ISSN: 0975-766X
CODEN: IJPTFI
Available Online through *Research Article*
www.ijptonline.com

COMPREHENSIVE RESEARCH AND ANALYSIS OF SDN

Darshan KN, Harsh Patel, Dr.Dinakaran.M
SITE School, VIT, Vellore, Tamilnadu.

Received on 25-10-2016 *Accepted on 02-11-2016*

Abstract:

Innovations in the networks are totally concentrating on improving the bandwidth rather than building a new design for the networks. Traditional networking devices such as routers, switches uses a centralised policy for each and every component present in the network, so every component in that network has to abide by the same policy and rules of the centralised standard protocols. So, Software Defined Networks (SDN) is the one stop solution to the above mentioned problems and for other such kind of issues related to the traditional networks which will be mentioned in the later sections of this paper. This paper will revolve around the scope and future of the networking using SDN and some of the current applications of SDN.

Keywords: SDN, SDN Controller, Virtualisation, Orchestration.

I. Introduction

SDN is one of the trending topics in networking which does designing of the networks, does network building and does the management of networks too. Some industrial giants of networking have implemented SDN, but it is still confidential and rest are researching to implement the same in the near future. Even though SDN is in research phase and has so many limitations, but features offered by it will help in overcoming the problems associated with traditional networks. This paper will give a comprehensive tour to the SDN with its applications and limitations associated with it and some proposed solutions to it. Traditional networking devices like routers will have data plane and control plane at the same site of the device as given in the figure 1 below.

Fig 1.

Data plane involves actual movement of data and carries actual traffic whereas control plane builds routing tables and does routing by following the security policies present in it. Problem with this is, each and every device connected to this router will have to follow same policies present in the control plane and an addition of new device to the network under this router requires manual configuring of the device. But SDN will have this control plane separated from the data plane to an off-site device as shown in below figure 2, so that each device can have its own policies and centralised control plane of the entire network will make addition of new devices easier.

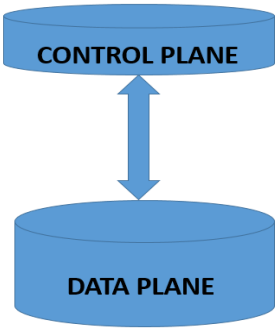

Fig 2.

II. Basic Architecture

Basic architecture of SDN will have Northbound Application Programming Interfaces (API's), Southbound API's and a SDN controller as its components.

Figure 3 shows the basic architecture of SDN.

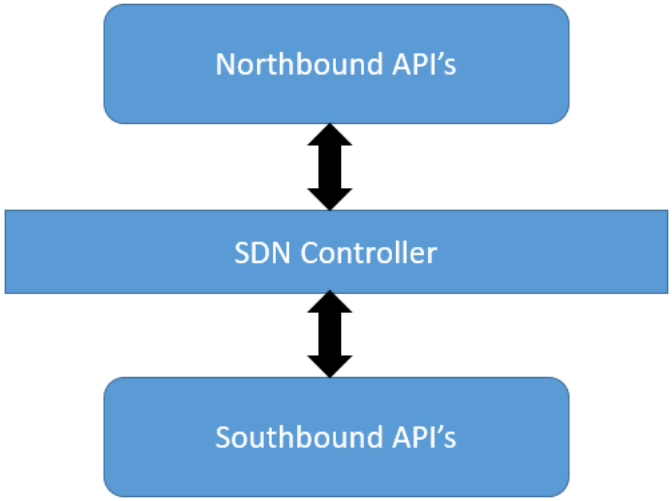

Fig 3.

Northbound API's actually interacts with the user applications and communicates with the central part called SDN controller to program the network as required by the applications. Northbound API's gives the actual requirements of the applications and asks SDN controller to do programming accordingly.

Southbound API's does the actually programming of the networking devices by communicating with the SDN Controller and pushes information onto the networking devices by following the instructions given by the northbound API's indirectly. "OpenFlow" is one such southbound API's currently available and it is also the protocol for the SDN. Intercommunications as shown in the figure 3 is actually done using protocols like OpenFlow.

SDN Controller is the main component of SDN and is also called as Middleware which has the software running in it to do programming of the network as informed by the applications using northbound API's.

SDN controller will have all the information regarding all the devices of the network. In traditional networks, networking devices makes routing decisions whereas, SDN controller does the routing by making appropriate decisions in software defined networks. OpenDaylight, Ryu are some of the open source SDN controllers which are available free of cost. Floodlight and Rosemary are some of the examples of research driven SDN controllers.

The separated part, control plane as shown in figure 2 will be in SDN Controller which will have every information associated with every node in it. Actual abstraction of the underlying topology as requested by the applications interacting through northbound API's are done by the so called middleware.

Every node in the network will communicate with SDN controller using protocols like OpenFlow. All forwarding decisions done by routers of traditional networks is done by SDN controller in software defined networks.

OpenDaylight controller is one such SDN controller. SDN controller works both reactively and proactively depending on the response of the packets. It can update, delete and can add flow entries with or without having any pre-defined rules. OpenFlow helps in establishing a secure connection between SDN controller and all network devices and defines control messages, reads the current state of all the devices associated with it and does installation of forwarding instructions. OpenFlow is currently being used in prominent datacentres, even though it was actually developed for researchers to run their experimental protocols. OpenFlow switch has 3 parts Flow table, Secure channel, and OpenFlow protocol. SDN topic is currently in trend because of so many advanced features bundled together in this. Next section will provide an overview of what all concepts that are bundled together to make this trendy.

III. Pillars of SDN

Main goal of SDN is to design, maintain and manage flexible, dynamic, large, heterogeneous networks by overcoming the issues associated with the traditional networks. To make this goal into reality there are some important concepts which are considered as main pillars of SDN, they are

1. Centralised control,
2. Programmability,
3. Orchestration and
4. Virtualisation.

Traditional networks will have same policy being implemented for each and every device connected in the network which mainly obstructs the main purpose of designing a network, so centralised control will have control of every device which in turn allows creating a more flexible innovative networks.

Centralised control is done by the middleware called SDN controller in software defined networks.

Programmability is one more concept that helps in realising the goals of the SDN,

Configuration of networking devices by networking engineers in traditional networks uses command line interfaces or GUI's which are cumbersome to learn and work.

One more issue is addition of a new node into the traditional network requires admins to configure manually for inclusion of new devices which is tedious, error-prone and time consuming too.

Management tools are available to overcome all these issues, but set of management tools are needed to do this. But, SDN will provide set of API's of many programming languages as an array of tools to the networking engineers.

Atuna Networks and Alcatel-Lucent's Nuage networks are some examples of orchestration tool providers which allows automation in software defined networks and reduces human errors and achieves all the goals successfully.

Virtualisation of networks allows abstraction of the underlying networking topology and makes the utilisation of the networking devices to its full potential and also allows physical networks to be shared among different virtual networks.

In SDN, network virtualisation is done with the help of overlays like Virtual Extensible LAN (VXLAN), Network Virtualisation using GRE (NVGRE), Stateless Transport Tunnelling (STT) in combination with OpenFlow.

Above mentioned concepts are considered as pillars because of its comprehensive support given to the SDN. Next section IV will enlist some of the important applications of software defined networks .

IV. Applications of SDN

- It is used in places where management and maintenance of large and heterogeneous networks like data centres.
- Networks which are spread across the globe requiring centralised control.

- Securing devices which are connected to the internet such airplanes, fridges, ovens, washing machines.
- In providing security to the sessions created in SSl/TLS.
- SDN in combination with wireless sensor networks.
- SDN based secure multicast communication.
- Carrier networks with concepts of SDN.
- SDN to secure LTE.
- Routing firewalls can be integrated with concepts of SDN.
- IoT for smart cities.

Even though after providing such a massive bundled features, SDN has got its own limitations which will be seen in the next section. Because of the applications that SDN can support and features that it can provide makes SDN a trendy topic currently and will be the future of networking.

V. Limitations of SDN

- SDN is still a research oriented concept: SDN is being implemented in many organisations which require flexible management of networks, and handling big data centres and so on, but still it is in research phase and not implemented by small organisations because of many reasons like cost, no standards available and so on.
- No standards are available at present for complete implementation of SDN: Each and every networking protocols or even networking devices are built based upon some standards provided by international organisations. But, no such kind of standards are available and no international organisation has provided any standards for SDN. Still now, OpenFlow is one standard which is an open source group approved standard.
- Protocols currently available like OpenFlow will not support multivendor network devices: Every network will have many devices which may be manufactured by multiple companies or devices of any single company, devices of single company in an entire big organisation is a rare case, because entire organisation cannot survive on devices manufactured by a single brand. Since SDN being supporting hybrid/ heterogeneous networks doesn't have a protocol which supports multivendor network devices. This is one of the main limitations of SDN.
- No backward compatibility support by OpenFlow switches: Network should have one backbone network in case the main network fails, that backbone should take care of the main operations, but in SDN, even though it has an overlay network to support virtualisation, but, switches of OpenFlow currently available doesn't

support backward compatibility of devices meaning once the switches of OpenFlow has been installed in a big organisations, it supports only OpenFlow protocol for communications and can't and will support other protocols to use in those devices.

- Costlier to migrate into software defined networks: because of the above given limitations of SDN, migrating to software defined networks presently is not a good idea, but it will be a boon for big datacentres which requires a flexible, new network design and not at all bothered about costs can easily switch over to SDN
- Centralised control of devices in SDN will lead to denial of service problem if central controller is hacked: Main crux of SDN is to have centralised control of entire network at a single device at a single device like SDN controller. But, the main problem in doing so is if anyone gets unauthorised access to this central device may result in collapse of entire network or the entire network can be changed into something bad which nobody knows. To overcome this problem, researchers have proposed so many solutions as discussed in the next section VI. Next section will provide a brief of proposed solutions given by many researchers in various research papers and so on.

VI. Proposed Solutions

Some of the solutions are proposed by researchers as given in the reference section papers.

To overcome the problems associated with the compromising of centralised control, multiple SDN controllers can be used in the same network and synchronisation among multiple SDN controllers has to be done. Interactions between multiple SDN controllers can be of two kinds, one is, both the controllers can have equal interactions of the switches and second kind is to have master- slave kind of interactions, in this second kind of interaction can have equal interactions of the first kind also.

Even though migration costs are more, it helps in cutting down costs on different aspects. Cost is not at all a problem for bigger organisations who can afford SDN to have all its features in their organisations networks even though SDN has limitations.

SDN being implemented on top of overlay networks will provide backup for actual core business of network, if SDN fails.

VII. Comparison of SDN with traditional networks

Normal traditional networks will have each and everything distributed across the network and management of the network devices configuration requires changing/reconfiguring at so many different distributed places across the

networks and doing automation of the same requires building up of an interface for each and every device in the network.

Whereas in SDN, control plane and data plane are disintegrated as discussed above and data plane never interacts with the control plane for each and every packet transfer, instead it does communicate with it only when certain major changes are done in routing.

In traditional networks, if peak threshold is reached then ultimatum is network down. Whereas in SDN, one the pillars discussed above such as virtualisation allows shift in the peak traffic to other networks just by changing the routing.

In Normal networks, if something goes wrong like path failure, device failure, then network administrator has to reconfigure the failure components, but in SDN, networks will heal itself by detecting the failures and switching over to other devices or other networks also if needed.

Normal networks are mere hardware with some basic software in it and in SDN, open source software is used for configuring the networks.

Merchant chip is being used in SDN and ASIC's and FPGA are being used in normal networks.

Normal networks are not inflexible since it doesn't support programmability, virtualisation and other pillars of SDN, whereas the main goal of SDN is to build a flexible network.

After listing many differences between normal traditional networks and SDN, both has its own limitations and advantages over the other and both has its own functionality and applications. Normal traditional networks are time-tested gives this an upper hand over the SDN.

VIII. Conclusion

This paper has given all the basics of SDN and overview of it regarding supporting concepts, advantages and limitations. Implementation of SDN is not under the scope of this literature survey.

Concluding part is, some more time is required for its complete implementation across the different areas of work and for common public usage. This is a great area of research for proposing a standard for communications such as a protocol, more of multivendor supporting SDN devices and much more.

SDN will be the future, so vendors have to start building SDN supportable equipments in order to make SDN available for smaller organisations and to the normal customers.

References

1. Olivier flauzac and Carlos gonz´alez and Abdelhak hachani and Florent nolot, "SDN based architecture for IoT", 2015 29th International Conference on Advanced Information Networking and Applications Workshops.
2. Alireza Ranjbar, Miika Komu, Patrik Salmela, Tuomas Aura," An SDN-Based Approach to Enhance the End-to-End Security", 2016 IEEE/IFIP Network Operations and Management Symposium.
3. Sandra Scott-Hayward, "Design and deployment of secure, robust, and resilient SDN Controllers", IEEE, 2015.
4. Younghwan Choio, Yunchul Choi, and Yong-Geun Hong," Study on Coupling of Software-Defined Networking and Wireless Sensor Networks", ICUFN 2016.
5. http://searchsdn.techtarget.com/tip/SDN-basics-Understanding-centralized-control-and-programmability
6. https://www.sdxcentral.com/sdn/?utm_source=pink_ball&utm_medium=link&utm_campaign=links&utm_content=SDN
7. Shaibal Chakrabarty, Daniel W. Engels,"IoT Architecture for Smart Cities", IEEE, 2016.
8. Karamjeet Kaur,Sukhveer Kaur, Vipin Gupta," SDN based Routing Firewall", IEEE, 2016.
9. Jianfeng ZOU, Guochu SHOU, Zhigang GUO, Yihong HU," Design and Implementation Of Secure Multicast Based On Sdn", IEEE, 2013.
10. Madhusanka Liyanage, ljaz Ahmad, Mika Ylianttila," Leveraging LTE Security with SDN and NFV", IEEE, 2015.
11. OpenFlow. http://en.wikipedia.org/wiki/OpenFlow.
12. OpenFlow Specification. [Online]. Available: https://www. opennetworking.org/sdn-resources/openflow, accessed Jul. 23, 2015.
13. Steven S. W. Lee, Kuang-Yi Li and Ming-Shu Wu," Design and Implementation of a GPON-Based Virtual OpenFlow-Enabled SDN Switch", IEEE, 2016.
14. Adel Zaalouk, Ronald Marx," OrchSec: An Orchestrator-Based Architecture For Enhancing Network-Security Using Network Monitoring And SDN Control Functions", IEEE, 2014.