International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 5 Issue: 2 103 – 106

Various Component handling services for Software Defined Networking (SDN)

Deepak Kumar*
Department of Computer Science
Himachal Pradesh University, Summer Hill, Shimla

Manu Sood
Department of Computer Science
Himachal Pradesh University Summer Hill, Shimla

Abstract: Software Defined Networking (SDN) provides us the platform for the easy management of various components, which was not possible in traditional networks**.** SDN is flexible and scalable because of the separation between control and data plane. OpenFlow protocol acts as the interface between both planes. The concept is that we can create control logic and implement it in network elements of the data plane through centralized control. The controller and the policy need to be updated regularly to meet the requirement of the network. In this paper we have discussed the need of security approaches that can detect any kind of fault or abnormal behavior with in the network and to manage the packet flow metrics. Also how can we distribute the global policy across each of the element in data plane so that the global policy becomes local to each networking elements.

Keywords: *SDN, OpenFlow, Policy, security.*

*****

I). INTRODUCTION

SDN [1] is a networking approach to build robust, flexible and scalable networks. It is the start-up for futuristic networks. It provides the platform for the innovation, and has become the hot trend in research for researchers; its uniqueness is in the control plane separation from the data plane. In SDN control plane is not the part of the network elements like switches, routers and access points etc., and is the part of the network edge; it allows various developers to develop software that acts as the interface to the network elements regardless of the type and vendors of the network elements. SDN also offers the possibilities to application coders to build any app that operators want to run on the top of the network operating systems. There are wide varieties of controllers available most of the controllers are open source such as POX [2], NOX [3], Beacon [4], Onix [5]and Floodlight[6] etc. The choice of using the controller depends upon the interest of users, e.g. if users is interested on working on java platform the floodlight controller is the suitable choice; because floodlight controller based on java platform. The controller is the part of the control plane. It is the place where whole network intelligence lies and controls the functionality of the networking elements in the Infrastructure layer.

Whenever the controller finalizes the routing rule and policy it implements in infrastructure layer through programmatic control.Control plane makes the decisions like, what to do with the packets, where to forward the packet and whether to drop the packet or not. With this functionality the management and monitoring of the network system become very easier. The OpenFlow [7] protocol helps to communicate with both the data and control plane. OpenFlow is developed by Open Networking foundation and it becomes the standard component of the SDN which is capable of handling multiple heterogeneous devices through centralized controller. SDN has also leads its deployment in wide area networks. OpenFlow protocol also simplifies applications and security functions integrations within the network system. It is founded in recent research that OpenFlow helps in deployment of various security policies within the network system, such as performing ever-changing load balancing, and providing trace back and intrusion detection mechanism etc. SDN and OpenFlow enable the networking operators to offers custom security services in order to meet the requirements of individual network host.

In section II section we are discussing the policy changeover in SDN, in III section we are discussing the security based management for SDN, in section IV we are discussing mechanism to monitor packet flow metrics, in section V we are discussing the reactive logic for SDN, in section VI we finally concluding and in the section VII we are discussing the future scope.

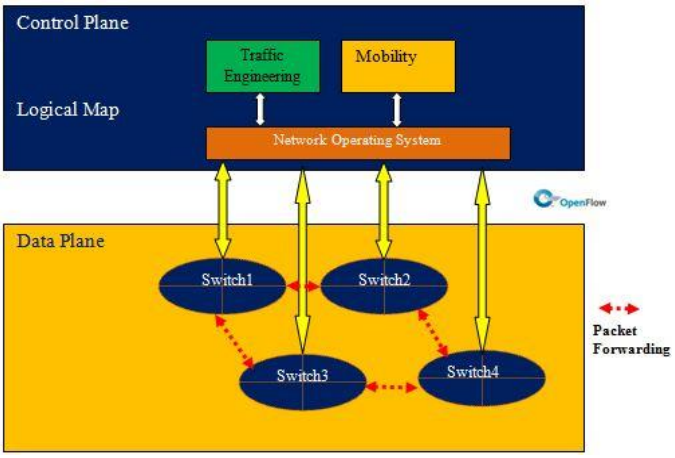

Figure1: Communication between control and data plane through OpenFlow

II). POLICY CHANGEOVER IN SDN

Policies are very important part for any networks that needs to be followed. In traditional networks it was difficult to manage

the various network resources, but SDN help us in managing the whole network through controller. Global policy changeover can be operated by move operation, merge operation or by splitting rules among multiple switches, while maintaining the correct behavior of a network. By using some tool, the policy which is written for individual switch can also be distributed to other set of switches, as shown in **Figure 2** and the policies pre-distributed over multiple switches can be consolidated. This is important because of the following reasons [8]**:**

Resource utilization:The policies which are large enough to fit in to any of the individual switch can be distributed across multiple switches.
Easy programming:For programmers Policies for a single switch are easier for programmers to write.

.

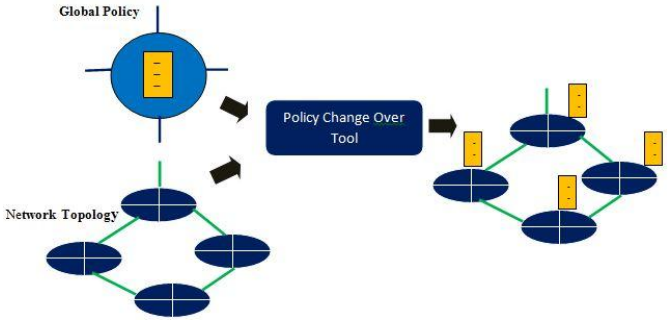

Figure 2: Global policy implementation in network topology through policy change over tool

Easily Deployable: Policies runs over one network topology can also run over other network topology.
Analysis: A policy which is distributed over different switches can be changed into single set of rules; in order to check the policy satisfies invariants.

The main aim of the policy change over tool is to allow the controller to change policies automatically across each of the hardware element working in the data plane.

III). SECURITY BASED MANAGEMENT FOR SDN

SDNOpenFlow protocol provides the flexibility to manage the today's dynamic networks. Control plane defined the rules, instructions and policies that specify the packet forwarding, to modify the packet or to drop the packet, for those packets which are incoming to the OpenFlow enabled switches [9]. Controller has the ability to communicate with multiple switches at the same time. The current research is on to provide the security to the whole network system, since whole network system is dependent on the centralized controller, which is the single point of failure. So there is need of applications and algorithms that can detect abnormal behavior (fault across any network element, performance issues, intrusion detection, to reroute the flow of packet with in the network) and inform the controller for taking the suitable actions.

All this is possible, if we have an efficient controller that can recover as soon as possible, capable of handling many faults at the same time. The network elements working in data plane must be supportive of the controller in the control plane. Network elements like switches router must be intelligent like controller so that, in any case, if controller fails then switches must be capable of taking self-based decisions like where to forward the packet, to drop the packet or to change the network topology. Also all the network elements must be intelligent enough to gather statistics about the each of the incoming and outgoing packet. By doing so we can minimize the risk of failure

IV). MECHANISM TO MONITOR PACKET FLOW METRICS

SDN provides the interface to implement various services like controlling, to monitoring and network management etc. It is considered to be as the major factor to implement network optimizing algorithms and Quality of Service (QoS) parameters. The accurate traffic monitoring is the key requirement for the network management in order to meet the traffic engineering [10] and QoS standards. Many of the resources are consumed by flow based measurements due to various monitoring requirements, where as other monitoring options leads to configuration management and deployment of hardware.

Internet Service Providers (ISP) proves more support to the network capacity in order to meet QoS necessities. TE requires the real time monitoring information in order to compute the most efficient routing based decisions. SDN's OpenFlow standard provides us the programming interface that allows the controllers to execute TE, so as OpenFlow provides interface that allows the controller to query for statistics and interject the packet with in the network. There is a need of mechanism that must be capable of providing TE controllers for online measurements and monitoring. In conventional networks many diff monitoring techniques are used. In traditional approach each measurement technique usually requires different software and hardware configuration which makes it very difficult and costly to implement it. However OpenFlow provides various methods.

- Active and Passive methods.
- Measurements across network and application layer.

Active methods: Measurements of the network can be categorized in to two groups, active and passive methods. In passive methods the network traffic measurement is done by

observation without the interjection of fake traffic. The benefit of using the passive method is that they do not generate overhead with in the network and therefore does not affect the performance of the network. Passive kind of measurements requires synchronization between the various observations, which creates difficulty in monitoring process.***Passive methods:*** based on the interjection of additional flow of packets with in the network and monitor their behavior. Additional flow of packet leads to the additional network load affecting the network.

Both of these methods are useful to monitor the flow of packet with in the network and to collect the statistics.

Measurements across network and application layer:In order to measure the network various measurements are performed on the different layers of OSI. The measurements of the application layer are preferred to accurately measure the application performance. The Internet Service Providers (ISP) also doesn't have access to the host devices, and for this reason the network layer for measurements is used. Networks layer measurement of the hardware devices like router, switches etc. to obtain the statistics.

V). REACTIVE LOGIC FOR SDN

The main working of the controller is to monitor the state of the network in a timely manner by interacting with the network elements such as switches and routers etc. to determine their availability and load. By reacting on measurements, controller can dynamically configure the switches to optimize the process of packet forwarding, to equalize the network load and to reduce the end to end delay. The SDNs control plane as implemented in OpenFlow protocol is based upon the concept of flow, where the flow is described in terms of the packet header fields.

SDN controller application selects the path that is to be followed by the packet flow by installing the entries in the flow tables of each networking device e.g. rules, policies and action (to forward, to delay or to drop the packet). The OpenFlow assigns two types of timeout to each incoming packet.

Idle timeout:It is the time after which the entry is removed if there is no new packet of flow available.

Hard timeout:This type of timeout tells us the absolute lite time of the each entry, after which each entry is removed.

The current scenario of SDN controller applications are primarily based on the proactive logic. Which require the network elements to accommodate the flow table entries which exceed the limit of their TCAMs (Ternary Content Addressable Memory). To provide the SDN networking elements with larger TCAMs is possible at the cost of operational and power consumption.

The make use of reactive logic [11], where the resources are allocated or free it depends upon the current network load. Also the Flow table entries used are of limited size so that can be easily updated, monitored corrected whenever needed.

VI). CONCLUSION

SDN has changed the way of thinking about networks. SDN networks has solves many of the issues those were in the traditional networks. What makes SDN suitable for current networking scenario; is the easy handling of various network resources through the intelligent controller. Despite of the several benefits of SDN; there are also some issues and those issues needs to be solve as it is hot research topic, so the chance of improvements are much more. We have discussed the methods to monitor flow metric so that we can monitor traffic flow and apply traffic engineering (TE) applications and to maintain the QoS of the network system. The reactive approach can be advantageous because of the limited capacity of the flow table and therefore it is easy to manage flow table entries and also it can be updated by controller through programmatic control. For any networking system security based management is the key-point for its efficient working and handling of services. SDN is beneficial for academics, industries and also for the ends –users.

VII). FUTURE SCOPE

We can use the application of the SDN in other related fields like cloud, data-Centre and in mobile networks. We need to develop various algorithms to detect the performance and security issues, by monitoring each networking element with in the network. Also we need to use security mechanism (firewalls, intrusion detection) in all three layer of the SDN so that each of the incoming packets should be filtered and it does not affect the other parts of the network. The advantage of having the firewall like security is that it will monitor the each of the application running over controller, if it finds any of the malicious application it will now allow that application to run over the controller. So this way we can make SDN much secure. SDN has many advantages and many more are yet to come. The future of SDN is bright; we can say that, because in the current networking scenario there is no better choice available than SDN. SDN is very huge topic, so chances of researches are much wider.

VIII). REFERENCES

[1] Software Defined Networks, Article available at [ONLINE]: http://www.opennetworking.org

[2] POX [online] Article available at link:http://www.noxrepo.org/pox/about-pox/

[3] N.Gude, T.Koponen, J.Pettit, B. Pfaff, M.Casado, N. Mckeown, S. Shenker, NOX: Towards an Operating System for Networks.SIGCOMM Comput.Commun. Rev.,38:105-110, July 2008.

[4] D. Erickson, The Beacon OpenFlow controller Proc. In HotSDN 2013.

[5] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Yuichiro Iwata, Hiroaki Inoue, Takayuki Hama, Scott Shenker, Onix: A Distributed Control Platform for Large Scale Production Networks. In OSDI 10, 2010

[6] Floodlight [online] Article available at link: http://floodlight.openflowhub.org

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

[8] N. Kang, J. Reich, J. Rexford, D. Walker. Policy Transformations in Software Defined Networks,SIGCOMM'12, August 13–17, 2012, Helsinki, Finland.

[9] A. Kamisinski, C. Fung. FlowMon: Detecting Malicious Switches in Software Defined Networks. SafeConfig'15, October 12, 2015, Denver, Colorado, USA.

[10] N.L.M.V. Adrichem,C. Doerr and F.A. Kuipers.OpenNetMon: Network Monitoring in OpenFlow Software Defined Networks, 978-1-4799-0913-1/14 copyright IEEE.

[11] M. Dusi, R. Bifulco, F. Gringoli, F. Schneider. Reactive Logic in Software Defined Networking: Measuring Flow Table Requirements, 978-1-4799-0959-9/14 copyright IEEE.