



Conspiring Open Source Technologies

Dr.Aruna Varanasi¹, S Manminder Kaur², Prof.T.Venkat Narayana Rao³

¹Professor and Head, , Department of Computer Science and Engineering

²Student, Department of Computer Science and Engineering,

³Professor , Department of Computer Science and Engineering

^{1,2,3}Sreenidhi Institute of Science and Technology, Yamnampet, Ghatkesar, R.R District , T.S, India.

Abstract: OpenDaylight is a new, ambitious and a pioneering project of the Linux Foundation which deals with the progress of the Linux kernel. Its main objective is to work with the Software-Defined Networking (SDN) that can take the virtualization to a next level by providing network architecture of virtualized network nodes. It is an approach towards computer networking, which is aimed at developing an improved way of networking among the computer devices. It is set to develop a framework that allows beneficial and robust networking, which will replace the current networking model with various virtualization technologies. OpenDaylight provides a platform that will link the traditional networking technologies with the future networking technologies. This paper states the importance of how the OpenDaylight can be collaborated with advance Software-defined Networking (SDN) that supports the emerging community led technologies and with its evolution.

Keywords: OpenDaylight, Software-defined network (SDN), virtualization of network function (NFV), OpenFlow.

I. INTRODUCTION

The struggle to keep up with the emerging business requirements and the need to reduce an increasing burden on the networks massivehigh-density scaleand cloud environments of multi-tenancy,they have indulged in providing a dynamic and explosive nature of the virtualized environments where multiple tenants are using the shared resources to drive their changing business needs.

In order to improve the efficiency, performance and security of the networks, new technologies like OpenDaylight, Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) have emerged to make the network agile and flexible by dissociating the forwarding plane control, in order to make the provision easy and to automate and orchestrate the services of the network. The virtualization of networks is attempting to adjust the network resources to meet the dynamic requirements of rich multi-tenant environments.

II. OVERVIEW

OpenDaylight is a synergic project by the Linux Foundation which was endowed in April, 2013 and its first release was disclosed on February 2014. This Foundation has begun the Open Platform for Network Function Virtualization (OPNFV) in September, 2014 which powered the OpenDaylight SDN controller for the project "BootStrap/Get Started!" which is aimed at sanctioning Virtualized Network Function (VNF) technologies [1]

[2][10]. This project has the calibre to exert the OpenDaylight as a classic open source SDN controller and a medial for assembling new and traditional networks. It is an open-source cloud computing platform that support in the expansion of networking technologies in the performance of the networking infrastructures [4][7].

OpenDaylight is established on Java and is highly applicable, expandable, commutable, multi-protocol controller infrastructure built for concurrent diversified multi-vendor networks. It is a cloud computing platform which is led by the community and supported by the industry. The framework consists of blueprints and code that promotes novelty which facilitates risk reduction and gives a translucent path to SDN. It maintains a platform that grants the users to write apps that easily works on various hardware and south bound protocols and also providing various products and services for the dealers. It provides a space for the vendors to unveil and contend by implementing best results for the users[2][4][7].

OpenDaylight project from the Linux Foundation, count on the framework which is collaborative and supervisory and which helps the architect to target for better results. It uses open measures to attain its objectives. Many big companies like Dell, Ericsson, Cisco systems and Brocade are aiding on this foundation in order to establish financial interpretation of their controllers to implement SDN and NFV [12]. In the meantime, other dealers are supporting OpenDaylight by introducing customization and economic support [2][4][10].

III. OPENDAYLIGHT RELEASES

A. INITIAL RELEASE

Hydrogen Release Plan

It is a concurrent release plan for OpenDaylight which was launched on December 9, 2013. It required planning for the project, mutuality in which the modules have desired specifications and quality to run their project [10].

B. PREVIOUS STABLE RELEASE

Helium Release Plan

It is a concurrent release plan which required code hygiene along with planning, modularity and quality. As part of this release the project expectations are to promote tests, packaging and documentation. It was initiated on April 14, 2014 [10].

C. CURRENT STABLE RELEASE

Lithium Release Plan

It is a simultaneous third release plan who's API's has been stated stable, provisional and tentative. In addition to the

previous stable release, the requirements of this plan were leadership and communication, testing, documentation, distribution and meeting deadlines [10].

D. THE NEXT RELEASE

Beryllium Release Plan

It is the fourth release plan which has notable differences from its former releases. This plan has stable and extended features such as Top-level and User-Facing feature, project integration and it performs three kinds of tests, Unit Test(UT), Integration Test (IT) and System Test (ST) [10].

IV. ACCESSION OF OPENDAYLIGHT

As OpenDaylight is established on java which is platform-independent, it can be installed on any operating system (OS). Most of the members use Linux. Before installing OpenDaylight, we need to be familiar with Linux OS and should find it easy to execute operations in the Bash shell.

The OpenDaylight platform contributes an outspread cluster of features which is used to implement broad variety of use cases. Before the installation, users should better understand the notable use cases and its tests which will help them to adapt the required features and modules.

V. SOFTWARE-DEFINED NETWORKING

Software-defined Networking is a novel architectural schema. At physical topology, it divides the system into control plane and data plane. Control plane has the authority to divert the traffic from the primary systems and Data plane has the authority to lead the traffic from these systems to their respective destination. The inventors alleged that this will formulate networking[3][4][8].

SDN is feasible in facilitating network shifts, elaborating network utilization efficiently and chastening the administration of networks. It is an advance computer networking that grants the network executives to govern the network services and features through conception of lower level utility [1][3][4]. It enables the users to easily program the network layers. SDN can provide users to exaggerate agile networks and modify their network resources, modernization of service; quicken time-to-market services, abstracting business perception and permit effective, work-driven virtual networks by implementing programmability[4][6][8].

As there is an enormous amount of data in the networking industry and due to the emerging cloud computing platform, the industry of virtualized networks is at a turning point [4]. This permits the users an exceptional demand over cloud resources. Virtualization and programmability of networks lead to the disintegration of the industry, making its acceptance complex. This made the vendors visualize that it cannot abide.

For the future of transmission and communication of information across the world, SDN exhibits both compelling threats and limitless opportunities. Hence companies are coming forward to adopt modern architectures with collaborative open source software development [10].

VI. SDN ARCHITECTURE

In SDN, all the complex operations along with routing, policy deposition, naming and guaranteed analysis is implemented by a central controller and the switches directly supervise the flow tables whose access is occupied by the controller. The controller decides the data streams that appear in the data plane of SDN. A consistent protocol and API is used as an interchange between the controller and the switches[3][4][8].

The SDN architecture as shown in Fig.1 is exceptionally extensible. At distinct protocol layers, it can work with various categories of switches. Layer 2 i.e. Ethernet switches, Layer 3 i.e. Internet routers, Layer 4 i.e. routing and application layer switching are resolved by switches and SDN controllers.

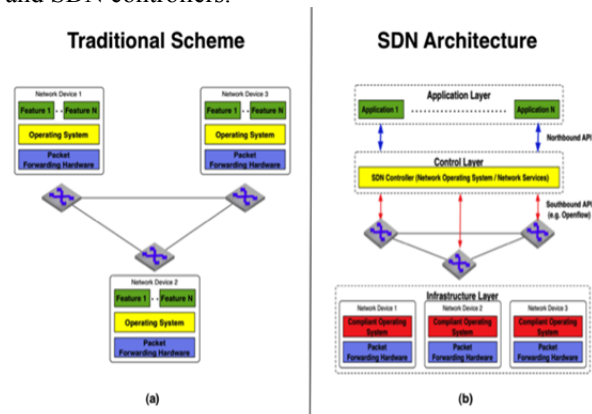


Fig.1. architecture of SDN

The operations performed by a switch are,

1. The switch facilitates the flow to be joined to the switch flow table which will be decided by the controller after the switch encloses and sends the first packet to an SDN controller.
2. The arriving packets are lead out of a relevant port by the switch depending on the flow table which may comprise of priority information.
3. As imposed by the controller, the packets can be dropped by the switch in a peculiar movement either permanently or temporarily [6].

In massive enterprise networks, to supervise all network resources by a single controller, will show undesirable results. In such networks, the operator partitions the total network into diverse non-extending SDN domains. These domains provide high scalability, personalized privacy policies and malleable incremental deployment [6].

VII. RESULT OF COLLABORATION

As an open source software is limitless. It produces the products and services which has a leading edge in every market. Originally, OpenDaylight will build a framework and a dynamic code to approve the developers and the users to configure the networks virtually, to dynamically balance the traffic and to easily adapt networking policies [9].

In addition to this, there will be applications that come up to aid contemporary use cases for constructive data centres, observing the conduct of application and even security. At present, SDN is backed by open standards and these standards are powered by the Open daylight, by which it is overwhelmed for the collaboration of business leaders like Open Networking Foundation (ONF). One of the illustrations of ONF is OpenFlow, which is an open SDN protocol refined by open community. The modern SDN Controller architecture situated on Services Abstraction Layer (SAL) approach was granted by the OpenDaylight (ODL) SDN Controller which also supports other protocols excluding OpenFlow. As OpenDaylight is open source, it is efficient to approve these standards [3].

As a result of this collaboration, the significant aspect is to provide Open Source Software which basically facilitates the customers to personalize and comprehend their benefits, although it is in favour of modernization which is feasible only from an open source community that is synergic.

VIII. OPENFLOW

In SDN, to disclose control plane with the data plane, we require some mechanism. One such system is OpenFlow. It is a protocol which is handled by the Open Networking Foundation that grants a server to instruct where to send the data packets [5]. Most dominant networking dealers like Cisco, Brocade, Extreme Networks, Alcatel-Lucent/Nuage Networks, Big Switch, Arista, Juniper and Hewlett-Packard are incorporated as its supporters[10][11].

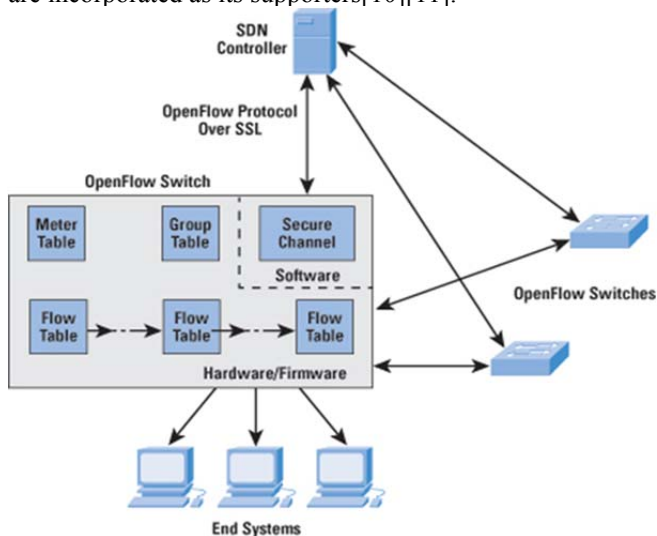


Fig.2. architecture of openflow

In a traditional network, all the switches have recovery software that instructs it what to do and the data path and control path appear on the same device [5][9]. The data path is detached from the control path with the OpenFlow switch. The data path segment is located on the switch itself and the packet-moving judgements are streamlined with an independent controller, so that the organization can be self-sufficient and centralized. The controller and the switch convey by the OpenFlow protocol as seen in Fig.2. This approach, established as software-defined networking (SDN) owns enhanced usage of network resources than is

available with the conventional networks [9]. Applications analogous to VM (virtual machine) portability, subsequent propagation of mobile networks established on IP and an aim-critical network has secured consideration by Open Flow[3][4][5].

The OpenFlow standards have been entrenched by certain associations including Google, HP and IBM either positively utilizing it or by publicizing their motive to support. To recover stranded measuring extent, the possibility to situate virtual machines all over the data centre has been resolved by Big Switch Networks which utilized OpenFlow networks that ran over conventional networks. Initially in 2012, Google's intrinsic network ran exclusively on OpenFlow[11].

IX. CONCLUSION

The open source scheme, OpenDaylight desires to knock down the obstacles to boost the acceptance of software-defined networking (SDN) in the enterprise and provokes modernization. The open networking conception and architectures demand the realization of SDN, as they are the forthcoming networking. This project guarantees to consolidate all of the controllers by providing a standard that can be embraced by other software controllers and providers of network hardware. As networking varies, this technology will change our current mode of application to an improved mode of operation. The use of open technologies equips adaptability of dynamic network behaviour and its reaction to business requirements. Hence, the community of OpenDaylight has captured the huge responsibility of leading all the contrasting technologies and better understanding of SDN and assembling it as a united platform.

REFERENCES

- [1] Zuhran Khan Khattak, Muhammad Awais and Adnan Iqbal. "Performance Evaluation of OpenDaylight SDN Controller".
- [2] Brankica Pažun, Zlatko Langović, Ana Langović Miličević. "Integration of technologies OpenDaylight, Openstack, NFV/SDN in business environment". 9th International Quality Conference June 2015 Center for Quality.
- [3] Dhanya Bhat, Komal Rao, Latha N. "A survey on software-defined networking concepts and architecture". International Journal of Science, Technology & Management. Volume No.04, Issue No. 04, April 2015.
- [4] Thyagaraju, Mrs. Jyothi. K.S, Girish.L. "Effective disaster recovery using Software defined networking". International Journal of Engineering Trends and Technology (IJETT) – Volume 24 Number 5- June 2015.
- [5] Open Networking Foundation "OpenFlow Switch Specification" Version 1.4.0, October 14, 2013.
- [6] Open Networking Foundation "SDN architecture" Issue 1, June 2014.
- [7] Basivireddy D. "OpenDaylight integration with OpenStack". Volume 4 Issue 4, 2015 www.ijsr.net.
- [8] János Farkas, Stephen Haddock, Panagiotis Saltsidis. "Software Defined Networking Supported by IEEE 802.1Q". March 17, 2014.
- [9] ONF Std. ver. 1.3.3, "OpenFlow Switch Specification," December 2013.
- [10] OpenDaylight: Java-based OpenFlow Controller, <http://www.OpenDaylight.org/>
- [11] OpenFlow: www.openflowswitch.org
- [12] Retrieved from www.cisco.com.