Security in Software Defined Networking : A Survey

Shreyas Prabhakar
Computer Science Department
Clemson University
Clemson, United States of America
sprabha@clemson.edu

***Abstract*—Software Defined Networking (SDN) is an emerging network paradigm in which all the network traffic may be managed dynamically based on the user requirements and demands, making it more feasible for all networking domains. SDN breaks the vertical paradigm of the traditional networking and provides the flexibility to program the network through (logical) centralized network control. SDN is characterized by its two distinct features, which include – 1. The decoupling of the control plane from the data plane by moving the control plane out of the network hardware and providing centralized network control. 2. Providing programmability of the control plane for any network application. As a result, SDN provides better resource management, high performance, efficient configuration and flexibility to innovate and build network designs and applications. Due to the increasingly pervasive existence of smart programmable devices in the network, SDN provides security, energy efficiency and network virtualization for enhancing the overall network performance. We present various security threats that are resolved by SDN and new threats that arise as a result of SDN implementation. We also provide a survey on the different strategies that are implemented to achieve network security through SDN implementation. In an effort to anticipate the future evolution of this new paradigm, we discuss the main ongoing research efforts, challenges and research trends in this area. With this paper, readers can have a more thorough understanding of SDN architecture, different security attacks and countermeasures.**

Keywords— Software Defined Networking; SDN; Network Virtualization; Network Programmability; Security Solution; Threat Vectors; Threat; Security; Counter Measures

I. INTRODUCTION

Emerging mega trends in Information and communication technologies domain, like mobile, social, cloud, big data and Internet of Things, put a lot of stress on the computer networks asking for higher bandwidth, ubiquitous accessibility and dynamic management. The growing popularity of the rich multimedia content on the Internet like streaming HD videos and audio, ever increasing applications of Internet of Things and the increasing demand for analytics on Big Data require faster connection speeds than ever before. The widespread consumption of mobile technologies and social networking has put a lot of demand on ubiquitous communications to fulfil the social need of the general population. With the emergence of big data and cloud computing, more computing, storage and other resources and moved to remote data centres. Higher bandwidth has become critical to access these resources via a network. As such, computer networks are becoming crucial in enabling the emerging technologies to move forward.

Software Defined Networking is an emerging networking paradigm that was developed to facilitate the innovation and enable simple programmable control over the network data-path. The separation of the data forwarding data plane from the control logic plane in SDN, makes it easier for the deployment of new protocols and applications, and straightforward network visualization and management. Instead of enforcing and running policies on the hardware devices which are distributed across the network, the network is simplified as in the simple data forwarding devices and the decision making central controller.

In this survey paper, we start by briefly discussing about the definition and Reference model of SDN in Section 2. In Section 3, we talk about SDN as a security solution as compared to the traditional networks. Then, we move on to attacks and attack vector that arise because of the implementation of SDN and possible solutions Section 4. Finally, in Section 5, we conclude the paper.

II. SOFTWARE DEFINED NETWORKING

A. Definition of SDN

The Open Network Foundation(ONF) is a non-profit consortium dedicated to development, standardization and commercialization of SDN. ONF has provided the most explicit definition of SDN as follows:

> SDN is an emerging network architecture where network control is decoupled from forwarding and is directly programmable.

As per this definition, there are two major characteristics which define the architecture of SDN –

1. Network as decoupled control and data planes.
2. The control plane of SDN architecture is programmable.

The uniqueness of SDN resides on the fact that it decouples the control and data planes and provides programmability of the central control layer. With the growing complexity of the networks, SDN makes it easier to manage and control the network by separating the control plane and the data plane.

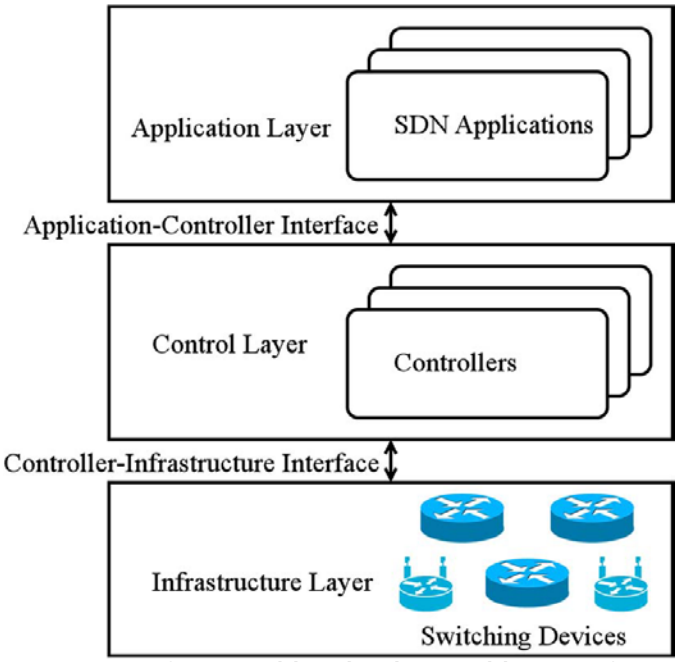

Fig 1: SDN Reference Model: a three-layer model, ranging from an infrastructure layer to a control layer to an application layer, in a bottom-up manner

This helps in abstracting the network design and virtualization of the network. In SDN, the intelligence is moved from the network hardware equipment, such as routers and switches, to the central control system. With this design, the switching systems can take care of forwarding the data and can be controlled externally by the software. The network control can be done independently on the control plane by using simple programming environments without affecting the data plane giving it more flexibility

B. *SDN Reference Model*

ONF has also suggested a reference model for SDN, as illustrated in Figure 1. This model consists of three layers, namely an infrastructure layer, a control layer, and an application layer, stacking over each other.

The infrastructure layer consists of switching devices such as routers and switches, and other network hardware resources in the data plane. The main functionality of this layer is data forwarding based on the instructions provided by the control layer. This layer is also responsible for collecting network status, storing them temporarily and forwarding them to the controller. This status may include network traffic statistics, network usage and network topology.

The control layer bridges the infrastructure layer and the application layer. For downward interaction (also called as south-bound interface) with the infrastructure layer, it specifies the functionalities to access and control the functionalities provided by the routers and switches. These functionalities may include communicating instructions for data forwarding and status reporting of the infrastructure layer. For upward interaction (also called as north-bound interface) with the application layer, it provides service access points in various forms, such as Application Programming Interface (API). SDN applications can use these APIs to access the network status reported by the switching devices to make routing decisions and other system tunings and send the packet forwarding instructions to the infrastructure layer using these API.

The SDN application layer consists of applications that are used to fulfil the user requirements. Through the programmable platform provided by the control layer of the SDN architecture, users are able to access and control the switching devices in the infrastructure layer using the APIs provided by the control layer.

III. SDN AS A SECURITY SOLUTION

The security aspects of SDN are discussed in two sections as shown in Figure 2. In this section (section 3), we discuss the role played by SDN as a solution to enhance the security in the existing networks. In the next section (section 4), we discuss about the security challenges that arise due to various threat vectors created by SDN and the countermeasures or those attacks vectors.

There are a lot of security threats that can be resolved using SDN. We will talk about the definition and countermeasures of these threats in this section:

A. *SDN as an Intrusion Detection System(IDS) and Intrusion Prevention System(IPS):*

An intrusion attack is an unauthorized activity on a network where attacks absorb network resources intended for other uses. Due to the reconfigurablity and programmability of SDN, the SDN can be actualized as IDS and IPS to screen the network activities ceaselessly to distinguish intrusion attacks. Most common intrusion attacks that can be protected by utilizing flexibility and programmability of SDN are

1. *Assymetric Routing Attack*: To bypass the intrusion sensors and certain network segments, the attacker uses more than one routing path to reach the targeted network device. If the network is not setup for assymetric routing, the network becomes vulnerable to this attack.
2. *Buffer Overflow Attack*: With the aim of Denial-of-Service (DoS), this attack overwrites specific sections of the device memory of a targeted network and replaces normal data in certain memory locations with a malware.
3. *Protocol-Specific Attacks*: The network protocols – such as TCP, UDP, ARP, IP, ICMP etc., may unintentionally leave a backdoor for network intrusions through spoofing or so with an aim of compromising or even crashing the targeted devices on a network. For example, while mapping IP network addresses to the hardware MAC addresses, the ARP protocol allows attackers to execute “man-in-the-middle” attack as it does not perform authentication on ARP Request and Reply messages.
4. *Traffic Flooding Attacks*: Attacker can generate traffic loads that are too heavy for the network to handle, and overwhelms the overall network resources. These kind of attacks can easily be controlled by the use of SDN.

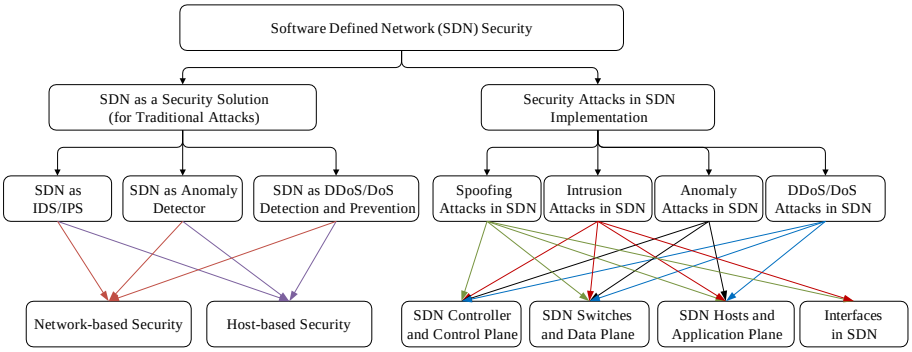

Fig. 2: Software Define Networking (SDN) as a Security Solution and Security Attacks that are unique to SDN

5. *Trojan-based Attack*: This attack instigates DoS attacks, erase stored data or open back doors to permit the system control by outside attackers.

There are a lot of different defence solutions for IDS and IPS that have been proposed, which are based on SDN.

B. SDN Based Anomaly Detection

Anomaly detection (also outlier detection) is the identification of items, events or observations which do not conform to an expected pattern or items. In recent times, the attacks are becoming more sophisticated and makes it very hard to trace the actual origin of the attack. The SDN architecture provides us a way to configure the network devices as per our needs. For example, home router that is configured using SDN works effectively to detect malware and spywares attacking the system.

With the implementation of SDN, collaborative detection can be implemented through already existing centralized SDN controller, where each each switch or ther network equipments report its attack detection to the centralized controller.

C. SDN for Distributed Denial-of-Service (DDoS) Attack Detection and Prevention:

The DDoS attack deny lefitimate users to get access to the network devices and services. These DDoS attacks can cause a significant damage to the organization by compromising the entire network. The traditional netwroks has some kind of services to detect DDoS attacks and to protect the network, but they are not a very reliable and flexible solution. Flexible and robust solutions can be desgined due to the reconfigurable nature and programmable features of SDN. These solutions can be deployed and evaluated to detect and prevent DDoS attack.

The mobile devices have become very powerful and the usage of these mobile devices have been exponentially increasing. This increases the chances of attacks in the network. Mobile malware detection approach has been proposed, where mobile traffic from the access points is directed to the controller attached with a malware detector. Four algorithms for detecting malware are given below:

1. IP Blacklisting: A list of all suspicious IP addresses is maintained in the sytem. The OpenFlow controller verifies the IP address to see if it is from the blacklist and drops the packet if the IP address is found in the blacklist.
2. *Connection Success Ration*: The user is identified as a malicious user, if the number of unsuccessful connections of the users exceeds the fixed threshold value.
3. *Throttling Connections*: The malicious device (or host) trying to attack is identified by maintaining a list called Recently Accessed Host (RAH) list, in the system. If the waiting list of the host exceed a fixed threshold value, then that particular user is identified as a malicious user.
4. *Aggregate Analysis*: If one host in the network iss compromised by malicious activity then the security of the other systems in the network is also at risk. This algorithm works well for detecting other infected hosts in the network based on the similarities with the infected host (like the connection time, destination and single platform).

The integration of SDN for the mobile cloud infrastructure has been further explored in [101] for designing a sophisticated mechanism for protecting the network. The prime cause for the occurrence of DDoS attack in system is due to botnets. The SDN plays a major role in detecting the malicious traffic routed from OpenFlow switch and discards them to prevent further damage to the network. Many other mechanisms and solutions have been proposed for the prevention of DDoS attacks.

IV. Security Attacks in SDN and Countermeasures

As we discussed in the previous section, SDN provides security solution for the various attacks as compared to the traditional networks, through its features such as programmability. However, there are many new threats and threat vectors that arise as

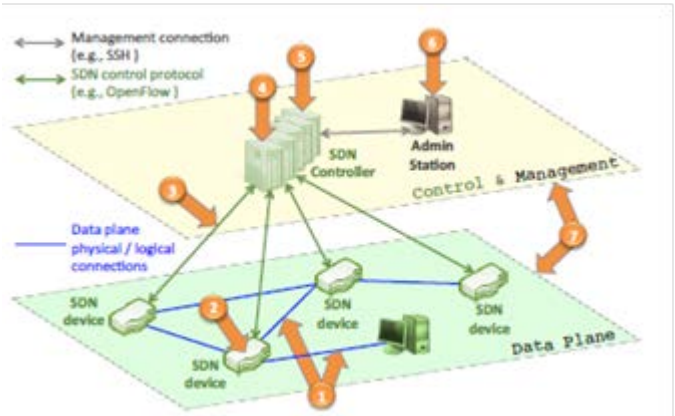

Fig. 3: SDN main threat vectors map

a result of implementation of the SDN architecture. In this section, we present security threats, challenges and countermeasures in SDN. The security vulnerabilities can ham the entire network and degrade the performance of the entire network. There are several attack vectors that arise along with the implementation of SDN. The typical attack vectors and their locations of occurrence in SDN are shown in Figure 3. The description of each of the threat vector given in Figure 3 is given below:

1. *Threat Vector 1: Forged or Faked Traffic Flows*: This threat can be triggered by faulty devices or by a malicious user. An attacker can use network elements (like the switches, servers, or personal computers) to launch a DoS attack against OpenFlow switches and controller resources.

Possible Solution: The use of IDS with support for runtime root-cause analysis could help identify abnormal flows. This could be coupled with mechanisms for dynamic control of the switch behaviour.

2. *Threat Vector 2: Attacks on Vulnerabilities in Switches*: One single switch could be used to slow down or drop the packets, clone or deviate the network traffic, or even inject traffic or forged requests to overload the controller or other neighbouring switches.

Possible Solution: The use of mechanisms of software attestation, such as autonomic trust management solutions for software components, is a possible mitigating factor. Mechanisms to monitor and detect abnormal behaviour of network devices can also be useful to detect this kind of threats

3. *Threat Vector 3: Attacks on Control Plane Communication*: It is well known in the security community that using TLS/SSL does not per se guarantee secure communication, and that compromises the controller-device link. The security of those communications is as strong as its weakest link. The attackers may be capable of aggregating enough power to launch DDoS attacks, once he gains access to the control plane. This could even enable the creation of a virtual black hole.

Possible Solution: One possible solution is the use of oligarchic trust models with multiple trust-anchor certificate authorities. Another solution is using threshold cryptography across controller replicas for securing communication.

Threat Vector 4: Attacks on Vulnerabilities in Controllers: A faulty or malicious controller could compromise an entire network. The use of a common intrusion detection system may not be enough, as it may be hard to find the exact combination of events that trigger a particular behaviour.

Possible Solution: Several techniques such as replication, employing diversity and recovery can be used. It is important to secure all sensitive elements inside the controller.

5. *Threat Vector 5: Lack of Mechanisms to Ensure Trust Between the Controller and the Management Applications*: Controllers and applications lack the ability to establish trusted relationships. The techniques that are used to certify network devices are different from those used to certify applications.

Possible Solution: Mechanisms for autonomic trust management could be used to guarantee that the application can be trusted.

6. *Threat Vector 6: Attacks on Vulnerabilities in Administrative Stations*: This threat vector is also very common in traditional networks. They are used to access the network controller. It becomes easy to reprogram the entire network from a single location.

Possible Solution: Use of protocols for requiring double credential verification. And the use of assured recovery mechanisms to guarantee a reliable state after reboot.

7. *Threat Vector 7: Lack of Trusted Resources for Forensics and Remediation*: In order to investigate and establish facts about an incident, we need reliable information from all the network components and all the domains in the network. Furthermore, this data can be used if and only if the trustworthiness of this data can be assured.

Possible Solution: Logging and tracing are the common mechanisms used. They are needed in both data and control planes. They should be indelible in order for them to be effective.

V. Conclusion

In this paper, we have discussed SDN as a security solution, as compared to the traditional networks. We have also talked about the new possible security threats and threat vectors that arise because of the implementation of SDN architecture. And also the possible countermeasures or solutions to overcome these threats.

REFERENCES

[1] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A Survey on Software-Defined Networking," IEEE Communications Surveys and Tutorials, vol. 17, pp. 27-51, 2015.

[2] D. Rawat, and S. Reddy, "Software Defined Networking Architecture, Security and Energy Efficiency: A Survey," IEEE Communications Surveys and Tutorials, vol. pp, issue. 99, pp.1-22, 2016.

[3] Kreutz et al, "Software-Defined Networking: A Comprehensive Survey," Proceedings of the IEEE, vol. 103, pp. 14-76, 2015.

[4] Astuto et al, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," IEEE Communications Surveys and Tutorials, vol. 16, pp. 1617-1634, 2014.

[5] S. Scott-Hayward, S. Natarajan and S. Sezer, "A Survey of Security in Software Defined Networks", IEEE Communications Surveys and Tutorials, vol. 18, pp. 623-654, 2016.

[6] D. Kreutz, F. Ramos, P. Verissimo, "Towards Secure and Dependable Software-Defined Networks", HotSDN, 2013.

[7] S. Ali, V. Sivaraman, A. Radford, S. Jha, "A Survey of Securing Networks Using Software Defined Netwroking", IEEE Transactions on Reliability, vol. 64, pp. 1086-1097, 2015.