TELKOMNIKA, Vol.17, No.6, December 2019, pp.3168~3174
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v17i6.13119

A review on software defined network security risks and challenges

Tsehay Admassu Assegie*, Pramod Sekharan Nair
Department of Computing Technology, Aksum University, Aksum, Ethiopia
*Corresponding author, e-mail: tsehayadmassu2006@gmail.com

Abstract

Software defined network is an emerging network architecture that separates the traditional integrated control logic and data forwarding functionality into different planes, namely the control plane and data forwarding plane. The data plane does an end-to-end data delivery. And the control plane does the actual network traffic forwarding and routing between different network segments. In software defined network the networking infrastructure layer is where the entire networking device, such as switches and routers are connected with the separate controller layer with the help of standard called OpenFlow protocol. The OpenFlow is a standard protocol that allows different vendor devices like juniper, cisco and huawei switches to be connected to the controller. The centralization of the software defined network (SDN) controller makes the network more flexible, manageable and dynamic, such as provisioning of bandwidth, dynamic scale out and scale in compared to the traditional communication network, however, the centralized SDN controller is more vulnerable to security risks such as DDOS and flow rule poisoning attack. In this paper, we will explore the architectures, the principles of software defined network and security risks associated with the centralized SDN controller and possible ways to mitigate these risks.

Keywords*: centralized SDN controller, distributed SDN controller, network security, SDN controller security, software defined network*

1. Introduction

A software defined network is network that can be programmed. This implies that, the equipment in traditional network such as, routers or switches can be replaced by software programs. And such devices can be controlled by a software program that is not embedded in the hardware itself. For example, in the traditional network, a link bandwidth on Ethernet switch is configured on the switch itself, but in software defined network, this can be done on the OpenFlow controller [1].

As the number of devices connected to the network increase day by day and the capability of the routing table is limited, a traditional IP network become increasingly difficult to manage the devices' in a network and configuration errors have becomes common problems [2-5]. The administrator has to issue usually a vendor specific commands to set policies, routing information and many network parameters required for the networking device to function and operate smoothly.

The complexity of devices management had therefore, motivated the researchers to think about a new networking paradigm, which is the software defined network. Unlike, the traditional network, the new approach is based on decoupled hardware, control program, which is called the SDN controller. The decoupling of the software from the hardware simplifies network management and is cost effective when compared to the traditional approach.

The software defined network is a network architecture where the control logic and forwarding functionalities are separated into different layers, the controllable program and the physical infrastructure layers. The key principles of software defined network are [6]:

- Control plane-deals with IP routing and forwarding decisions
- Data plane-responsible for end to end delivery of data in the network.
- Programmable network centrally managed- the management functionalities are centralized.
- Open interfaces between the device in the control plane and the data plane.

In software defined network, the control plane, manages the entire device in the network; therefore, it is the core element of the network. The routing policies or the flow

Received May 14, 2019; Revised July 2, 2019; Accepted July 18, 2019

rules, security policy and every management aspect of the network are centralized through the use of SDN controller. While the use of the centralized SDN controller is an opportunity to simplify the network management and administration, it has also introduced new security issues because of the centralized SDN controller architecture [7]. On each layers of the software defined network, there are different security risks, and the focus of this paper is on security issues and risks on the control plane.

2. Related Works

This section focuses on some of the research works related to the implementation, challenges and future directions of the software defined network primarily, the security challenges at control layer. Although, there are some works related to the design and implementation of secure software defined network architectures that can solve the problems of security in software defined network, software defined network is still in its infancy and further researches are required to address the security challenges to the centralized software defined network (SDN) controller.

A software defined network as a new technological advancement in networking, is supposed to solve the traditional networking complexity issues by breaking down the data forwarding spoke from the brain or control software. But the shift from the traditional to the software defined networks is facing a lot of practical implementation problems. One of these problems is security issues associated with centralized software defined network (SDN) controllers [8]. Software defined network being a platform that provides a cost-effective more flexible and centrally controlled high speed network services when compared with the traditional network [9, 10], has also introduced a security risks which are the main concerns in the traditional network and new security risks.

Unlike, in traditional networks where every device is managed by a program embedded into the device by its manufacturer, in software defined network, the central controller is used to manage each device in the network. And the devices are connected to the central controller with the help of standard protocols such as the Open Flow protocol. This makes network management task simpler and better network flexibility and programmability. With this approach, devices from different vendors may be used in the network using this standard. However, the devices in software defined network can receive a manipulated flow rules, and policies from the centralized SDN controller, this is one of the security risk in this approach.

One of the most significant risk factor in software defined network is the possibility of compromising the SDN controller, known as an attack on the controller. Due to the centralization of SDN controller, an attacker can control the entire network if the SDN controller is attacked and in the worst cases the entire network may go down, because, the attackers may also be motivated on the centralized SDN controller to manipulate or alter the network. The SDN control layer is also susceptible to denial of service attack (DOS), an attacker can use the switches to cause the software defined network control plane to be flooded with numerous requests that possibly causes a delay or drop of the requests [11] to the controller. The possible defense against to this form of attack is implementing a distributed physical software defined network controllers.

The characteristic features of security risks and challenges in software defined networks are different in some aspects [12] from that of the traditional network, due to the particular network implementation and SDN's characteristic control and programmability characteristics. For example, the concept of the centralized control logic may expose the flow rules and configuration files to attackers and the ability to directly access the control plane results in a new attack surface [13] and the possibilities of the entire network getting down.

Several studies [14] have attempted to study distributed SDN controllers. However, despite their attempt at distributing the control plane, they require a consistent network wide view in all the controllers. But distribution of the controllers creates another problem, that is, synchronization of distributed SDN controllers and concurrency control in a highly dynamic software defined network environment is challenging. Since the SDN switches look to the controllers for answers while handling unfamiliar flows, knowing which controller to ask is important. Moreover, the controllers themselves need to have methodologies to decide who controls which switch, and who reacts to which events. And most of all, consistency in the security policies present on the controller is paramount, the absence of which might result in

attackers using application traversing across multiple partitions of the SDN environment without permissions.

3. SDN Security Classification

The separation of control logic from the data plane in SDN has introduced new vulnerabilities, which are not found in the traditional network. For example, the use of transport layer security is optional in the OpenFlow network. The nature of the communication protocol can thus introduce security issues such as DoS attack, fake flow rule insertion, and even flow rule alteration [15]. Figure 1 demonstrates the different components of SDN: (1) application layer, (2) control layer and (3) infrastructure layers, which are exposed to attacks. For example, there can be software vulnerabilities in SDN controllers, such as POX which can be exploited by an attacker to perform an attack on control plane. Additionally, the communication paths between the three layers in the SDN, the northbound interfaces connecting the controller with application layer and the southbound interfaces connecting the infrastructure layer to the SDN controller, can face security attacks. Some of the possible attacks against each layer of the software defined network is discussed in detail below:

- Application layer security: The applications implemented for the adaptation and other SDN operations can have security vulnerabilities. All the security issues that can be present in a typical software application such as, a buffer overflow attack also apply to the software defined network. And the malicious or a compromised application allow blowout of attack in the complete network [16].
- Control layer security: The control plane consists of one or more controllers, such as NOX, POX, and other programs used for managing the infrastructure layer device and handling different kinds of protocols such as, the OpenFlow. The attacker can generate traffic from spoofed IP address and send a huge volume of traffic to the controller as discussed by [17]. By performing such attacks, the communication between the switch and the controller can be saturated, thus increasing service latency or in the worst case bringing down the SDN controller which results in the entire network getting down.
- Infrastructure layer security: an attacker can delay traffic between the SDN controller and infrastructure layer by generating network traffic with poisoned addresses and forwarding the traffic to the controller to overload it [18]. Buffer overflow attack is another possible attack on infrastructure layer. The switches are responsible for forwarding data but, fake flow rules generated by attacker can overflow the switch's flow table since they have limited memory.
- The Communication Channels security: The communication channel between the infrastructure layer devices, such as switches and the SDN controller, which is the southbound interface, the SDN controller and the application layer, which is the northbound interface are exposed to a man-in-the-middle attack as showcased in [19], the switch's flow table can even be filled with a fake flow rules and by poisoning the flow table, traffic can be redirected to a man in the middle. Other attacks researched by authors that can be performed on the communication channel include eavesdropping traffic between hosts, and stealthily modifying the traffic between hosts as they traverse through the physical communication channel.

3.1. Centralized SDN Controller

In this section, we will briefly introduce the centralized SDN controller architecture and the security challenges associated with this architecture, primarily on the control layer. The centralized SDN controller architecture is illustrated in Figure 1. As demonstrated in Figure 1, the architecture is layered into three layers, the controller, application and network infrastructure, and we focus on the security issues on controller.

The centralized control or logically centralized control that is, distributed but coordinated control function exposes the data such as configuration file, vulnerable to the attackers [20]. Attackers may attempt to operate the common network services or even control the entire network by misleading or compromising the controller which is the heart of the network. The centralized architecture of SDN controller and lack of defenders capability, and undeveloped technology possibly will benefit the attackers. For instance, the introduction of malicious controller programs may allow a wider impact of the attack [21]. The solution to

this issue is storing the critical configuration files and the flow rules in a completely distributed control domain. The distributed SDN controller architecture is discussed in detail in the following section.

Another issue with software defined network arises from its flexibility. The increased flexibility decreases the possibility of stronger defending mechanism for the software defined network. Because of this feature the SDN becomes dynamic and in such configurations the state of the system is difficult to determine. This will create an opportunity for the attackers, once they gain access into the system; identification of the legitimate users from the attacker is difficult.

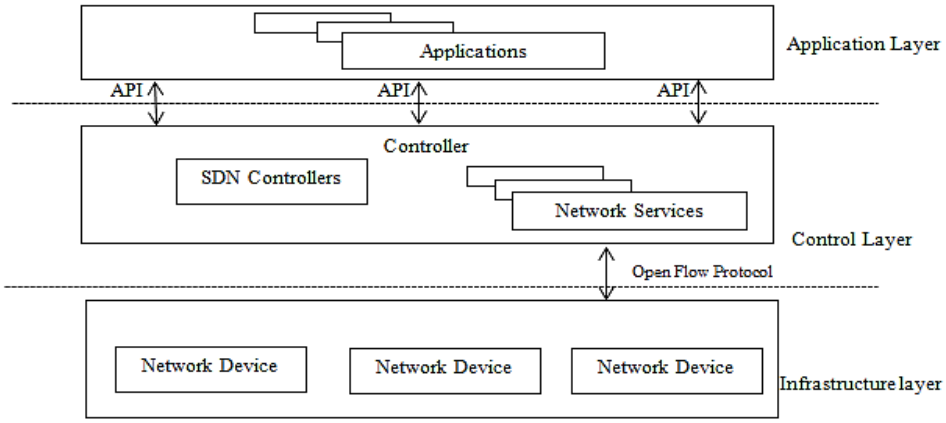

Figure 1 .The Centralized SDN controller architecture [22, 23]

3.2. Distributed SDN Controller

This section discusses software defined network controller implementation options and the proposed distributed software defined network controller architecture. While the software defined network controller centralization has solved the problems of network manageability it has created a compromise and may cause single point of failure if the system is compromised by malicious guys or attackers.

The original design of SDN was based on a centralized control plane as the objective of SDN is to separate the control plane from the data plane and allow the global view of the whole network. This allows the controller with a comprehensive network-wide view and permits for the development of control applications and for easier policy enforcement. However, centralizing the control plane in SDN is fraught with scalability and security challenges associated with the SDN controller being a bottleneck [24].

Secure software defined network architecture can solve the problems of security in software defined network [25] at the control layer. But a solution to this problem requires an in-depth long-term research. Without the use of a secure architecture for SDN, the controller plane is at risk. Some of the solutions proposed to deal with the security issues on the controller layer are:

- Hardening the control logic program on the SDN controller-this deals with making the controller itself secure from any form of attack. As there is no completely secure system that is ever built since, this is very complex and one day the SDN controller may be compromised.
- Decentralising the SDN controller- a completely distributed implementation of SDN controller architecture can solve the problem with centralized SDN controller. However, the complexity of network management will increase as a result of SDN controller distribution. The distributed SDN controller architecture is demonstrated in Figure 2.

Centralized architecture of the SDN controller creates additional complexity and challenges for security controls on the control layer. As short listed in [26], there are three options to implement the SDN controller architectures, centralized, decentralized or distributed

and multi-layer architectures but, it is common to couple software defined network architecture with a centralized SDN control to deploy network services or applications.

As shown in Figure 1, the software defined network architecture, which is proposed by the open network foundation (ONF), consists of three layers, namely the application, the SDN controller and the data forwarding layer and these layers are vulnerable to attack. And this vulnerability specially, the control layer vulnerability can be used by attackers to gain unauthorized access to the SDN controller, to intercept and manipulate network traffic. For example, the use of pre-configured clear text shell access on switches running on Linux and the outdated SSL implementation puts the entire software defined network system at risk. Apart from that, the introduction of software defined network functions can increase the attack surface. A security breach in these applications enables an attacker to evade isolation mechanisms and compromise the complete network or execute illegal actions on other networks [27].

The centralization of the control logic creates additional security risks to the SDN controller, if this core component of the software defined network is compromised, then the entire SDN network is at risk of failure. There are two security risks with the centralization of SDN controller, these are, 1) the controller itself is prone to attacks, such as DDOS and flooding, 2) malicious flow rules can be generated and forwarded to the forwarding plane, as there is no mechanism used to identify which flow polices are genuine and which are malicious, this creates a problem on the network. This is because, once the controller is compromised it can transmit a malicious flow rules to the network elements in software defined network.

In [28], comprehensive security architecture was recommended to provide a variety of security services which also includes the implementation of required network security policies, packet data scan detection mechanisms, altering network policies into flow entries, authentication and authorization for addressing the security challenges related to policy enforcement and attack detection in a software defined network.

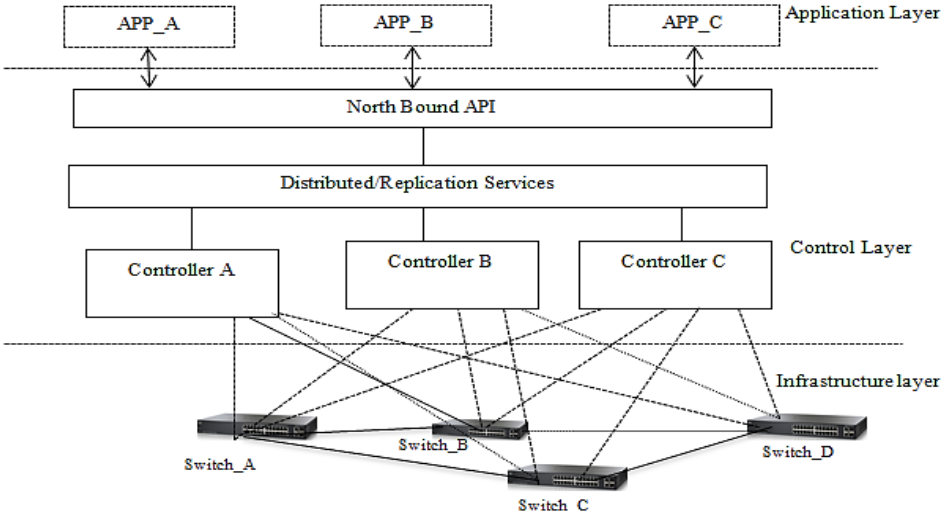

Figure 2. The distributed SDN controller architecture [29]

4. SDN Security Framework

As discussed in earlier sections, the use of software defined network, especially the centralized SDN controller architecture exposes the controller to different forms of attacks like, DDOS, switch flow table flooding, insertion of fake flow rule and so on. This has encouraged the researchers to the deployment of a variety of network security functions, intrusion detection and prevention systems, and access control and identity management in the software defined network to protect the SDN controller and even all of the layers form different forms of attacks. The architecture of the preliminary software defined network security framework focuses on the functions of the frameworks [30] deployment architecture is demonstrated in Figure 3 and some of the frameworks are discussed below:

- Intrusion detection and prevention- intrusion detection and prevention systems, such as stateful packet inspection tools, like firewalls and packet inspection tools can be used to mitigate security issues related to SDN.
- Access control-stateful packet inspection and next-generation firewalls performing deeper, context-related traffic analysis can solve the problem of the centralized SDN controller attack.
- Protecting Denial-of-service (DoS) - these are systems implemented and configured specifically for detecting and defending against DoS attacks. This protection primarily comprises DoS defence systems, many of which perform traffic analysis and enforce access control and flow policies. It also includes the use of switches and routers with traffic rate-limiting capabilities.

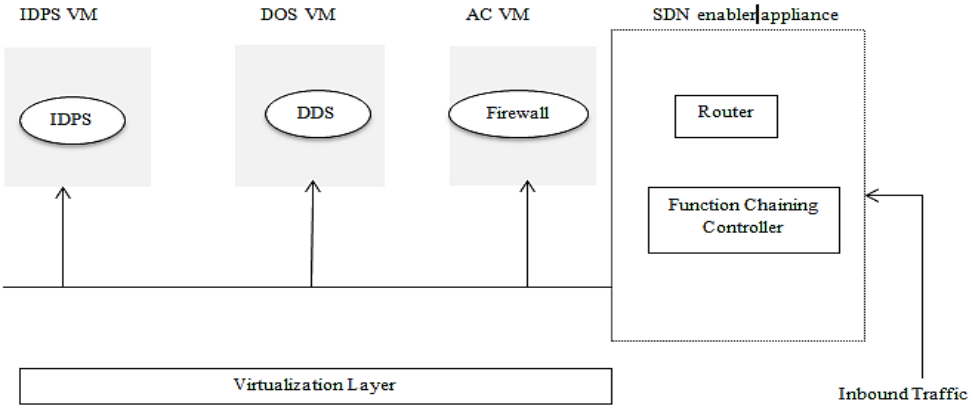

Figure 3. Deployment scenario for SDN security frameworks [31]

5. Conclusion

The software defined networking architecture provides a virtualized network and is supposed to transform today's network into flexible, cost-effective and programmable platforms. The future of networking will depend on software defined network because of its flexibility, simplicity and dynamic nature, in turn; it will become the new norm for networks. However, there is a critical security problem that needs to be addressed regarding the control logic and applications layer, before it can be securely deployed.

In this paper, we have explored the security risks at the SDN control layer and a decentralized SDN controller architecture as a solution to mitigate the security risks associated to the SDN controller. Finally, we have discussed the frameworks and some of the methods used in protection of critical assets such as the configuration files, flow entries and flow rules from an attack.

References

[1] Tsehay Admassu Assegie Pramod Sekharan Nair. Performance analysis of emulated software defined wireless network. *Indonesian Journal of Electrical Engineering and Computer Science*. 2019; 16(1): 311-318.
[2] Keti F, Askar S. *Emulation of Software Defined Networks Using Mininet in Different Simulation Environments*. 6th International Conference on Intelligent Systems, Modelling and Simulation. 2015: 205-210.
[3] Rawat DB. Software Defined Networking Architecture, Security and Energy Efficiency: A Survey. *IEEE Communication Surveys & Tutorials*. 2017; 19(1): 325-346.
[4] Kreutz D. Software-Defined Networking: A Comprehensive Survey. *IEEE*. 2014.
[5] Nguyen XN, Saucez D, Barakat C, Turletti T. Rules Placement Problem in OpenFlow Networks: a Survey. *IEEE Communication Survey and tutorials*. 2015; 18(2): 1273-1286.
[6] King D, Rotsos C, Aguado A, Georgalas N, Lopez V. *The Software Defined Transport Network: Fundamentals, Findings and Futures*. 18th International Conference on Transparent Optical Networks (ICTON). 2016: 1-4.

[7] Sezer S, Scott-Hayward S, Chouhan PK. Implementation Challenges for Software-Defined Networks. *IEEE Communications Magazine*. 2013; 51(7): 36-43.
[8] Benzekki K, El Fergougui A, Elbelrhiti Elalaoui A. Software-defined networking (SDN): a survey. Security and communication networks. 2016; 9(18): 5803-5833.
[9] Hande YS, Akkalakshmi M. A Study on Software Defined Networking. *International Journal of Innovative Research in Computer and Communication Engineering*. 2015; 3(11).
[10] Sahay R, Meng W, Jensen CD. The application of Software Defined Networking on securing computer networks: A survey. *Journal of Network and Computer Applications*. 2019; 131: 89–108.
[11] Sinha Y, Haribabu K. A survey: Hybrid SDN. *Journal of Network and Computer Applications*. 2017; 100: 35–55.
[12] Karakus M, Durresi A. A survey: Control plane scalability issues and approaches in Software-Defined Networking (SDN). *Computer Networks*. 2017; 112: 279–293.
[13] Zhang T, Giaccone P, Bianco A, De Domenico S. The role of the inter-controller consensus in the placement of distributed SDN controllers. *Computer Communications*. 2017; 113: 1–13.
[14] Schallera S, Hoodb D. Software defined networking architecture standardization. *Computer Standards & Interfaces*. 2017; 54: 197–202.
[15] Wibowo FXA, Gregory MA, Ahmed K, Gomez KM. Multi-domain Software Defined Networking: Research status and challenges. *Computer Standards & Interfaces*. 2017; 54: 197–202.
[16] Singh D, Ng B, Lai YC, Lin YD, Seah WKG. Modelling Software-Defined Networking: Software and hardware switches. *Journal of Network and Computer Applications*. 2018; 122: 24–36.
[17] Birkinshaw C, Rouka E, Vassilakis VG. Implementing intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks. *Journal of Network and Computer Applications*. 2019; 136: 71-85.
[18] Asadollahi S, Goswami B, Gonsai AM. Software Defined Network, Controller Comparison. *International Journal of Innovative Research in Computer and Communication Engineering*. 2017; 5(2).
[19] Underdahl B, Kinghorn G. Software Defined Networking for Dummies. John Wiley & Sons. 2015.
[20] Al-Abri Z, Al Maashri A, Al-Abri D, Shiginah FB. Using SDN as a Technology Enabler for Distance Learning Applications. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*. 2018; 6(2): 225-234.
[21] Jammal M, Singh T, Shami A, Asal R, Li Y. Software defined networking: State of the art and research challenges. *Computer Networks*. 2014; 72: 74-98.
[22] Open Network Foundation. Software-Defined Networking: The New Norm for Networks. ONF White Paper. April 13, 2011.
[23] Farahmandian S, Hoang DB. Security for software-defined (cloud, SDN and NFV) infrastructures–issues and challenges. *Computer Science & Information Technology (CS & IT)*. 2016: 13– 24.
[24] Milenkoski A, Jaeger B, Raina K, Harris M, Chaudhry S, Chasiri S, David V, Liu W. Security Position Paper Network Function Virtualization. Cloud Security Alliance. 2016
[25] Palo Alto. ONF TR-511. Principles and Practices for Securing Software-Defined Networks. January 2011
[26] Dacier MC, Dietrich S, Kargl F, König H. *Network Attack Detection and Defence Security Challenges and Opportunities of Software-Defined Networking*. Dagstuhl Seminar 16361. 2016; 6(9): 1.
[27] Koldehofe B. Flexibility and Adaptability for Attackers and Defenders, 16361–Network Attack Detection and Defense. 2016
[28] Scott-Hayward S, O'Callaghan G, Sezer S. *SDN security: A survey*. Future Networks and Services (SDN4FNS), 2013 IEEE SDN For. IEEE. 2013: 1–7.
[29] Kalkan K, Gur G, Alagoz F. Defence Mechanisms against ddos Attacks in sdn Environment. *IEEE Communications Magazine*. 2017; 55(9): 175– 179.
[30] Gao S, Li Z, Xiao B, Wei G. Security Threats in the Data Plane of Software-Defined Networks. *IEEE Network*. 2018; 32(4): 108-13.
[31] Romão D, Van Dijkhuizen N, Konstantaras S, Thessalonikefs G. Practical Security Analysis of Openflow. University of Amsterdam, Amsterdam. 2013.