SDN Security - A Survey

Shivam Pandit
Clemson University
pandit@clemson.edu

***Abstract*—Software defined networking has not just changed the way networking used to be in the past but also revolutionized the way devices interact with each other and given more power and control to the network administrators. One of the best things about SDN is that it decouples network control and data planes. Since, rate of network expansion is very rapid, their was a need to have a better control of the network along with the underlying heterogenous components. Earlier, network control used to be very complex due to components having vendor programming like Cisco and they even used to have inter communication problems. SDN solved this problem by centralizing network controller and thus achieving more agility, flexibility and end-to-end control.**

SDN enhances flow control and thus network security by means of centralizing network control which eliminates all sorts of issues that may arise between heterogenous components. Centralized network controller and decoupled data planes empowers the network to solve many networking challenges like monitoring heterogenous traffic, flow control, policy control and even network forensics. It also helps to implement network virtualization and thus providing cloud services. Thus, SDN provides many opportunities and potential to make complex powerful network systems. In this survey paper, we will discuss SDN security concepts, main techniques that are used, issues and problems in their implementation along with the future trends in this field.

I. Introduction

Software-defined networking (SDN) is one of the most happening and exciting new networking technologies that is rapidly moving from vision to reality with majority of major internet companies either already adopted SDN or in a process of migrating from conventional networks to SDN. The decoupling of data and control planes have empowered networks to function which used to be researched for a long period. It has also paved way to advanced technologies like network virtualization that gave us the cloud services which has made lives so easy.

In SDN there is one centralized control plane and other forwarding plane that will be acting based on the decisions from the control plane. So, to implement any changes to the network one has to just write software-based logic in the control plane which gets deployed to the decision logic in forwarding plane through interfaces [1]. Network operating system (NOS) does the mapping of different networks and services which are implemented in the control plane. Open Flow is the widely used SDN architecture. Open Flow applications interact with control plane through north-bound API and open flow controller interacts with data plane through OpenFlow protocol south-bound API. With OpenFlow their can be runtime traffic forward rule manipulation. With network wide policy implementation, SDN eliminates any chance of policy collision. Security monitoring application monitors the datapath and can request flow samples to check for errors and fix that through rerouting data packets. The three main layers in SDN represented in fig 1 are constituted of:

1) Application Layer: This is upper layer and contains SDN applications that communicate to controller via APIs (northbound communication) and is also responsible to set flow rules.
2) Control Plane: This is the middle layer and contains controllers that communicate via southbound protocol. OpenFlow is most widely used here to route packets using different flow rules.
3) Data Plane: This is the lower layer also called infrastructure layer and contains all the equipments that make the network like switches, router, etc.

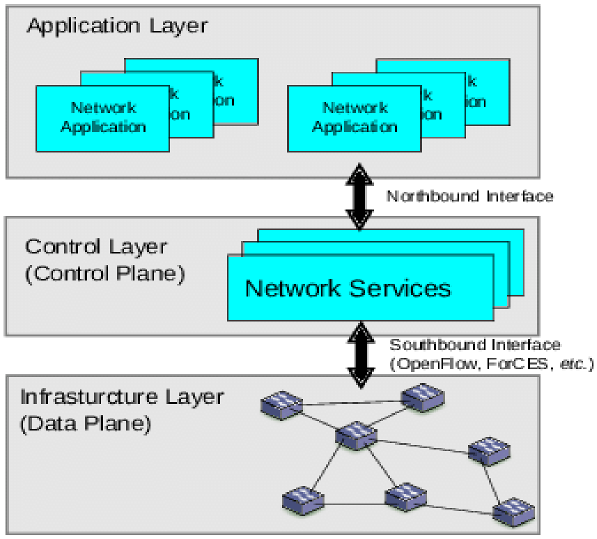

Fig. 1. A three-layer software-defined networking (SDN) architecture

Success of SDN is evident from the fact that software giants like Google, Facebook have already migrated to SDN systems. SDN has helped Google to improve efficiency of data center operations to more than 90 percent which is great. However, with the latest internet revolution, billions of devices are added on the internet through either wired or wireless mode of access and the rate of increase is very sharp which leads to the importance of security aspect. Users are now more prone to attacks as with the sophistication and advancement of technology, attacks have also become more dangerous and difficult to detect. With the advent of IOT, and more of the systems turning smart, they are more vulnerable to these attacks that can even risk human life like safety critical IOT system infected by some attack can change behavior and may even cause death of the user. Network security techniques are implemented in application layer and they get network state from control plane through north-bound interface.

OpenFlow [2] being most popular still has one major drawback that there is strict definition of the fields that make up the forwarding rules. Due to this, transition from ipv4 to ipv6 cannot be done without changes in OpenFlow. Since if hardware is using older OpenFlow version, it will fail to forward ipv6 traffic. Link Layer Discovery Protocol (LLDP) is used to do network discovery.

II. Main Techniques

In this section we briefly describe main techniques in SDN Security and how it became so popular in such a short period of time.

A. SDN Concepts and implementation

SDN helped in taking out control from individual nodes to a centralized controller which helps in better control of the network. It helps in decoupling the networking control and forwarding functions which includes routers, bridges, etc. So, we get a data plane that has all individual components that do the actual forwarding but controlled by the network controller which gets a global view of the network and can enforce flow rules, policy implementation and detection of some malicious traffic. With the technological advancement and the advent of smart home devices, global view of network was need of the hours which SDN gave as shown in the Fig 2 below which is reference architecture of SDN along with the underlying components. We can see how a security policy is implemented in application plane which redirects the traffic through control plane to its target location.

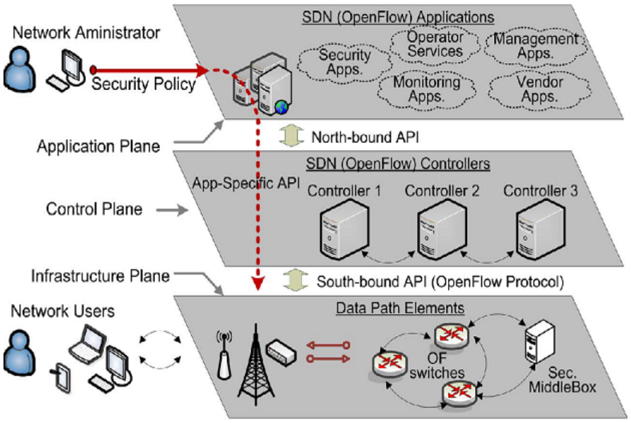

Fig. 2. SDN reference architecture showing network constituents

The separation of the network and data planes helped in implementing network abstraction where we just have to program the SDN network controller and all the logic will go down to individual nodes through southbound API's and protocols. Here, controller pushes the instruction from controller and virtually control the network. All applications work on the top of the software layer but sends all the instructions down to the components. This helps in adapting to changing business requirements very fast as all individual components don't need to be physically programmed but just the network controller. This saves a lot of time as well as abstracts the network view. Since OpenFlow is the most popular standard, we will be discussing that below along with the three planes that constitute the SDN.

1) OpenFlow: OpenFlow is fundamental to the three-tier approach of the SDN. It shifted the earlier paradigm of vendor-based interfaces to a more unified interface. OpenFlow describes the protocols and logic how controller and forwarding devices should communicate. Due to the unified view, it gets easier to fix network issues and major security problems. OpenFlow makes SDN systems feasible to be implemented in the current networks. Also, it has made possible for network managers to easily spot spoofed packets, network intrusion and also fixing vulnerabilities that may be present in the system. To implement security functions, all the security applications are deployed on top of control plane. In case of multiple network controllers, OpenFlow sets two modes of operation:

- Equal interaction: In this mode all controllers have read/write access to control switch.
- Master/Slave interaction: In this mode, their will be one master and multiple slaves.

2) Application Plane: Application plane provides platform for the application deployment that manipulate the network through control layer. It is on the top of control layer and is the place where network security systems are implemented that control all components down the infrastructure layer or data plane. Since, security exploit attacks were turned into more sophisticated, there was need of the hour to implement security across the networking components. SDN helped to solve that issue through application plane that gives full network view and protocols to deploy such systems. Even application plane is responsible for managing DNS services, firewall services, virtual network services, etc.

3) Control Plane: Control Plane in SDN is implemented in separate centralized plane away from individual nodes. SDN controller implements all the necessary control functions through NOS to form global view of network resources. API's provide the network information to the network controller which performs all flow settings. OpenFlow protocol provides standard and unified approach for controller communication with switches. The switch checks the flow tables when new packet arrives, if it finds a matching entry flow is executed or else it sends the flow to the network controller which updates the flow table for next packet flows.

4) Data Plane: Data Plane is also called infrastructure layer. This is the place where all the networking components exist which are heterogenous and include routers, bridges, gateways, etc. SDN decouples data and control layer which gives better control of the components at this layer. Network controller uses a secure communication channel to reprogram network subcomponents at this layer without physically accessing them. These remote calls can be for different purposes depending on the need. OpenFlow switches are reconfigurable and robust. Flow tables are maintained at the switches that are set by the network controller.

B. Enforcing Security Policy

There are products that enforce policies at link layer while abstracting the underlying network. One example is Sane model where any communications between server A and server B is 3 step process as mentioned below:

SDN enables us to deploy network security solutions on the top of the control layer which is propagated in entire network through southbound protocols.

- First step includes each communicating party to be authenticated with controller.
- Second step involves communicating with the controller what service A wants to request and what B wants to provide.
- Final step involves the server checks which ensures all criteria is met and establishes the service.

C. Cloud Security

Since most of the data is outsourced and PaaS offerings gaining popularity, cloud security is a major challenge. Securing them is very difficult since they are running on virtualization layer and many systems are sharing the same layer. Reaching to network packet is challenging since all packet fields are hidden. However, SDN provides cloud network security services that help in detecting intrusions and fixing vulnerabilities.

III. Issues and Problems

Every system with benefits has some drawbacks. SDN also suffers from some issues which we will be discussing now

A. Securing the Application Plane

Centralized control in SDN makes it easy to deploy new applications that implements new security services. Various networking programming languauges are implemented such as Frentic, NetCore,etc which makes it easy to implement security solutions [3]. Access Control, Security Compliance and permission control must be taken care of to prevent the network controllers from any attack that will compromise the whole network. Any flow that violates the policy must be discarded or sent to null interface. Read/Write permissions if not properly secured can be exploited to get the control of the network. This emphasizes the use of authentication and encryption methods for all sorts of communication between applications and services. Using TLS or SSH is best practice here.

B. Securing the Control Plane

Control Control Plane security is very important as it is backbone of the network which controls the entire network. Best security practices involving Linux servers is implemented here with supervision of network managers. Control Plane is to be secured from malicious applications, DDoS attacks, Scalability exploitation, etc. Control plane access should be heavily monitored to prevent unauthorized activity. Potential attackers can pretend to be SDN controllers or network administrators [4] and break into controller which is the biggest challenge to mitigate. Challenges in this plane include:

- DoS Mitigation: Using redundant controllers help us to mitigate DDoS Attacks and by analyzing flow behavior we can mitigate these attacks.
- Controller Scalability: This is very important in the time of rapid expansion of networking systems. Scalability defines load balancing among the control plane either by minimizing the load, distributing load or maximizing computing power.
- Controller Positioning: Topological study while making networks is very important to make a resilient network with lowest latency. Optimal placement of the controller is fundamental for effective implementation of SDN.

C. Securing the Data Plane

Data Plane contains all the infrastructure components that do the actual forwarding of packets using flow rules set by the controller. Generally, systems at this layer are x86 based systems using TLS or SSL protocol to secure control plane. Malicious applications at this level can modify flow rules and even gain administrative control. So, secure communications mitigate chances of eavesdropping and risk of spoofing of packets which is the responsibility of network managers to implement. FortNox is a platform that detects any flow contradictions in real-time and set the flow rules [5]. It uses digital signatures to ensure flow authorizations and restricting flow rule updations by malicious applications. A configuration verification tool called FlowChecker detects inconsistent flows and has capacity to enforce end-to-end checks. One more network debugging tool VeriFLow identifies faulty rules inserted by applications and fixes it from causing anomalous behavior.

D. Interoperability

Traditional legacy systems before SDN need to support SDN protocols for smooth transition. Making a new network based on SDN would be easy but implementing SDN on legacy systems is very difficult [6]. This is one of the reason transitions from ipv4 to ipv6 is still not done. We cannot leave out the systems for transition till everything is on the platform unless there is some closed environments like campus net, etc. This emphasizes need of interoperability. Migration from conventional systems to SDN enabled systems may require several techniques like virtualization, backward compatibility and unified protocols.

E. Ensuring Performance and flexibility

While implementing SDN systems, we must ensure high performance and high flexibility which is again a challenge. Performance means to upgrading the processing power of the systems while flexibility means capability to change the instructions. Performance upgrade has limitation of underlying network architecture and compatibility. Flexibility is also a challenge since one needs to have full knowledge of hardware before he can program it and set rules to take full advantage of computing power.

F. Protecting SDN controller

Since SDN controller virtually controls full network, any malicious attack on the controller will expose whole network. This emphasizes use of best security practices at SDN controller to ensure it never gets hacked or intruded by some hacker pretending to be SDN controller. Biggest threat to SDN controller is DDoS attacks that may result in server getting down and thus network collapse. Their must be rules set to monitor and filter traffic that reaches SDN controller and using SSL or TLS protocols to ensure communication is secured and not open to sniffing. Mutual authentication between SDN controller and switches may help in overcoming any intruder to get in the communication channel. However, with multiple controllers working in sync in bigger networks, securing communication turns even more complex.

IV. Future Trends

SDN marketplace is rapidly growing and need of close to 100% utilization of network is the challenge. SDN has immense benefits with some security challenges. Today, we are moving towards agile development which require scaling, high performance and high availability. SDN is key to achieve all these. Still many companies have reluctance in adopting it. SDN enabled networks are fast, easy to maintain, cheaper and even more flexible [7]. Till now just about 30% organizations have successfully migrated to SDN which include software giants like Google. Currently, only 30-40% network utilization is a standard while as Google is about to hit 100% utilization ratio in managing traffic flow in its internal network. SDN's operational efficiency can even help corporates to increase the pace of application delivery. Collaboration is very important to successfully migrate legacy systems into SDN systems. Different variations of SDN architecture are gaining prominence especially software-defined-WAN. There is significant upward trend in the future of SDN which is evident from below factors:

- Increasing number of private clouds.
- Heavily automating the network to decrease manpower utilized to monitor and operate the traffic.
- More deployment of virtual environments.
- Adding flexibility to the networks.
- More emphasis on centralized control.

Even traditionally closed companies that were earlier hesitant to SDN are slowly migrating to it. One of the latest entrants is AT&T which is working on SDN code and also the Microsoft that has implemented SDN on its Azure environment. Passionate contributors will further contribute towards making migrating to SDN possible.

References

[1] Sezer, Sakir, et al. "Are we ready for SDN? Implementation challenges for software-defined networks." IEEE Communications Magazine 51.7 (2013): 36-43.

[2] Flauzac, Olivier, et al. "SDN based architecture for IoT and improvement of the security." Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on. IEEE, 2015.

[3] Scott-Hayward, Sandra, Gemma O'Callaghan, and Sakir Sezer. "SDN security: A survey." Future Networks and Services (SDN4FNS), 2013 IEEE SDN For. IEEE, 2013.

[4] Cabaj, K., Wytrebowicz, J., Kuklinski, S., Radziszewski, P., & Dinh, K. T. (2014, September). SDN Architecture Impact on Network Security. In FedCSIS position papers (pp. 143-148).

[5] Scott-Hayward, S., O'Callaghan, G., & Sezer, S. (2013, November). SDN security: A survey. In Future Networks and Services (SDN4FNS), 2013 IEEE SDN For (pp. 1-7). IEEE.

[6] Networkworld.com, Software defined Networking, https://www.networkworld.com/category/software-defined-networking

[7] Searchsdn.techtarget.com, The future of SDN in 2018 moves toward increased automation, https://searchsdn.techtarget.com/tip/The-future-of-SDN-in-2018-moves-toward-increased-automation