IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 5, May 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305310 |

Overview on Software-defined Networking (SDN) Security

Mayur Bhagat, Pushpa Tandekar, Ashish Deharkar

Student, Department of Computer Science & Engineering, Shri Sai College of & Engineering & Technology, Chandrapur, India

Assistant Professor, Department of Computer Science & Engineering, Shri Sai College of & Engineering & Technology, Chandrapur, India

Assistant Professor, Department of Computer Science & Engineering, Shri Sai College of & Engineering & Technology, Chandrapur, India

ABSTRACT: Software-Defined Network (SDN) radically changes the network architecture by decoupling the network logic from the underlying forwarding devices. This architectural change rejuvenates the network-layer granting centralized management and re-programmability of the networks. From a security perspective, SDN separates security concerns into control and data plane, and this architectural recomposition brings up exciting opportunities and challenges**.** However**,** till today, SDN has not been well recognized by the security community yet. In this systematic survey on SDN security, we investigate how the new features provided by SDN can helpenhance network security and information security process. By systematically reasoning the opportunities introduced by SDN to network security, we hope to provide new insights for futureresearch in this important area.

KEYWORDS: Software-Defined Network (SDN), SDN Security, Data Plane Security, Security countermeasures.

I. INTRODUCTION

Traditional IP network devices are purpose-built and application-specific with integrated circuits and chips designed to achieve high throughputs. This 'hardware- centric' network model requires network operators to configure each device separately through low-level vendor- specific commands. Hence, in heterogeneous networks, network configuration is tedious, and automatic recon- figuration and response are virtually impossible. More- over, the data and control plane bundling reduces flexibility, hinders innovation and slows down the evolution of networking infrastructure. In fact, as shown by recent studies, in the long run, traditional networks are inca- pable of coping with the increasing demand and continuous expansion in the number of devices and applications brought through advances in Cloud Computing, Internet-of-Things (IoT), and Cyber-Physical Systems [43, 62, 71, 77).

Software-Defined Network (SDN) is a new network paradigm that decouples the control from data plane in networking devices. This architectural recomposition places the 'brain' of the network on a specialized central controller, enabling centralized management and global view of the network. The data plane is composed of 'dummy' devices, forwarding packets based on rules specified remotely. These rules may be specified by the application running atop of the controllerand triggered according to packet-level extracted information.

The reason why many researchers and practitioners have interests in SDN is mainly because by decoupling the control logic from the closed, proprietary implementations of traditional network switch infrastructure, SDN enables us to design and distribute innovative flow handling and network control algorithms easily, and it helps us add much more intelligence and flexibility to the control plane. With the help of SDN, we can dynamically control network flows and monitor network status easily. For example, by employing SDN, we can easily implement a network load balancing function that is not easily and cheaply solved with existing techniques. These powerfuland rich functions from SDN enable people to create new and creative network services or architectures.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305310 |

II. METHODOLOGY

Overview of SDN architecture

SDN is a kind of emerging network architecture which decou- ples data forwarding from the control logic; currently, these aspects are tightly integrated in traditional network equipment,such as switches and routers. The decoupling of data forwarding and the control logic enables the network control and applications to be programmable [16], Generally, SDN architecture can be divided into three layers, respectively called the data forwarding layer, the control layer and the application layer from bottom to up.

Data forwarding layer

The data forwarding layer consists of many SDN switches, which are physically connected by wired or wireless media. Each switch is a simple device in charge of forwarding network packets and has a forwarding table, named the Flow Table, which contains thousands of rules that are used to formulate forwarding decisions.

Each rule item in the Flow Table is made up of three fields: the action, the counter and a pattern. The pattern field defines the flow pattern, which is basically the set of header field values of the packet. When data packets are received, the switch will search its Flow Table to find a rule that matches the fields.

Once the switch finds such a rule, the counter of the rule increases, and the action corresponding to the particular rule will be performed. Otherwise, the switch will notify the controller to requestfor help or simply discard the packet It is worth noting that forwarding rule items are not generated by the switch node itself, but are pushed down by the controller from the control layer

Control layer

As the SDN's brain, the control layer man mages and controls the entire network. We refer to the network node that implements these functionalities as the SDN controller, and it is generally deployed as a separate physical device with specific software. The SDN controller communicates with the switch through a standard south-bound API, e.g., Open Flow, and has a global view of the entire network topology at the data forwarding layer, i e., switches and links. Various routing protocols, such as BGP and OSPF, run on the SDN controller so that all the data forwarding taking place in the data layer is based on instructions placed by the controller.

As the de factor standard of SDN. Open Flow was initially designed with a single controller for simplicity, which constitutes however a potential single port of failure. Therefore, almost all recent SDN architecture implementations, such as Floodlight [17], NOX [18] and Open Daylight [19], support multiple distributed controllers, which improve the scalability and availability of network resources. In the multi controller architecture, each individual controller is responsible for controlling only a portion of the switches. In order to maintain the consistency of thenetwork's status and work collaboratively, an individual SDN controller can communicate with other controllers in the network through east-west- bound APIs, as discussed in [20].

Application layer

The application layer allows network operators to respond rapidly to the various innovative application software has been bust to function en sup controllers top of SDN that various application requirements met [21], such as network virtualization [22], topology discovery [23], traffic monitoring [24], security enhancement [25] load balancing [26] and others.

The application layer communicates with the control layer through north-bound APIs, such asthe REST APL The control layer provides an abstraction of the net work's physical resources for the application layer, which means that network operators can change the data paths of packets using only software programming centrally on the SDN controllers, and not configure all the physical switches in the date path time by one.

SDN simplifies the network design and operation. It also simplifies the network devices as the devices no longer need to understand and process protocols but simply accept instructions from SDN controller. Salient features of Distinguishing features of SDN are discussed here:

A. Logically centralized intelligence

In SDN model, network control functionality is entirely distributed from forwarding using a standard southbound interface viz., Open Flow. By centralizing entire network intelligence, decision making is based on a global view of the

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305310 |

network. This is opposed to today's prevailing networks. Today's networks are built on an autonomous system view and nodes are not aware about the overall state of the entire network.

B. Abstraction

In such network, business applications consuming SDN services are abstracted from various underlying network technologies. The network software is resident in the Control Layer. To ensure portability and future proofing of investments in network services, network devices are abstracted from the control layer.

C. Programmability

Software Defined Networks are controlled through software functionality. These software are provided by different vendors or the network operators. This kind of programmability makes the management paradigm to be completely replaced by automation. This is influenced by rapid adoption of the cloud By making available open Application programming interfaces (APIs) for various applications to interact with network, such networks can achieve innovation and differentiation.

D. Increased network reliability and security

SDN makes IT to define high level configuration and various policy statements. Which in turn are translated to the infrastructure Such network architecture completely eliminates requirement of individual configuration of various network devices every time an end point, service, or application is either added, moved, or a policy changes. This reduces the likelihood of network failures because of policy Inconsistencies and configuration.

Security analysis of SDN architecture

Now, we will look into the SDN architecture's characteristics' impact on security. Compared to traditional network architectures, security threats of SDN will be even more concentrated as opposed to the dispersion seen in the network elements of traditional networks. Therefore, because of its design nature, SDN has security advantages and security defects. Its advantages include:

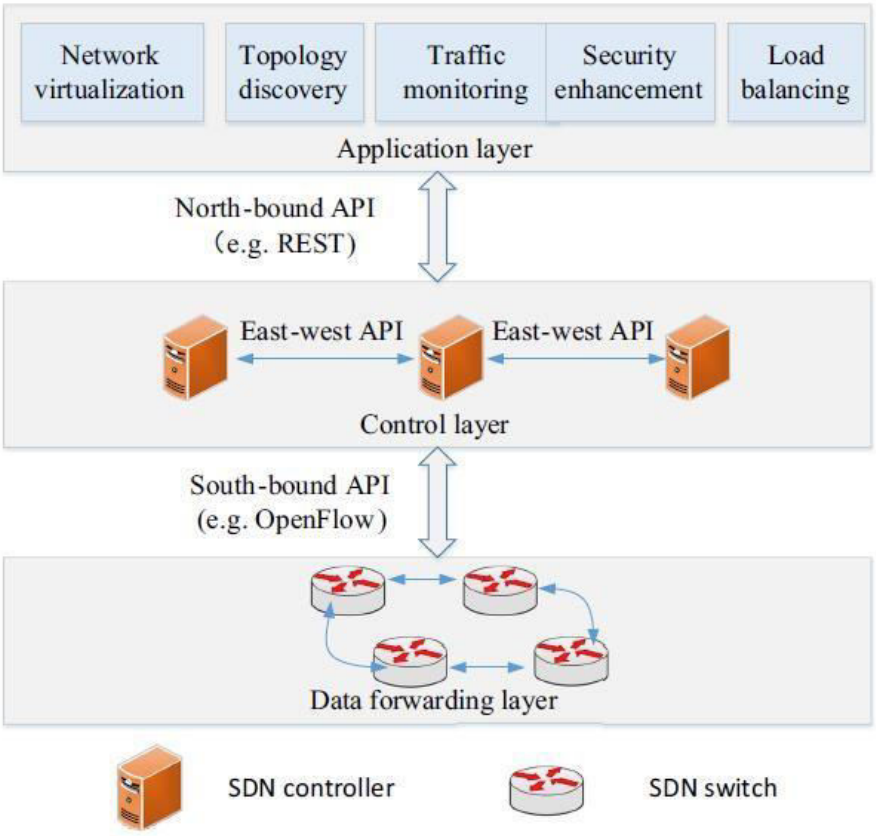
1. Effective monitoring of abnormal traffic. Due to the fact that the SDN controller can perceive the entire network traffic simultaneously, it is easier to notice abnormal behaviorin network traffic caused by an attacker.

2. Timely dealing with vulnerabilities. Another important advantage can be attributed to the nature of the programmable network environment. Once a new threat has been detected, operation can program new software to analyze and deal with the vulnerability immediately, without spending time to wait for an update of the operating system or application software integrated in the manufacturer-proprietary devices. In addition, the SDN controller can achieve a security policy configuration covering 2-7 layers of the Open Systems Interconnection (OSI) architecture, and provide more granular security control

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305310 |



III. CONCLUSION

Represented taxonomy of attacks against Software-Defined Networks based on their scope and impacts. Specifically, we discussed how an adversary can leverage vulnerabilities of different SDN components to target network policies, their enforcement, and implementation. Moreover, we argued the importance of security SDN data planes, surveyed the latest advances in data plane development and analyzed the security implications of stateful data planes. Thereafter, we established a set of requirements for a working solution and reviewed the existing solutions with respect to these requirements. Finally, we provided a set of suggestions for future research directions, where we hope to attract researchers' attention to a relatively dormant yet critical aspect of securing SDNs.

We hope this study can not only provide a quick introduction and systematic survey but also givesignificant insights for using SDN for better security applications and stimulate more future research in this important area.

REFERENCES

1. Chen M, Zhang Y, Li Y, Mao S, Leung V (2015) EMC: emotionaware mobile cloud computing in 5G. IEEE Netw 29(2):32–38
2. Wan J, Yan H, Suo H, Li F (2011) Advances in cyber-physical systems research. KSII Trans Internet Inf Syst 5(11):1891–1908
3. Suo H, Liu Z,Wan J, ZhouK (2013) Security and privacy inmobile cloud computing. In: Proceedings of the 9th IEEE International Wireless Communications and Mobile Computing Conference,
4. Cagliari, Italy
5. Cisco Inc. (2013) Software-defined networking: why we like it andhow we are building on it. White Paper
6. McKeown N, Anderson T, Balakrishnan H, Parulkar G, Peterson L, Rexford J, Turner J (2008) OpenFlow: enabling innovation in campus networks. ACMSIGCOMM Comput Commun Rev 38(2):69–
7. 74
8. Liu J, Li Y, Chen M, Dong W, Jin D (2015) Software-definedinternet of things for smart urban sensing. IEEE Commun Mag 53(9):55–63
9. Hong CY, Kandula S, Mahajan R, Zhang M, Gill V, Nanduri M, Wattenhofer R (2013) Achieving high utilization with softwaredrivenWAN. ACM SIGCOMM Comput Commun Rev 43(4):

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305310 |

10. 15–26
11. Google Inc. (2012) Inter-datacenter WAN with centralized TEusing SDN and OpenFlow. Open Network Submit
12. Jain S, Kumar A, Mandal S, Ong J, Poutievski L, Singh A, VenkataS,Wanderer J, Zhou J, Zhou M, Zolia J, Hölzle U, Stuart S, Vahdat
13. A (2013) B4: experience with a globally-deployed software definedWAN. In: Proceedings of the ACM SIGCOMM, pp 3–14
14. VMware NSX. [Online] http://www.vmware.com/products/nsx/
15. Nuage Networks VSP. [Online] http://www.nuagenetworks.net/products/virtualized-services-platform/
16. Ahmad I, Namal S, Ylianttila M, Gurtov A (2015) Security in software defined networks: a survey. IEEE Commun SurvTutorials 17(4):2317–2346
17. Seyed K. Fayaz, Yoshiaki Tobioka, Vyas Sekar, and Michael Bailey. Bohatei: Flexible and Elastic DDoS Defense. In Usenix Security'15, 2015.
18. S. K. Fayazbakhsh, L. Chiang, V. Sekar, M. Yu, and J. C. Mogul.
19. Enforcing network-wide policies in the presence of dynamicmiddlebox actions using FlowTags. In NSDI'14, 2014.
20. Nate Foster, Michael J. Freedman, Rob Harrison, Jennifer Rexford, Matthew L. Meola, and David Walker. Frenetic: a high-level language for OpenFlow networks. In Proceedings of
21. the Workshop on Programmable Routers for Extensible Servicesof Tomorrow, 2010.
22. Aaron Gember-Jacobson, Raajay Viswanathan, ChaithanPrakash, Robert Grandl, Junaid Khalid, Sourav Das, and Aditya Akella. OpenNF: Enabling Innovation in Network Function Control. In SIGCOMM'13, 2014.
23. Gigaom. Why Network Virtualization is Important. http://gigaom.com/2009/02/02/why-network-virtualizationis-important/.
24. Stephen Gutz, Alec Story, Cole Schlesinger, and Nate Foster. Splendid Isolation: A Slice Abstraction for Software-Defined
25. Network. In HotSDN'12, 2012.
26. Brandon Heller, Srini Seetharaman, Priya Mahadevan, Yiannis Yiakoumis, Puneet Sharma, Sujata Banerjee, and Nick McKeown. ElasticTree: Saving Energy in Data Center Networks.
27. In Proceedings of the 7th USENIX Symposium on NetworkedSystems Design and Implementation, NSDI, 2010.
28. Sungmin Hong, Robert Baykov, Lei Xu, Srinath Nadimpalli, and Guofei Gu. Towards SDN-Defined Programmable BYOD (Bring Your Own Device) Security. In NDSS'16, 2016.
29. Sungmin Hong, Lei Xu, Haopei Wang, and Guofei Gu. Poisoning Network Visibility in Software-Defined Networks: New
30. Attacks and Countermeasures. In NDSS'15, 2015.
31. Sotiris Ioannidis, Angelos D. Keromytis, Steve M. Bellovin, and Jonathan M. Smith. Implementing a distributed firewall.In Proceedings of the 7th ACM conference on Computer andcommunications security, 2000.
32. Jafar Haadi Jafarian, Ehab Al-Shaer, and Qi Duan. Openflow Random Host Mutation: Transparent Moving Target Defense using Software Defined Networking. In Proceedings of the
33. First Workshop on Hot Topics in Software Defined Networks,HotSDN '12, 2012.
34. Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer,Junlan Zhou, Min Zhu, Jon Zolla, Urs H¨olzle, Stephen Stuart, and Amin Vahdat. B4: Experience with a Globally-deployed Software Defined Wan. In Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM, 2013.
35. Min Suk Kang, Virgil D. Gligor, and Vyas Sekar. SPIFFY:
36. Inducing Cost-Detectability Tradeoffs for Persistent Link-Flooding Attacks. In NDSS'16, 2016.
37. Lowlesh Yadav and Asha Ambhaikar, "IOHT based Tele-Healthcare Support System for Feasibility and perfor-mance analysis," Journal of Electrical Systems, vol. 20, no. 3s, pp. 844–850, Apr. 2024, doi: 10.52783/jes.1382.
38. L. Yadav and A. Ambhaikar, "Feasibility and Deployment Challenges of Data Analysis inTele-Healthcare System," 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), Raipur, India, 2023, pp. 1-5, doi: 10.1109/ICAIIHI57871.2023.10489389.
39. L. Yadav and A. Ambhaikar, "Approach Towards Development of Portable Multi-ModelTele-Healthcare System," 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), Raipur, India, 2023, pp. 1-6, doi: 10.1109/ICAIIHI57871.2023.10489468.
40. Lowlesh Yadav and Asha Ambhaikar, Exploring Portable Multi-Modal Telehealth Solutions: A Development Approach. International Journal on Recent and Innovation Trends inComputing and Communication (IJRITCC), vol. 11, no. 10, pp. 873–879, Mar. 2024.11(10), 873–879, DOI: 10.13140/RG.2.2.15400.99846.
41. Lowlesh Yadav, Predictive Acknowledgement using TRE System to reduce cost and Bandwidth, March 2019. International Journal of Research in Electronics and Computer Engineering (IJRECE), VOL. 7 ISSUE 1 (JANUARY-MARCH 2019) ISSN: 2393-9028 (PRINT) | ISSN: 2348-2281 (ONLINE).

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details