DOI: 10.2507/35th.daaam.proceedings.xxx

SDN AND NETWORK SECURITY

Adriano Bubnjek, Damir Regvart*, Karlo Josic

This Publication has to be referred as: Bubnjek, A[driano]; Regvart, D[amir] & Josic, K[arlo] (2024). SDN and Network Security, Proceedings of the 35th DAAAM International Symposium, pp.xxxx-xxxx, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-xx-x, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/35th.daaam.proceedings.xxx

Abstract

This paper explores Software Defined Networking (SDN), detailing its architecture, components, and advantages over traditional network infrastructures. It also examines the security vulnerabilities inherent in SDN systems and strategies for their mitigation. The initial section of the paper covers SDN's essential features, such as scalability, enhanced visibility, and centralized control. It discusses its capability to operate independently of hardware while still utilizing it for traffic forwarding. The advantages and limitations of the SDN model are also analyzed. Subsequently, the paper categorizes various SDN security threats and attacks based on its unique architecture. It outlines preventive measures and responses to potential cyberattacks, including DDoS, ARP spoofing, and resource depletion.

Keywords: Software-Defined Network (SDN); network security architecture, network virtualization.

1. Introduction

Software Defined Networking (SDN) is an approach that allows network engineers and others to control underlying hardware infrastructure, which spans data centres, both private and public clouds, and application frameworks, using software-defined controllers or APIs (Application Programming Interfaces) from a centralized console or platform [2]. Essentially, SDN offers a programming-centric way to configure network and data traffic flow [4]. One of the notable advantages of SDN is enhanced network security [3], a primary focus of this seminar. Given the inherent complexity and layered nature of SDN, which is a critical component in many of today's enterprise infrastructures, this paper will briefly touch on certain SDN features and concepts without going into extensive detail. A fundamental distinction between SDN and traditional hardware-based networking is that SDN's control plane makes infrastructures more scalable, flexible, and manageable [1]. Furthermore, SDN differs significantly from conventional networking in terms of security implementations. This is partly due to its compatibility with Network Function Virtualization (NFV), which separates essential network security functions such as firewalls and load balancers (WAF) from dedicated hardware, instead hosting them in a software-defined instance that secures virtualized, containerized, or bare metal workloads. This paper will explore SDN architecture, its components, and core features in subsequent sections. It will also discuss how businesses can benefit from adopting a software-defined model, consider important security principles during the design phase, and address common threats, attacks, and vulnerabilities associated with SDN.

2. SDN Architecture and Key Features

As introduced earlier in this paper, Software Defined Networking (SDN) is a forward-thinking approach that leverages virtualization to separate the primary components and roles typically housed within traditional networking devices such as switches and routers [2]. Below is a figure depicting the SDN architecture and some of its integral components.

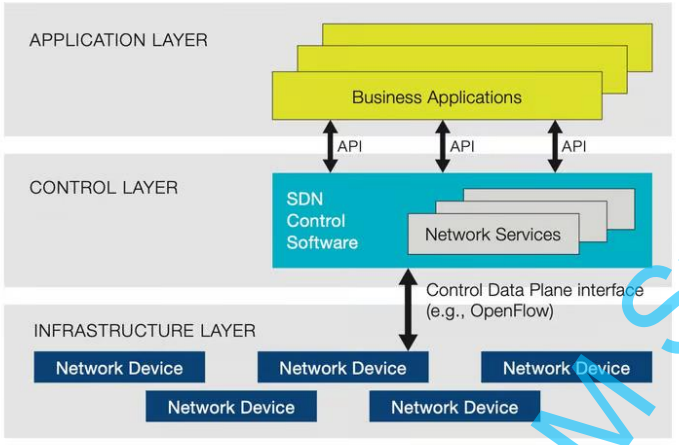

Fig. 1. Components and Architecture of SDN, https://www.sdxcentral.com/networking/sdn/definitions/what-the-definition-of-software-defined-networking-sdn/, accessed on 2024-05-05.

One of the principal distinctions in SDN is the division between the controller or control plane and the data plane [3]. The control plane creates network topologies or tables and sets packet handling policies. In contrast, the data plane handles the intensive tasks of forwarding packets and segmenting and reassembling data. These are standard functions within any conventional networking device. However, such a distributed networking framework does not scale well and lacks the speed required to keep pace with the rapid expansion of business development across various industries. Moreover, modifying or enforcing existing security policies within this traditional model is time-consuming, especially without third-party automation solutions like Ansible, Puppet, and Terraform, which may be excluded due to complexity or skill deficits [4]. SDN addresses these issues by centralizing the brain of the operation, the control plane, typically within a console or interface. In contrast, the data plane performs the primary network device functions. Other SDN components include application planes that support routing, load balancers, firewalls, etc., Northbound and Southbound APIs which facilitate connections between the SDN application layer, SDN controller, and networking devices, and OpenFlow, a protocol used to manage the southbound interface connecting the controller to network devices [4].

The SDN architecture is segmented into three layers: the application layer, the controller layer, and the networking devices layer [2]. The application layer contains programs communicating with the controller via API calls, such as load balancers and firewalls. The controller serves as the centralized element of SDN technology, acting as middleware between the SDN application and networking devices layers, facilitating data transfer between them. Lastly, the networking devices layer includes the hardware that hosts the data plane component. One might question the rationale and benefits of integrating such a sophisticated and costly model into an existing environment that seemingly meets all current business application and operational needs. A major issue in IT today is documentation, particularly of the network layer and its topology. All may seem well until a problem or legitimate threat arises. At this point, engineers often spend significant time manually investigating their own or their client's environments before they can even begin to address the issue. This process detracts from their primary responsibilities. This challenge becomes more manageable over time within the same company or with the same clients due to familiarity with the environment. However, this is neither sustainable nor scalable, particularly in large enterprises experiencing frequent changes [3].

In the below figure, Fig. 2., the SDN model proves its value, especially regarding networking. One of the most powerful features of a robust SDN model is its visibility and the ability to automatically generate network topologies from a centralized graphical console interface [4]. While this may not seem groundbreaking, it is a fundamental feature expected in any high-quality SDN product, such as NSX-T from VMware, ACI, and Meraki by Cisco, Dell SDN, Juniper SDN, and others. It is not uncommon for companies to rely on or complement their documentation with topologies created by SDN.

Beyond visibility and scalability, SDN facilitates the easy implementation of Quality of Service (QoS) for VoIP or video streaming, optimizes applications and services based on network metrics collected by SDN, reduces operational costs by minimizing the time required to troubleshoot and resolve issues, abstracts cloud resources, and simplifies the enforcement and implementation of security policies and models, which will be discussed in the following chapter. In

summary, SDN offers significant advantages for organizations seeking flexibility, scalability, visibility, modernity, and security. However, SDN is not without its drawbacks and limitations.

Fig. 2. Network Topology of vSphere with Tanzu Created by NSX-T

Its complexity necessitates reconfiguring existing network infrastructure, a daunting task that could span years depending on the size of the infrastructure and its business impact. Additionally, SDN demands highly trained and skilled employees due to its steep learning curve, which can prolong the implementation phase and complicate system maintenance once completed. This often leads companies to outsource, potentially resulting in vendor lock-ins and dissatisfaction among existing staff. Moreover, certain security limitations within SDN may be exploited by attackers, a topic that will be explored in the next chapter.

3. SDN Architecture and Component Attacks

When discussing attacks and vulnerabilities in Software Defined Networking (SDN), they can be broadly categorized into two groups: those targeting the SDN architecture and those targeting its components.

Attacks on the SDN architecture typically focus on the application, control, and data plane layers, along with their respective interfaces [5]. In the application layer and its interfaces, three primary categories of attacks are prevalent: unauthorized access, malicious applications, and configuration issues. Specific examples within these categories include the lack of TLS implementation, insecure provisioning, and unauthorized application attacks. The security challenges posed by the separation of control and data planes in SDN architectures and the necessity of integrating security measures at each layer [5]. The control layer is susceptible to additional categories of attacks, such as data leakage, modification, denial of service, and broader system-level security issues. Examples include communication flooding between controllers and network devices and controller hijacking, which can have severe consequences [6]. The attacks on the data layer and its interfaces mirror those on the control layer, except for malicious applications, as the data layer does not directly interact with the application layer. Furthermore, attacks targeting SDN components are classified based on the affected component (Data Plane, Control Plane, Application Plane, Northbound API, Southbound API). The Application Plane is vulnerable to four major types of attacks: storage attacks, control message attacks, resource attacks, and access control attacks. There are techniques for enhancing the security of southbound infrastructure in SDN, which supports the need for robust security protocols and monitoring systems that could prevent such attacks [6]. A successful storage attack can lead to data corruption or theft from the shared storage used by SDN applications. Control message attacks can disrupt or manipulate communications between controllers and networking devices, affecting traffic management. Resource attacks aim to deplete memory or CPU resources on applications and controller components, causing disruption and performance degradation. Lastly, access control attacks target the controller, often due to misconfigured authentication, authorization, and accountability settings.

Three attack types are noted regarding the Data Plane: device, protocol, and side-channel attacks. Device attacks search for vulnerabilities in the software or firmware of SDN switches or routers. The utilization of multi-site solutions to mitigate potential network attacks offers practical approaches to securing SDN environments that could counter device attacks [7]. Protocol attacks target weaknesses in the OpenFlow protocol used to manage network traffic. In contrast, side-channel attacks analyze CPU consumption, power usage, and traffic flow to deduce patterns that may reveal information about specific services or applications [5]. For the Control Plane, potential attacks include manipulation,

availability, and software hack attacks. Manipulation attacks target the controller's algorithms and understanding of the Data Plane, potentially leading to erroneous decisions [5]. Availability attacks aim to temporarily disable the controller, isolating parts of, or the entire, network through methods like resource depletion, flooding, and specific network traffic congestion [5]. Software hack attacks compromise the underlying system on which the controller operates, with lesser-known or non-enterprise-grade operating systems being particularly vulnerable. A security-hardened, enterprise-grade OS like Photon OS can significantly reduce these risks. Northbound and Southbound APIs face three similar attack types: interception, eavesdropping, and availability attacks. For example, availability attacks might involve denial of service through excessive traffic; interception attacks could manipulate the flow table, resulting in incorrect packet forwarding; and eavesdropping attacks might involve control message spoofing, potentially leading to unauthorized control over network devices.

4. SDN Security Measures and Mitigations

Before delving into specific mitigation strategies for attacks, it's crucial to outline some general principles and security measures for hardening SDN protocols, components, and interfaces to minimize threats. A fundamental and critical security measure is the establishment of a robust framework for authentication and authorization [8]. This framework hinders unauthorized access to your SDN infrastructure and restricts internal access to approved entities. Additionally, implementing comprehensive auditing policies is vital, not only for compliance with standards like ISO 27001 but also for the overarching benefit of the organization. There should also be an effective method for evaluating new controls and their impact on the network's confidentiality, integrity, and availability. Moreover, a defense-in-depth strategy should be adopted to deter attackers and protect the system from various angles.

Specific security measures can be tailored according to the SDN layer involved. For the Application Plane, security can be enhanced by integrating protective measures directly into applications to monitor and actively defend against threats. This proactive security approach is crucial for managing potential security incidents. Proper authentication and encryption must be established to secure communications between the application and controller [8]. Additionally, publicly exposed web applications should be safeguarded with tools such as Web Application Firewalls (WAFs), standard firewalls, and proxies. Implementing TLS to encrypt traffic between networking devices and the SDN controller for the Data Plane is essential for preventing spoofing and eavesdropping. Remote access connections should use SSH instead of Telnet, and SNMPv3 should replace SNMPv2 for secure network device management over IP. Furthermore, tunnel endpoints must be secured with strong passwords. The SDN controller should be configured for high availability to eliminate potential single points of failure caused by attackers or system failures [9]. For example, deploying at least three controller instances across three virtualization hosts in a geographically distributed cluster can enhance system resilience. Role-Based Access Control (RBAC) policies must be enforced to ensure that only authorized personnel can make configuration changes. Additionally, logging and auditing mechanisms should be in place to detect unauthorized access and modifications. The following table outlines common attacks on SDN and corresponding mitigation steps:

Attack	Mitigation Steps
Password guessing/brute force	Enforce frequent password changes, use complex and multi-language passwords with special characters, change default settings, and avoid reusing passwords [8].
API exploitation	Regularly update and patch servers and SDN components.
Traffic sniffing	Encrypt traffic using robust protocols (e.g., TLS instead of SSL).
Side channel attack	Utilize robust encryption algorithms like AES and ECC to secure SDN components and network elements.
ARP spoofing	Implement network segmentation (VLAN), Dynamic ARP Inspection (DAI), and switch port security.
Network manipulation attack	Design a redundant SDN Control Plane layer with multiple controller instances and secure communications.
Misconfiguration attack	Implement RBAC, automated auditing, continuous monitoring with intrusion alerts, and automation tools for repetitive tasks. Provide security training and adopt a zero-trust networking approach.
DDoS attack	Use a high-availability (HA) controller architecture, rate limits, and packet-dropping techniques at the control plane, and monitor network traffic with an SDN-distributed firewall [9].
Resource attack	Monitor system metrics and implement rate limits and resource quotas where possible.
DNS spoofing	Implement DNSSEC to secure DNS communications.
Rogue SDN controller	Establish a robust authentication framework, use strong encryption for component communications, and implement RBAC, continuous auditing, and Multi-Factor Authentication (MFA).

Table 1. SDN Attacks and Mitigations

Although the strategies above aim to secure SDN from various threats and minimize the likelihood of attacks, complete protection against cyberattacks is not feasible. Therefore, having a mitigation strategy and countermeasures ready for well-known attacks is imperative.

5. Future Works

One significant area for future research involves the integration of artificial intelligence (AI) and machine learning (ML) into SDN. These technologies could revolutionize how networks predict, detect, and respond to anomalies and security threats. Exploring how AI can optimize network traffic flow and security protocols could lead to more autonomous network systems that reduce human error and enhance security efficacy. This advancement could usher in a new era of intelligent networking where systems are capable of self-management and real-time security enforcement. Another promising research trajectory is the development of advanced security protocols tailored explicitly for SDN environments. This includes creating dynamic security policies that adapt in real-time to emerging threats. Leveraging the centralized control inherent in SDN to implement rapid security updates across the network could significantly enhance the responsiveness and adaptability of network defenses. Additionally, as quantum computing poses new risks to traditional encryption methods, future studies should consider the implications of quantum technologies on SDN security, particularly concerning data encryption and secure communications. The role of SDN in emerging networking paradigms such as 5G, Internet of Things (IoT), and edge computing offers ground for research. These areas, characterized by their vast scale and complexity, could greatly benefit from SDN's capabilities in managing large arrays of devices and data flows. Investigating how SDN can provide dynamic, efficient solutions to manage and secure these expanding networks will be crucial as these technologies continue to grow and intersect.

6. Conclusion

This paper has thoroughly explored the multifaceted domain of Software Defined Networking (SDN), delineating its architecture, components, and intrinsic advantages over traditional network frameworks. The paper highlighted SDN's transformative potential in modern network management and security by unpacking critical features such as enhanced scalability, visibility, and centralized control. A focal point of the paper was the detailed examination of SDN's security landscape, where various threats and vulnerabilities specific to its unique architecture were categorized and analyzed. The paper highlighted prevalent security risks such as DDoS attacks, ARP spoofing, and resource depletion and articulated strategic mitigation measures to fortify SDN environments against such threats. Emphasizing the importance of robust authentication, authorization frameworks, and comprehensive auditing protocols, the paper advocated for a proactive security posture that aligns with contemporary compliance and operational standards.

Looking beyond the current applications of SDN, the paper proposed several promising directions for future research. Integrating artificial intelligence and machine learning is a revolutionary next step, potentially enabling more autonomous and dynamically secure networks. The call for enhanced interoperability and the development of standardized security protocols tailored to SDN environments further indicates this technology's ongoing evolution and maturation. Moreover, exploring SDN's applicability in burgeoning fields like IoT and 5G illustrates its pivotal role in the next generation of network technology.

7. References

[1] Jimenez, M. B. & Fernandez, D. (2022). A Framework for SDN Forensic Readiness and Cybersecurity Incident Response, Proceedings of the 2022 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Published by IEEE, pp. 112-116, ISBN 978-1-6654-7334-7, Phoenix, USA. DOI: 10.1109/NFV-SDN56302.2022.9974648, in press.

[2] Tsai, C. & Song, K. (2023). Discussion on Network Security under SDN Architecture, Proceedings of the 2023 26th ACIS International Winter Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Winter), Published by IEEE, pp. 53-57., ISBN 979-8-3503-4586-5, Taiyuan, China. DOI: 10.1109/SNPD-Winter57765.2023.10223982, in press.

[3] Liang, H.; Liu, H.; Dang, F.; Yan, L. & Li, D. (2021). Information System Security Protection Based on SDN Technology in Cloud Computing Environment, Proceedings of the 2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), Published by IEEE, pp. 432-435., ISBN 978-1-6654-3561-1, Dalian, China. DOI: 10.1109/AEECA52519.2021.9574276, in press.

[4] Ahmad, A.; Harjula, E.; Ylianttila, M. & Ahmad, I. (2020). Evaluation of Machine Learning Techniques for Security in SDN, Proceedings of the 2020 IEEE Globecom Workshops (GC Wkshps), Published by IEEE, pp. 1-6., ISBN 978-1-7281-7307-8, Taipei, Taiwan. DOI: 10.1109/GCWkshps50303.2020.9367477, in press.

[5] Hatim, J.; Chaimae, S. & Habiba, C. (2023). SDN/NFV Security Challenges and Proposed Architecture, Proceedings of the 2023 7th IEEE Congress on Information Science and Technology (CiSt), Published by IEEE, pp. 145-149. ISBN 978-1-6654-6133-7, ISSN 2327-1884, Agadir - Essaouira, Morocco. DOI: 10.1109/CiSt56084.2023.10409955, in press.

[6] Tupakula, U.; Karmakar, K. K.; Varadharajan, V. & Collins, B. (2022). Implementation of Techniques for Enhancing Security of Southbound Infrastructure in SDN, Proceedings of the 2022 13th International Conference on Network of the Future (NoF), Published by IEEE, pp. 1-5., ISBN 978-1-6654-7254-8, ISSN 2833-0072, Ghent, Belgium. DOI: 10.1109/NoF55974.2022.9942644, in press.

[7] Ahmed, Z. & Bashir, B. S. A. (2022). Optimized and Secured Utilization of Infrastructure Resources using VMWare Stretched Cluster Multi-Site Solutions, Proceedings of the 2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS), Published by IEEE, pp. 1-4., ISBN 978-1-6654-6883-1, Kochi, India. DOI: 10.1109/IC3SIS54991.2022.9885490, in press.

[8] Redžepagić, J.; Dakić, V.; Stanešić, J. & Bašić, M. (2023). Analysis of Password Security Policies and Their Implications on Real-Life Security, Proceedings of the 34th DAAAM International Symposium, pp.0077-0081, ISBN 978-3-902734-41-9, ISSN 1726-9679, Vienna, Austria. DOI: 0.2507/34th.daaam.proceedings.011, in press.

[9] Dizdar, K.; Morić, Z., Dakić, V. & Bašić, M. (2023). Denial of Service Attacks Using the Example of Croatian Hosters, Proceedings of the 34th DAAAM International Symposium, pp.0088-0095, ISBN 978-3-902734-41-9, ISSN 1726-9679, Vienna, Austria. DOI: 10.2507/34th.daaam.proceedings.013, in press.