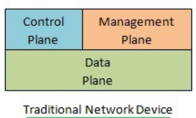
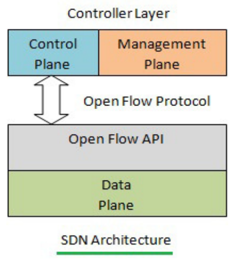
Starting in 2004, the 4D project advocated a separation between the routing decision logic and the protocols governing the interaction between network elements. It proposed giving the "decision" plane a global view of the network, serviced by a "dissemination" and "discovery" plane, for control of a "data" plane for forwarding traffic. These ideas provided direct inspiration for later works such as NOX which proposed an "operating system for networks" in the context of an OpenFlow-enabled network.

E. NETCONF

In 2006, NETCONF was proposed as a management protocol for modifying the configuration of network devices. The protocol allowed network devices to expose an API through which extensible configuration data could be sent and retrieved. The NETCONF protocol accomplishes the goal of simplifying device (re)configuration and acts as a building block for management, there is no separation between data and control planes

F. Ethane

The immediate predecessor to OpenFlow was the SANE / Ethane project which, in 2006, defined a new architecture for enterprise networks. Ethane's focus was on using a centralized controller to manage policy and security in a network. Similar to SDN, Ethane employed two components: a controller to decide if a packet should be forwarded, and an Ethane switch consisting of a flow table secure channel to the controller [5]. Ethane laid the foundation for what would become Software-Defined Networking.

Traditional Networking	**Software Defined Networking**
They are Static and inflexible networks. They are not useful for new business ventures. They possess little agility and flexibility.	They are programmable networks during deployment time as well as at later stage based on change in the requirements.
	
They are Hardware appliances.	They are configured using open software.
They have distributed control plane	They have logically centralized control plane.
They use custom ASICs and FPGAs.	They use merchant silicon.
They work using protocols.	They use APIs to configure as per need.

V. Software Defined Networking

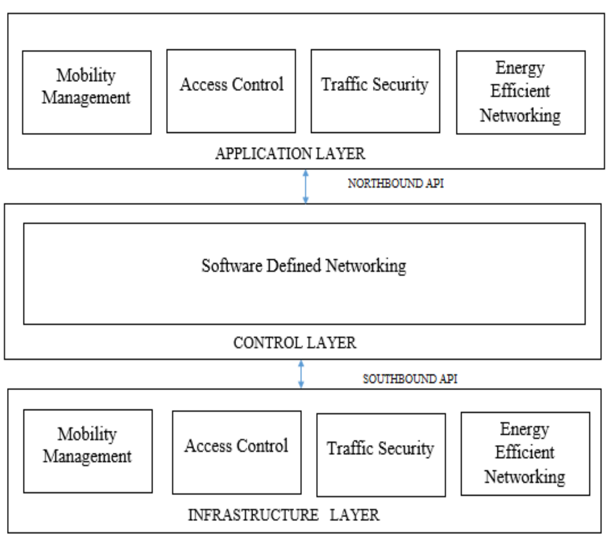


Fig.2 SDN Architecture

Software-Defined Networking is regarded as a technology that is capable of managing the entire network efficiently and transforming the complex network architecture into the simple and manageable one. SDN is regarded as the hardware independent next generation networking paradigm in which networking device from any vendors could be controlled through SDN [2][4]. SDN has decoupled application plane, data plane and control plane. It has two prime components: a) Controller b) switches**.** SDN controller is responsible for the management of entire network whereas networking switches are responsible for operating based on the instructions deployed through SDN controller

TABLE 1: Difference between SDN and Traditional Networking

VI. SDN Architecture

A. *Infrastructure layer*

Infrastructure layer in SDN comprises of network devices such as router, switch and access point. The main function of the data plane is forwarding the packets according to the assigned rules/policies [6].

B. *Control Layer*

Control layer consists of a controller which controls the overall SDN functions. This layer acts as a mediator for infrastructure layer and application layer [4]. The controller is responsible for managing the entire traffic flow and slowly takes decisions on routing, flow forwarding and packet dropping through programming .The controllers in the distributed environment communicate with each other through east-bound and west-bound interfaces. The control layer and the infrastructure layer communicate with each other through south-bound API such as OpenFlow, NetConf, etc.

C. *Application Layer:*

The Application layer is the foremost layer in the SDN. It is responsible for handling software related business and security applications. Network virtualization, Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), firewall implementation and mobility management are some examples of applications handled by this layer[8]. This layer communicates with the control layer using Application Control Plane Interface (ACPI) also called as northbound application interface

D. *Southbound interface*

In SDN, the most popular southbound interface is the OF protocol. OF enables communication between controller and the nodes in the network, so the controller discover network topology, get reports from the nodes, instruct and manage them as needed and implement requests relayed to flows via northbound APIs.

E. Northbound interface

Interfaces between SDN Applications and SDN Controllers are known as Controller Application Interaction or SDN Northbound Interfaces (NBIs). Northbound Interface of component can be introduced as an element that conceptualizes the lower level details of functions used by component. This interface is usually positioned at the top of the corresponding component, which is the basis of the "northbound interface. Despite southbound interface that is well defined in protocols such as ForCES and OF, no comprehensive standard is defined for northbound interface and they are more likely to be developed for particular SDN applications[6] One justification is that the southbound must enable hardware implementation, while northbound interface's definition completely in software. For various reasons controllers may need to communicate with each other, on the other side, network applications may require extraction of information about the underlying network policy aspect, then there should be a clearly defined interface

VII. Applications and Research Trends

In this section we introduce SDN applications and recent researches including Software defined ICN, Multimedia, network management, network virtualization, cloud and datacenter, security, wireless and mobile networks.

A. Software defined ICN

In recent years many researchers claimed that current internet architecture is not able to response the emerging and future need of users. Based on this claim, new architectures were introduced. Information centric network is one of these architectures [6][7]. In ICN, the information name is unique and independent of locations, applications, storages and distribution and network primitives are done based on the names. To retrieve named information, various transmission techniques are introduced, including name-based routing, name-based resolution, and etc. To support these techniques and exploit the advantages of ICN, dramatic changes to the network devices deployed in current Internet are needed, which leads to challenge of ICN implementation. A number of projects proposed implementing ICN over SDN. This leads to decreasing in implementation costs. It also enables innovation and optimization of network resources and functionalities.

B. Multimedia and QOS

Today's internet architecture is based on end-to-end communication control which enables best effort services [1]. This is valuable for data transmission but not for multimedia traffics. Multimedia applications such as video streaming, video on demand, video conferencing, WebTV and etc. require steady network resources and tolerate special amount of delay, jitter and error-rate.[10] Providing these QOS requirements needs to select optimum path among all paths available in the network They also need specialized software and hardware requirements for implementation. Software defined networks make it possible to select different paths for different traffic flows by use of different routing protocols (performing routing prioritization), based on flow's requirements and a centralized view of the network.

C. Network management

SDN architecture, made network management more flexible, control in the packet or flow granularity levels. Management in software defined network from a centralized logical point upon flow-table at controllers and using that flow-tables distributed in network (switches) cause a flexible network management . Network configuration because of A) high-level policies specifications in a distributed low-level configuration mutable network state is difficult frequently changes happening in a network, using a high level language for network configuration, fault and error recognition and troubleshooting Which is based on functional reactive programming (FRP) and High level policies could be translates into a set of rules and four control domain is in operators' hand: time, data usage and authentication status.

D. Network virtualization

Network virtualization is one of the important research areas of today's network that enable users to share resource and infrastructure. A system for managing virtual networks and sharing infrastructure based on layers [6]. The benefits of network virtualization consist of ease of configuration and flexibility of header fields that allows users to create different slice and different virtual network. Network virtualization without interfering between different slice so users are enable to modify the behavior of own virtual network. [9]

E. Cloud and data center

One area that SDN has been attended a lot is Cloud Services and data center. One of the main characteristics of cloud is that users gain the adequate resources based on requirement in real time[4]. Cloud management is the most important challenge that has always been and many solutions have been proposed for that. SDN is highly regarded as one of the newest solutions, which makes it possible to configure and manage cloud and data center easily[7]. Implementation is faster and easier to configure because of SDN centralized controller and abstraction management plan[7]. The other challenge of cloud system is maintain costs, that SDN due to centralized location management solve this problem well.

ISSUES	NETWORK VIRTUALIZATION ISSUES	SECURITY ISSUES	PLACEMENT ISSUES	SDN STACKS ISSUSES
PROMINENT REASONS	Virtualized Operating System, Virtual Network Programming	Centralization of Controller, No Interoperability between Standards, Flow Authentication	Depend on Topology of Network	Control Flow Saturate between Controller and Switches
SUGGESTED SOLUTION	Add extra hypervisor virtual security layer and firewall for this layer. Authentication and Access layer Security	Floodlight, FRESCO, FortNox	Controller may vary according to size of network	Include Authentication or digital signature mechanism for higher priority given to flow
AVAILABLE RESOURCES	Node, Link	Controller platform, NOX, POX, FLOODLIGHT, Jaxon, Trema, beacon, MUL, Node Flow		N/A
AFFECTED PARAMETER	Other Virtual Machines, difficult to manage large amount of virtual machines state with consistency	Malicious flow	Latency, response time	False flow rules
SOLUTION PROVIDER	Xen VMware	Nicira, Big switch networks		Veriflow
OPENFLOW ENABLE TOOL	Ns3, Mininet			

TABLE 2: SDN Issues

F. Security

Due to centralized architecture of SDN, it used to detect security problem. Supervision of SDN on whole network flow and monitoring behavior of users makes SDN possible to detect attack rapidly and prevent more damage and it also used for fake IP is easily detectable and system prevents large percentage of attacks such as stealthy scanning, worm propagation and etc.[8]

G. Wireless and mobile

SDN can be applied in wireless sensor network Generally, using SDN in WSNs provided the SDN benefits such as flexibility, easier management, optimized resource utilization, etc. The network controllers have the power to set policies to support several applications by utilizing sensor based software defined wireless network. Also this approach would permit using the same sensor nodes for several applications [5]. OF can be used for applying flexible control in Wireless Mesh Networks.[8] This approach benefits both features of Mesh networks and OF which are self-configuration and flexible forwarding, respectively. To apply concepts of abstraction to wireless ad hoc network of smartphone, software defined networking in ad hoc networks was developed. This Hybrid platform has been implemented on Android operating system.

SIMULATOR/ EMULATOR	Open Source	Language	Platform
Mininet	Yes	Python	Ubuntu or Fedora
Ns-3	Yes	C++, Python	Linux, Mac
EsitNet	No	-	Linux

TABLE 3: Simulators and Emulators

VIII. CONCLUSION

This paper addresses brief overview about SDN. We began our discussion with difference between networking and software defined networking. Thereafter we discussed about the architecture in three layers: control, infrastructure and application layers and also about early programmable network such as SOFTNET, SANE, ForCes, etc. and their issues. We also described different tools for testing SDN. We compared Mininet, EstiNet and NS3 as SDN simulators and emulators. Finally, we discussed the applications and future research trends such as software defined ICN, virtualization, wireless and mobile networks, cloud and datacenters, multimedia over SDN and the works done on security of SDN.

Reference

[1] Kapil Dhamecha, Bhushan Trivedi, "SDN Issues – A Survey" International Journal of Computer Applications (0975 – 8887) Volume 73– No.18

[2] Gurbani, Vijay K., et al. "Abstracting network state in Software Defined Networks (SDN) for rendezvous services." Communications (ICC), 2012 IEEE International Conference on. IEEE, 2012.

[3] Shiva Rowshanrad, Sahar Namvarasl; Vajihe Abdi, Maryam Hajizadeh , Manijeh Keshtgary "A survey on SDN, the future of networking" Journal of Advanced Computer Science & Technology, 3 (2) (2014) 232-248 ©Science Publishing Corporation

[4] Wenfeng Xia, Yonggang "A Survey on Software-Defined Networking" IEEE COMMUNICATION SURVEYS & TUTORIALS

[5] S. Scott-Hayward, G. O'Callaghan, and S. Sezer, "SDN Security: A Survey," in Future Networks and Services (SDN4FNS), 2013 IEEE SDN for. IEEE, 2013, pp. 1–7

[6] Fei Hu, Qi Hao∗, Ke Bao "A Survey on Software-Defined Network (SDN) and OpenFlow: From Concept to Implementation" IEEE Communications Surveys & Tutorials

[7] Yuefeng Wang Ibrahim Matta “SDN Management Layer: Design Requirements and Future Direction” 2014 IEEE 22nd International Conference on Network Protocols

[8] S. Scott-Hayward, S. Natarajan, and S. Sezer, “A survey of security in software defined networks,” IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 623–654, 2015.

[9] Alsmadi and D. Xu, “Security of software defined networks: A survey,” computers & security, vol. 53, pp. 79–108, 2015.

[10] W. Li, W. Meng, and L. F. Kwok, “A survey on OpenFlow-based Software Defined Networks: Security challenges and counter measures,” Journal of Network and Computer Applications, vol. 68, pp. 126–139, 2016.