ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

Review On Architecture & Security Issues of SDN

Gagandeep Garg[1], Roopali Garg[2]

Research Scholar, Dept. Of IT, U.I.E.T., PU, Chandigarh, India[1]

Coordinator, Dept. Of IT, U.I.E.T., PU, Chandigarh, India[2]

ABSTRACT: Software Defined Networking (SDN) is an emerging networking technology which separates the control-logic of data-flow from networking devices. SDN programmatically modifies the functionality and behaviour of network devices using single high level program. It separates control plane and data plane, also provides centralized control. SDN provides several benefits including, network and service customizability, improved operations and better performance. But there are some security issues that need to be taken care of. This paper describes the emergence of SDN as an important new networking technology. The main focus is to explore Security issues related to SDN. Also, the paper reviews and evaluates the salient features of SDN.

KEYWORDS*:* Software defined networking, architecture, OpenFlow, virtualization, security.

I. INTRODUCTION

Software Defined Networking is an evolving technology that decouples the intelligence of network (i.e. Control) from forwarding network devices like switches, routers, hubs etc. It separates data plane and control plane. It controls the flow of data programmatically through single high level program. Software Defined Networking emerged in 2008 by research at UC Berkeley and Stanford University. Certain communities like Open Networking Foundation,OpenFlow, etc., were founded to promote SDN. SDN architecture enables the network control to become directly programmable. It uses virtualization to make the underlying physical infrastructure to be completely hidden from applications and network services. The high level control programs are transferred from control plane to data plane using secure transfer protocols like OpenFlow. OpenFlow is most widely used SDN-Controller protocol for secure control flow [1]. One of SDN's defining characteristics is that it centralizes the network which increases the flexibility and network utilization. Using SDN, we can easily infer the behaviour of our network. Abstraction in SDN keeps the complexity of data-paths hidden from operators that's why SDN is easy to operate and maintain [2]. SDN provides reusability as single high level program can be implemented for multiple data-transfers. SDN provides rapid innovation by eliminating the dependence of hardware embedded services. SDN uses multiple controllers to provide reliability and handle the traffic-load in the network. Due to all these beneficial factors, several cloud service providers and big data centres are looking forward to SDN.

The rest of paper is organized as follows: In 2, we review the history of SDN and find out where its concept came from. In 3, we discuss the SDN architecture and different planes. In 4, we review certain features of SDN and its dominance over current traditional technology. In 5, Security of SDN discussed and mechanisms provided by different research for securing SDN. Finally in 6, we conclude the paper and discuss the future work for securing SDN.

II. BACKGROUND AND EMERGENCE

SDN emerged in 2008, but it has its roots inside the earlier technologies of late 1980's. SDN came into picture by modification of previously used technologies like NCP (Network Control Point), Active Networks and RCP (Routing Control Platform) etc. Initially in distributed networks, network administrators found that network configuration could be very buggy and unpredictable. To overcome this, people had taken the low level configuration files and tried to infer the network behaviour. But it was very difficult and time consuming. So instead of that, in 2004 they developed a logically centralized control platform known as RCP-Routing Control Platform [3]. RCP uses existing protocol as

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

control channels and computes the routes on behalf of routers. Then these routes were transferred to forwarding devices through existing protocol like inter-domain BGP-Border Gateway Protocol as shown in Fig.1.

But, the problem in RCP was that transfer of control was constrained by what existing protocols can support [4]. So in 2005, further modification done to this and RCP was generalized to 4D architecture i.e. Data, Discovery, Dissemination and control as shown in Fig.2. Data part contains only forwarding devices; Discovery part checks for the availability of the resources for transfer of data; Dissemination part discovers the networking topology to be followed for transfer of data; and Decision part provide the logic for control to be followed during transfer.

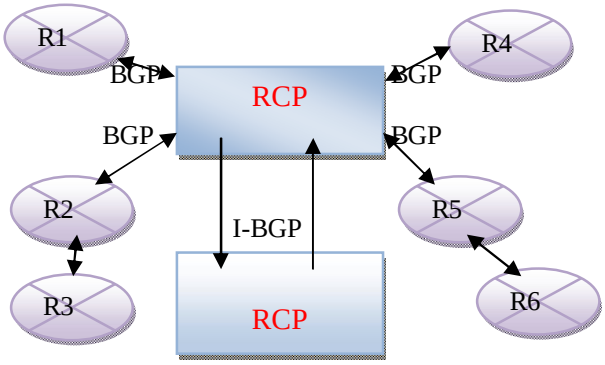

Fig.1: Routing Control Platform (RCP)

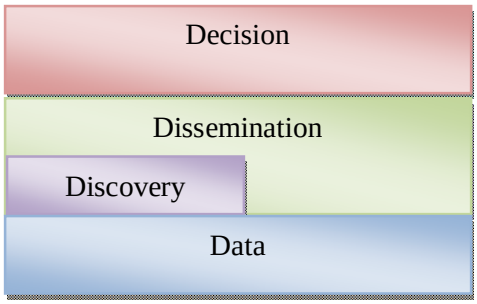

Fig.2: 4-D Architecture for distributed systems

This architecture allows better and robust control. But it was very complex and also had very high response time. Due to its inefficiencies, in 2007 a new architecture was built which is known as "ETHANE". In Ethane customized hardware was used along with a domain controller. Domain controller computes flow table entries based on network access control policies [5]. These entries were then shifted to customized hardware devices. Ethane architecture is shown in Fig.3. Now, the problem in Ethane was that it required custom switches as forwarding elements for its working. So it was very costly to implement whole new hardware. It did not provide backward compatibility with existing hardware. Finally people were looking forward to develop an architecture which could be able to provide backward compatibility; also could be able to implement the whole procedure and policies with centralized controller. In 2008, OpenFlow provided such architecture. OpenFlow provided a secure interface between controller and networking devices. This interface is compatible with both i.e. new policies of SDN controller and existing switch hardware's flow-rule updating. OpenFlow is shown in Fig.4.

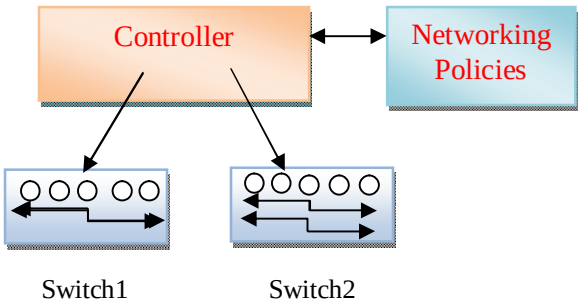

Fig.3: Ethane Architecture

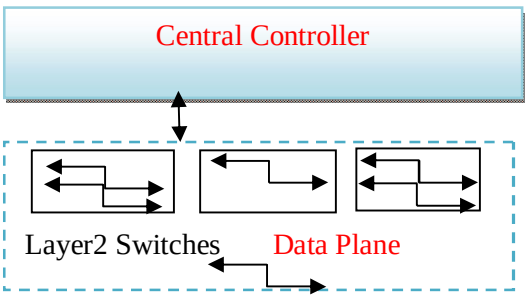

Fig.4: OpenFlow

The architecture using OpenFlow considered as the best suited architecture till now because it provides secure transfer of control from control plane to data plane [6]. It updates the flow table entries of forwarding elements according to the logic of high level control program. Later In 2009, term SDN was coined and accepted widely. OpenFlow is just an instantiation of SDN which provide protocol that configures switches using a process like an API. However SDN architecture provides programmable interfaces for high level control automation.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

III. SDN ARCHITECTURE

SDN architecture contains separate planes for Application, Control and Data. These different planes are connected by different CPI's (Control Plane Interface). Control plane resides in between application plane and data plane. It communicates with both of them through these CPI's [7]. Architecture of SDN is illustrated in Fig.5 and different planes are connected by South-bound interface and north-bound interface.

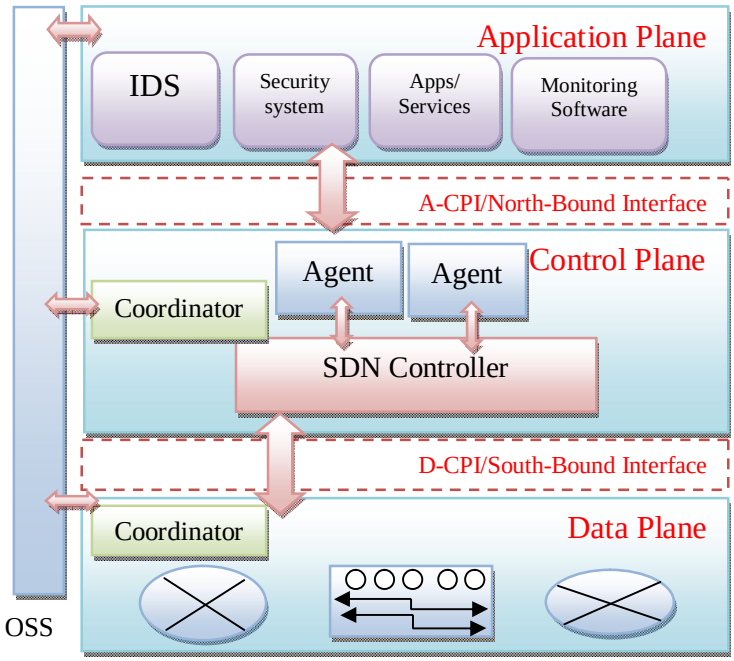

Fig.5: SDN Architecture

A. *Data Plane:*

In data plane of SDN, different physical network resources are present. These can be routers, switches, hubs, bridges etc. Data plane contains Agent which provides view of network resources known as resource Information model. Data Plane connects to SDN control plane by D-CPI (Data-Control plane interface) which is also called South-bound interface. Through D-CPI, data plane receives the control signals from SDN controller. According to these signals, network devices do packet forwarding in the network. The Coordinator in data plane is connected to OSS (Open Support System) which performs the dynamic management and handles the available network resources.

B. *Control plane:*

A Control Plane of SDN is core portion of SDN Architecture. It contains SDN controller which provides centralized control. All programming logic about packet forwarding, all network switching decisions and network routing should be written dynamically inside SDN-Controller. This high level program is then transferred to data plane via D-CPI or southbound interface like OpenFlow. In case of distributed environments, control plane can contain multiple SDN-Controllers which work in synchronization. Agents in control plane connect the SDN control logic to Application plane via *A-CPI* (Application-control plane interface) [4]. A-CPI provides virtual view of network resources available in the network to applications and other SDN-Controllers. Coordinator in control plane performs similar function of management as in data plane.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

C. *Application Plane:*

Application Plane of SDN contains applications like security system, IDS, monitoring services, etc., and general applications which require access of network devices. Applications connect to SDN controller agent via A-CPI and see the virtual view of resources available, provided by controller agent. Then applications demand service of those resources from SDN controller to transfer data. SDN controller then decides the efficient logic for traffic flow through programs and sends the control to data plane for actual transfer of data and acknowledges the application. Coordinator in application plane also manages the resources allocated to different applications.
Besides the paradigm of SDN architecture, which provides separation of data and control, SDN overcomes different failure of traditional networking [8], which are:

- Simple technical inability to handle modern mega-data centres
- The high costs of networking equipment as compared to other equipment in data centre.
- A mismatch to the rate of innovation.

SDN introduced server virtualization that allowed data centers to reach a scale that is unsustainable for traditional networking technologies.

IV. SDN FEATURES

SDN has several features which contribute towards the enhancement and efficiency of networking, which makes it a promising future technology for networks and big data centres. Some of the prominent features [9] can be discussed as:

A. *Centralized Control:*
Network intelligence is logically centralized in SDN control plane, which gives us a global view of the network and the whole network appears as a single logical switch to applications. With SDN, network administrators gain vendor-independent control over the entire network from a single logical point, which highly simplifies the network operations and design. Centralized control is very beneficial in case of distributed networking scenario.

B. *Abstraction and Virtualization:*
SDN uses abstract forwarding on each layer of layered approach used in SDN architecture, which hide the complexity of traffic flow in the network. SDN hides complexity of network from applications by providing logical view of network resources available and abstracting the actual traffic-flow control logic. Router can be divided into different virtual networks to implement different program logic on those networks.

C. *Programmability:*
SDN gives us freedom to write immediate program logic for controlling the data-flow dynamically. Instead in traditional networking, network devices like switches, routers etc., compute the best path for traffic flow by its own. Hence it increases the speed of data flow by minimizing the delay of path computation inside network devices as network devices only perform packet forwarding.

D. *Rapid Innovation:*
SDN helps in rapid innovation of new services deployment. In current traditional networking devices, services are already embedded with the hardware. These devices perform operations like path computation etc. by itself. So deployment of new application is limited to the services that came embedded with the hardware. However in SDN, control plane and data plane separation allows us to rapid deployment of unlimited services as networking devices only perform packet forwarding.

E. *Openness:*
SDN provides open standards due to which several open source communities like OpenFlow, Open Networking Foundation, ON.Lab etc. are working dedicatedly towards SDN. SDN contains open programming API's where any network administrator can write the control-logic of traffic flow according to its own infrastructure needs. Such openness allows flexibility and faster growth of new networking techniques.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

V. SECURITY OF SDN

Securing SDN is very critical area which must be taken care of; as SDN provides global view and programmability to control. It also enhances the threat of attacks possible on our SDN network as compared to existing traditional networks. It provokes our attention towards ensuring prominent security mechanisms for SDN. SDN centralized control including with certain benefits, also includes the increased threat of Denial of Service (DOS) attack potential on our network. Attackers just only concentrate on that central part for attacking or sending fake multiple service requirement requests, which can lead to overflow or full flow table entries and Denial of Service.

To cope with such threats some authentication and trust rules are defined in SDN. In 2006 SANE architecture (Secure Architecture for the Networked Enterprise) is presented which was responsible for authentication of hosts. But its implementation requires extreme changes into network environment and infrastructure which was not feasible for all organizations. So another architecture which does not require such large changes called Ethane is proposed with authenticable controller. OpenFlow is currently providing secure control transfer in network. But these approaches still have certain drawbacks and still experiencing certain security threats like traffic flooding and malicious injection of data at South-bound interface. Malicious programming logic implementation by intruder can also pose bigger threats which invoke the requirement of verification of programming. All these aspects possess a vital requirement for concentration on Security of Software defined networking.

Current Research on security introduced some larger amount of Virtualization scenario on different layers of SDN, so that attackers cannot come to know about our actual physical network. Also, SDN abstracts the inner data flow to reduce the malicious network injection and traffic flooding introduced by intruders [10]. *Abstraction on each layer causes important effects and produces difficult scenarios for attackers to breach through.* Some of the security mechanisms provided in the previous research on SDN security are:

- A survey presented in [11] has shown the various categories of security threats associated with SDN layered framework defined in SDN architecture. This defines different security attacks like Data leakage i.e. spoofing, unauthorized access, data modification, denial-of-service, malicious applications etc. that are possible at different parts of SDN framework.
- SDN Scanner [12] exploits the network header field change scanning, which scans networks as changing header fields and records the response time of each packet. It compares the response time and then use statistical tests. With the fingerprinting results of SDN, it is possible to conduct resource consumption attack on SDN with 85.7% accuracy. Hence new defence solutions need to be designed to overcome such threats.
- AvantGuard [13] proposed new architecture as data plane extensions to protect our network from control plane saturation attack that disrupts network operations. AvantGuardintroduces connection migration, actuating triggers over the data plane's existing statistics collection services. It provides both detection of, and responses to, the changing flow dynamics within the data plane. Connection migration enables the data plane to protect the control plane from such saturation attacks. AvantGuard increases the scalability and responsiveness towards threats, with connection delay overhead of 1%.
- AMQ [14] proposed a technique of detecting and isolating insecure network devices in Data centres, before they effect negatively to the network. On discovering a potential threat, AMQ automatically identifies the problem and download the patches necessary to resolve it. It automatically allows the device to re-join the network on resolution. In AMQ, two primary security network service modules (NSM) hosted on the controller. First a Bot-Hunter monitor the network and detect a malware-infected host in real-time. Secondly a threat responder NSM directs the controller to initiate the quarantine procedure to isolate the threat. When a host is quarantined, the Web Proxy Notifier is activated to inform the user on the infected host that security has been compromised. It can be used for moderate-speed links only.
- OpenWatch [15] proposed a method that performs adaptive zooming in the aggregation of flows to be measured. It provides more flexible and interactive interface to anomaly detectors to detect anomaly like Port-Scan. This method takes advantage of centralized view of SDN and provides an algorithm that intelligently allocates flow counting task among multiple switches to reduce single switch overhead. It proposed an adaptive algorithm to control temporal and spatial aggregation of the counting function based on linear prediction.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

OpenFlow networking community including with several other organizations are implementing different abstraction and virtualization rules on each layer for enhancing the security of SDN. Using already existing protocols and also designing new protocols like implementation of middle boxes and monitoring systems for dynamic threat suspicion. Middle-boxes transforms, inspects, filters and manipulate traffic and include Firewalls, NAT-Network Address Translators etc. With all these security mechanisms we can further explore the security threats for SDN by implementing IDS (Intrusion detection system) and other detection methods like analysing the traffic, auditing data-flow on centralized control of SDN.

VI. CONCLUSION AND FUTURE WORK

Separation of planes and the various features of SDN make it very efficient technology which is going to be the future of networking. The importance of SDN lies in its characteristic that it provides flexibility by programming the control, including the centralized control, which helps in handling the whole network. It becomes more beneficial in case of distributed SDN networks working in synchronization. Also it is helpful in controlling, providing scalability and management of data in large data centres. The Abstraction and Virtualization of resources helps in securing the network and hiding complexity. There are still different areas which are required to be taken care of, like securing the SDN control plane, as whole control of SDN is centralized in control plane; hence security of Control plane is very important and prevention from several attacks is main area of concern. Also openness of SDN system allows people to write control programs so it is essential to design some protocols or use existing protocols efficiently that will check the correctness of programming logic before implementation i.e. it will check the authentication and authorization so as to prevent the collisions of data inside the network which causes congestion. Security of southbound interface needs special attention as control transfers through this interface. Procedures to be designed for prevention from DOS attack. Different anomalies in fuzzy logic need to handle which are causing obstructions to wide deployment of SDN.

REFERENCES

1. S. A. C. Risdianto, E. Mulyana, “Implementation and Analysis of Control and Forwarding Plane for SDN”, IEEE Telecommunication Systems, Services, and Applications (TSSA) 7th International Conference,pp. 227 – 231, 2012.
2. Z. Bozakov, P. Papadimitriou, “Towards a Scalable Software-Defined NetworkVirtualization Platform”, IEEENetwork Operations and Management Symposium (NOMS), May, pp. 1 – 8, 2014.
3. N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, and J. van der Merwe, “The case for separating routing from routers”,ACM Proc. SIGCOMM Workshop on Future Directions in Network Architecture, pp. 5-12, 2004,.
4. M. Caesar, D. Caldwell, N. Feamster, J. Rexford, A. Shaikh, J. van der Merwe “Design and Implementation of a Routing Control Platform”, ACM Proc. of NSDI, Vol. 2, pp 15-28, 2005.
5. M. Casado, M. J. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker. “Ethane: Taking Control of the Enterprise”, ACM Proc. of SIGCOMM conference on Applications, technologies, architectures, and protocols for computer communications, Kyoto, Japan, pp 1-12, 2007.
6. N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar , L. Petrson , J. Rexford, S. Shenker , and J. Turner , “OpenFlow :Enabling innovation in campus networks”, ACM SIGCOMM Computer Communication, Vol. 38, Issue 2, pp. 69-74, 2008.
7. M. Betts, S. Fratini, N. Davis, R. Dolin and others, “SDN Architecture”. Open Networking Foundation ONF SDN ARCH, Issue 1, June, 2014.
8. P. Goransson, C. Black, “WHY SDN”, Software Defined Networks- A Comprehensive Approach, pp. 21-35, 2014.
    a. Y. Ding, J. Crowcroft, S. Tarkoma and H. Flinck, “Software defined networking for security enhancement in wireless mobile networks”, Computer Networks,(Elsevier) vol. 66, pp. 94 -101, 2014.
9. M. H. Razaa, S. C. Sivakumar, A. Nafarieha, B. Robertson, “A Comparison of Software Defined Network (SDN) Implementation Strategies”, Elsevier Proc. of 2nd International Workshop on Survivable and Robust Optical Networks (IWSRON). Vol. 32, pp. 1050-1055, 2014.
10. S. Scott-Hayward, G. O’Callaghan and S. Sezer,”SDN Security: A Survey”, IEEE Proc. of SDN4FNS, Trento Italy, pp. 1 - 7,2013.
11. S. Shi, G. Gun, “Attacking Software-Defined Networks: A First Feasibility Study”, ACM Proc. of HotSDN, Hong Kong, China, pp. 165-166, 2013.
12. S. Shin, V. Yegneswaran, P. Porras, G. Gu, “AVANT-GUARD: Scalable and Vigilant Switch Flow Management in Software-Defined Networks”, ACM Proc. of CCS, Berlin, Germany, pp. 413-424, 2013.
13. M. McBride, M. Cohn, S. Deshpande, M. Kaushik, M. Mathews, S. Nathan, “SDN Security Considerations in the Data Center”,Open Networking Foundation- ONF SOLUTION BRIEF, 2013.
14. Y. Zhang, “An adaptive flow counting method for anomaly detection in SDN”, ACM Proc. of CoNEXT, Santa Barbara, California, USA December, 2013, pp. 25-30.