electronics

MDPI

Review

Security and Privacy Issues in Software-Defined Networking (SDN): A Systematic Literature Review

Muhammad Shoaib Farooq *, Shamyla Riaz and Atif Alvi

Department of Computer Science, University of Management and Technology, Lahore 54770, Pakistan; shamyla.riaz@umt.edu.pk (S.R.); atif.alvi@umt.edu.pk (A.A.)
* Correspondence: shoaib.farooq@umt.edu.pk

Abstract: Software-defined network (SDNs) have fundamentally changed network infrastructure by decoupling the data plane and the control plane. This architectural shift rejuvenates the network layer by granting the re-programmability and centralized management of networks which brings about exciting challenges. Although an SDN seems to be a secured network when compared to conventional networks, it is still vulnerable and faces rigorous deployment challenges. Moreover, the bifurcation of data and control planes also opens up new security problems. This systematic literature review (SLR) has formalized the problem by identifying the potential attack scenarios and highlighting the possible vulnerabilities. Eighty-six articles have been selected carefully to formulize the SLR. In this SLR, we have identified major security attacks on SDN planes, including the application plane, control plane, and data plane. Moreover, this research also identifies the approaches used by industry experts and researchers to develop security solutions for SDN planes. In this research, we have introduced an attack taxonomy and proposed a collaborative security model after comprehensively identifying security attacks on SDN planes. Lastly, research gaps, challenges, and future directions are discussed for the deployment of secure SDNs.

Keywords: SDN; software defined networking; application plane; control plane; data plane; SDN security

Citation: Farooq, M.S.; Riaz, S.; Alvi, A. Security and Privacy Issues in Software-Defined Networking (SDN): A Systematic Literature Review. *Electronics* **2023**, *12*, 3077. https://doi.org/10.3390/electronics12143077

Academic Editors: Xianfu Chen, Bo Zhang, Jie Li, Lei Shi and Yangjie Cao

Received: 23 May 2023
Revised: 10 July 2023
Accepted: 10 July 2023
Published: 14 July 2023

1. Introduction

The software-defined network (SDN) has become one of the top networking architectures for simplifying network management by enabling innovation in network communication. The fundamental characteristic of SDN architecture is to decouple the control plane from the data plane. A control plane is logically centralized to maintain the network state and gives instructions to the data plane [1]. In the data plane network, devices forward data packets by following the control instructions. However, this architectural transformation has received immense attention from the network industry and academic fields. Additionally, the advantages of SDNs have been proven in different scenarios, such as Google B4, NTT's edge gateways, and Microsoft's public cloud [2–4].

SDN also offers a standardized or consistent application programming interface (API) for adding up-to-date programmable features to the network to overcome flexibility and programmability issues in traditional networks. In addition, SDNs help network service providers to obtain a more flexible, manageable, and programmable network architecture [5]. These properties of SDNs help to empower the control plane and accomplish a global view of network topology to dynamically modify the functionalities of the network. Although SDNs manage networks in a more centralized way, it is now endorsed by both academic and industrial practitioners. This is because security has become a major concern at all levels, especially in newly designed network systems, such as cloud networks and peer-to-peer networks. Therefore, despite the number of advantages, SDNs have many inherent network security challenges, including scalability, reliability, controller placement, and latency [6]. Moreover, several security attacks were also investigated by other

researchers and were examined in other network systems [7–10]. On the downside, the increased potential of security attacks on SDN layers has become a prime concern. The increased potential of security threats, including the consistency of flow rules, controller vulnerability, legitimacy, malicious applications, and standardized and northbound and southbound communications, occurs due to a lack of best practices of SND functions, components, and the open programmability of networks. From the literature, it is clear that due to the multi-layered architecture of SDN, security threats to different layers are different.

SDN architecture is presented in Figure 1, with the most common and major security challenges on SDN layers. The application layer is also known as the management layer and is the topmost layer in the SDN architecture. All business and security applications that are designed by developers are executed on this layer. Applications controlled by this layer consist of firewall implementation, access control, load balancer, intrusion prevention system (IPS), intrusion detection system (IDS), and network virtualization.

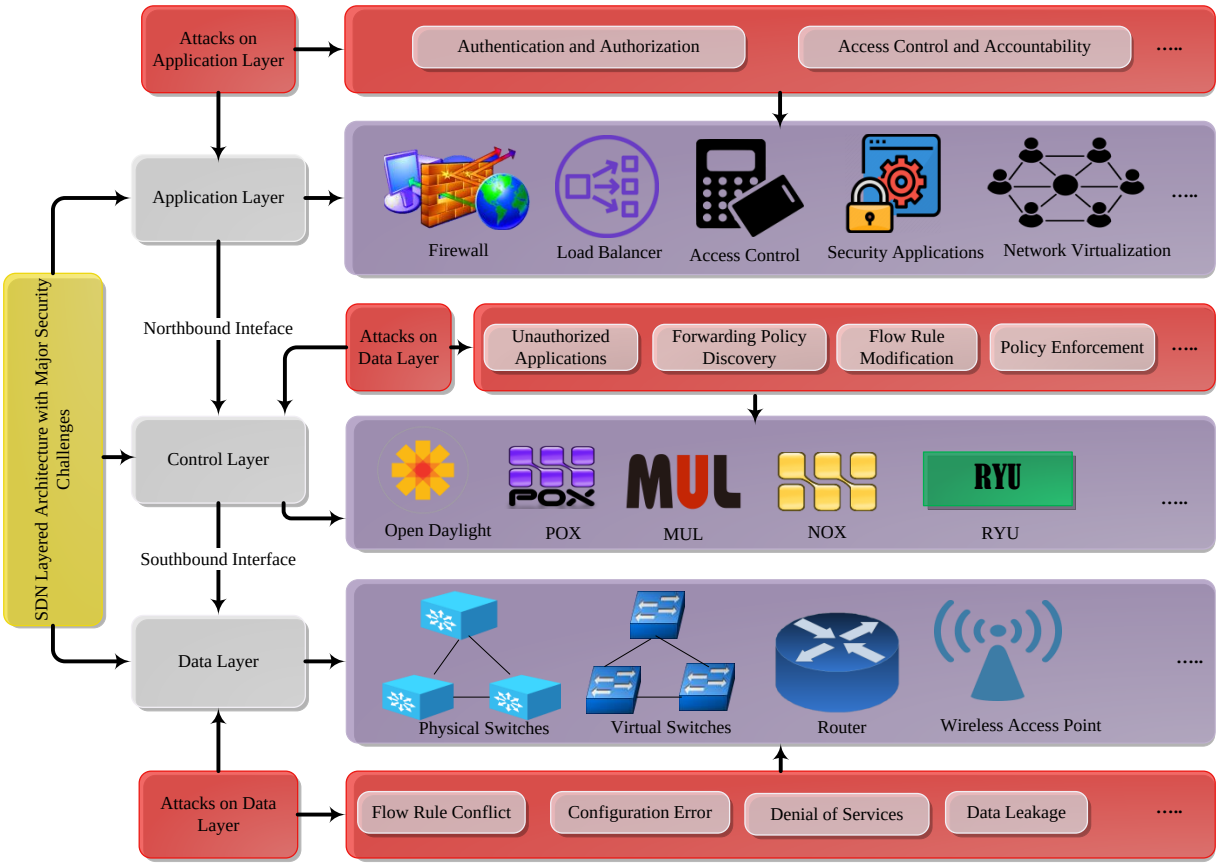

Figure 1. SDN Architecture with major security attacks.

The application layer communicates with the control layer by using northbound API [11]. Although applications deploy multiple network services in SDN, there are still some major security challenges in SDN due to the emerging abilities of hackers. The most common security threats at this layer are authentication, authorization, access control, and accountability. The control layer is the mediator between the application layer and the data layer, which consists of the SDN controller or network operating system (NOS). The overall responsibility of this layer is to manage the functionality of the entire network by taking decisions on packet forwarding and routing [12]. The control layer communicates with its lower layer (data layer) by using southbound API. The logically centralized controller consists of node flows like flood light, POX, NOX, MUL, Jaxon, etc.

This layer is only responsible for decision making due to its logically centralized controller; therefore, it is easily targeted in order to perform malicious tasks across the

entire network. Major security challenges on this layer are unauthorized applications, flow rule modification, policy enforcement, and forwarding policy discovery. The data layer is responsible for packet forwarding, according to the assigned policies of forwarding devices, such as physical switches and virtual switches.

To deploy packet forwarding policies in SDN, OpenFlow switches are used to support the flow tables and flow rules, because OpenFlow is the most widely used southbound interface in SDN scenarios. Flow tables are the main data structure in OpenFlow devices that contain a set of flow entries. Packet forwarding devices analyze the incoming packets through flow tables and take corrective measures regarding the received information. However, flow rules are used to control the behavior of packet forwarding, which is identified via matching packet fields as well as other features, such as packet counters. When the first packet arrives from the new host at the open flow switch, it is necessary for the controller in the flow table to install a new flow rule. However, due to the space constraint, every switch has a limitation on flow table entries, which generates new security issues. The most common security challenges that arise at this layer are DoS, configuration errors, flow rule conflicts, and data leakage. Over the last few years, many meetings and conferences have been held to discuss security issues as well as solutions. Apart from this, researchers and practitioners have also presented some security solutions related to SDNs through authentication mechanisms and policy conflict resolutions. However, there is a need for an further degree of attention in order to achieve SDN deployment in data centers or individual organizations.

The objective of this research is to present a comprehensive systematic literature review related to security and privacy issues on SDN planes. This SLR provides contributions from four perspectives:

- This review presents all major security attacks on SDN planes, including the application plane, control plane, and data plane.
- The SLR identifies the approaches used by researchers and industry experts to present security solutions for SDN attacks on planes.
- Moreover, SLR also presents a security model after identifying malicious attacks on the SDN application plane, control plane, and data plane.
- Additionally, we also present challenges and research gaps as well as suggest future research directions to produce a sustained solution for SDN security.

The organization of the SLR is as follows. Section 2 provides the related work on SDN security challenges. In Section 3, the research methodology is presented by defining the search string, research questions, and exclusion and inclusion criteria in order to collect relevant studies regarding security issues on SDN planes (applicant plane, control plane, data plane). Research results are presented in the classification table in Section 4 by categorizing the selected studies according to the defined search protocol. In addition, in this section, we also present the major security attacks, causes, and solutions on SDN planes. In Section 5, we proposed a security model by analyzing the security attacks and causes through our findings. In Section 6, research gaps and future research directions are presented. The conclusion of the paper is presented in Section 7.

2. Related Work

The decoupling of the data plane and control plane represents an excellent future for networks, but it has brought new security challenges into existence. For example, the communication channels can be targeted between isolated planes in order to impersonate one plane to attack the other. Moreover, the control plane is also more appealing to vulnerable attacks due to its visible nature, such as DoS and DDoS attacks. In addition, the SDN controller will also down the whole network if there is any security compromise. Security challenges in SDN are growing gradually with the deployment of its technologies in different areas. Therefore, it is necessary to highlight the security issues so that required security measures can be properly taken in order to obtain the full advantages of SDN. In this section, we have identified the security vulnerabilities at different SDN layers. To

describe the security issues in SDNs, Kreutz et al. [13] presented threat vectors, which demonstrated that there is no persuasive mechanism for building a trusted relationship between applications and the controller in SDNs. Thus, forged traffic flows can be injected into controllers and switches that can be triggered by a malicious user. In this way, an attacker will use network elements such as servers, switches, and computers to generate a DoS attack. Similarly, attacks on vulnerabilities in controllers and switches can wreak havoc on the network; therefore, malicious controllers compromise the whole network. The prevention of DDoS attacks has been a primary concern for researchers and network security administrators [14–17]. DDoS attacks are highly frequent; therefore, it is necessary to develop robust solutions that are effective in detecting and mitigating DDoS attacks [18,19]. From the literature, it can be seen that a DDoS security mechanism must be able to prevent attacks from within and outside the network [20].

Schehlmann et al. [21] presented an evaluation methodology for SDN security and a comparison with other conventional networks. SDN security measurement criteria consist of authenticity, availability, confidentiality, consistency, and integrity. Although authors argue that a number of attacks in SDN can also exist in conventional networks, they have not explored attacks on SDN layers and their impacts on the overall SDN architecture. Abdulkarem et al. [22] have identified the DDoS attack in the SDN data layer and proposed a solution for attack detection and mitigation by implementing python language. Moreover, the central controller is one of the most vulnerable components of SDN architecture due to weak authentication, information disclosure, and incomplete encryption. Hence, if the controller on the control plane is not properly secure, then the entire network will be badly affected [19]. Switches have the weakest performance in terms of hardware because an attacker attacks communication channels in order to destroy the link between switches and controllers. According to OpenFlow, standard switches will be changed into the standalone mode or fail-secure mode [23]. Further, the multi-controller implementation will divide the whole network into multiple networks, which leads to consistency and privacy issues [24]. However, the consistency and legitimacy of flow rules are the major security issues in the data layer. During the release process, malicious tampering or transmission delays cause inconsistency issues to take place between switches and controllers [25].

Furthermore, the northbound interface also faces a number of security challenges, the biggest security vulnerability at this interface is standardization [26–28]. There is no consistent provision regarding authentication and authorization methods due to diversity as well as regular updates in SDN applications. By exploiting the programmability and openness of the northbound interface, an attacker can launch the attack and access the important resources in the control to change or occupy the network status. In the past few years, studies have focused on passive differential power analysis (DPA) and active differential fault analysis (DFA) by measuring the consumption of power of one or more operations. However, some attacks are considered side-channel attacks in order to obtain patterns from extracted information [29].

Moreover, the southbound interface also suffers different security attacks due to the leakage of open flow protocols, because open flow uses TLS/SSL protocol for data encryption, which is not secure [30–35]. Additionally, the southbound interface also faces data leakage, controller spoofing, eavesdropping, and many other security attacks. Scott-Hayward et al. [36] presented a comprehensive survey on different security challenges in SDNs and identified the proposed solution as well as describing the holistic approaches that are necessary for SDN security. Ahmad et al. [37] identified the security threats in SDN at the application plane, data plane, and control plane along with security platforms that can secure each plane from different attacks. Table 1 shows already published SLRs, highlighting the contribution of our research. We have compared our defined research questions with already available solution where ✓symbol indicates the matched contribution of the study and × symbol shows the gaps of their study.

Table 1. Comparison of LSR with other studies.

Ref.:	Addressed Attacks and Solutions for Application Plane	Addressed Attacks and Solutions for Data Plane	Addressed Attacks and Solutions for Control Plane	Proposed Security Model against Each Attack	Discussed Challenges, Gaps, and Future Research Directions
[38]	×	✓	×	×	✓
[39]	✓	✓	✓	×	✓
[40]	×	×	×	×	✓
[41]	×	×	×	×	✓
[42]	✓	✓	✓	×	✓
[43]	✓	✓	✓	×	
[44]	×	×	×	×	✓
[45]	✓	✓	✓	×	✓
[46]	×	✓	×	×	✓

3. Research Method

The primary objective of a SLR is to present inclusive knowledge from the literature in the research field in an organized and holistic way. Apart from that, a SLR can also help to identify existing research gaps as well as the consequent recognition of avenues for research in the future. In this section, we define the method used to conduct this SLR. The method involved research questions (RQ), search string, data sources, eligibility criteria (inclusion and exclusion criteria), screening and selection process, and quality assessment (QA) criteria.

3.1. Research Questions

The objective of this SLR is to provide a comprehensive review of security and privacy issues in SDNs. Therefore, we designed seven research questions in the first phase of this SLR. We evaluated the security attacks/threats on SDN layers/planes and proposed solutions for those attacks. The RQs are given in Table 2 with their corresponding motivations.

Table 2. Research questions (RQs).

NO	Research Question	Main Motivation
RQ1	How has the frequency of research approaches been changed over time in the field of SDN security?	To identify the published studies in the SDN security area over time.
RQ2	What are the primary publication channels for identifying security attacks and their solutions on SDN planes?	To identify the primary sources published in SDN security and privacy issue-related studies.
RQ3	What research approaches have been used by researchers to identify the security and privacy issues on SDN planes?	To identify the proposed research approaches related to SDN security issues, security solutions, and attack causes.
RQ4	What are the major attacks targeting planes in SDNs addressed by the literature?	To address the security attacks and proposed solutions for SDN planes.
RQ5	What are the major types of attacks identified by researchers?	To identify the attacks/threats on SDN planes.
RQ6	What are the major causes of attacks addressed in the literature on SDN planes?	To identify the causes of attacks on SDN planes.
RQ7	What proposed solutions have been implemented to address security attacks on SDN planes?	To identify the proposed solutions for security attacks on SDN planes addressed in the literature.

3.2. Search String

The search was conducted at the start of December 2020 by applying the search string to different databases. We applied the search string via the Boolean operators "ANDs" and "ORs" as follows: ("Software Defined Networks" OR "SDN") AND ("SDN security" OR "SDN threats") OR ("SDN Security Solutions" OR "SDN Layers").

3.3. Data Sources

The next phase was to search for the source references. In this phase, multiple search terms were implemented, such as search keyword, search source, relevant papers selection, and filtering. The main research was carried out by looking at the keywords, paper titles, and abstracts for each paper or journal. There were four types of publication, namely journals, conferences, symposiums, and reports. The obtained results from each database are shown in Figure 2. The digital search process was carried out by implementing the search query in seven different databases. The chosen databases were:

- IEEE Xplore (ieeexplore.ieee.org) in 1 April 2023.
- Science Direct (sciencedirect.com) in 2 January 2023.
- Springer (springerlink.com) in 1 February 2023.
- Hindawi (hindawi.com) in 15 February 2023.
- MDPI (mdpi.com) in 25 February 2023.
- Plos (plos.org) in 15 April 2023.
- Wiley (onlinelibrary.wiley.com) in 21 April 2023.

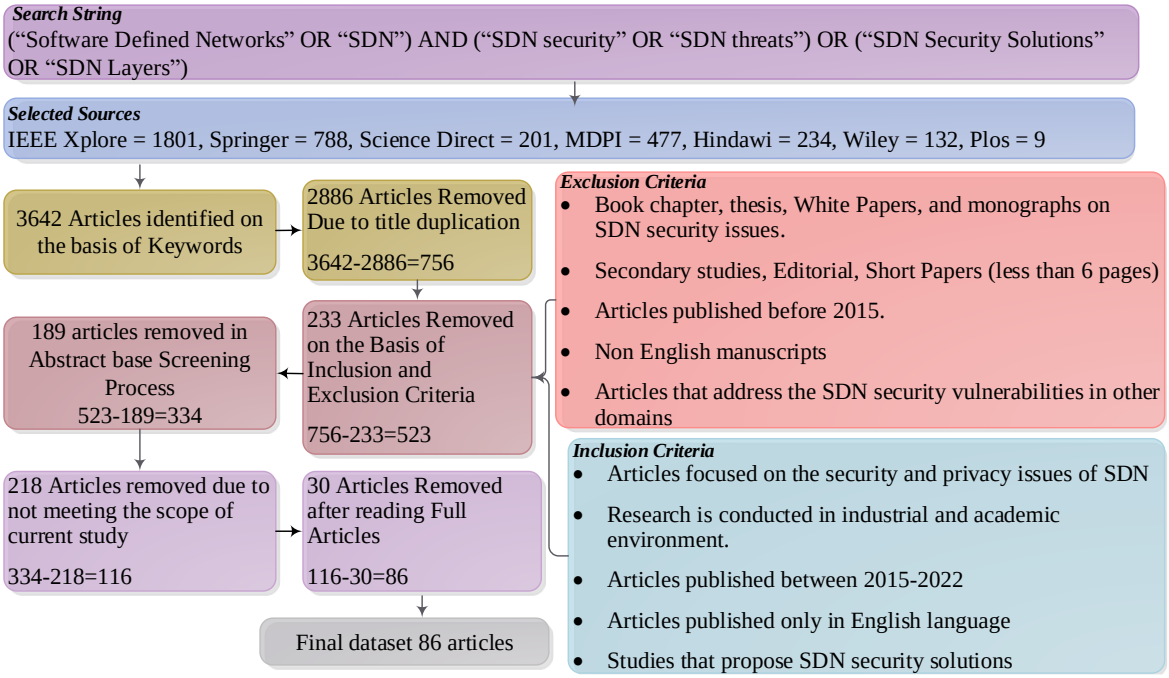

Figure 2. Selection and screening process.

3.4. Screening Process

After implementing the search string in different scientific databases, we used a search process defined by Dybå et al. [47] for screening relevant papers. In the first stage of screening, we selected the papers on the basis of their title and excluded papers that were not related to the research field. For example, the search query returned papers related to SDN security in different fields, such as the internet of things, cloud computing, blockchain, etc. Moreover, some phases make it difficult to measure the actual relevancy of screened papers on the basis of their title. To resolve this issue, we passed the papers to another screening phase. In the second phase of screening, we read the abstract of each paper selected in the first phase.

3.5. Inclusion and Exclusion Criteria

The selection of the published peer-reviewed articles is governed by defining the exclusion and inclusion criteria. Because all studies in searchers were not exactly related to the defined RQs, the selection of the papers was needed to identify their actual relevance.

The exclusion and inclusion criteria for this SLR is shown in Figure 2. After applying the defined exclusion and inclusion criteria to the papers, we included the selected studies in the next screening and selection phase.

3.6. Study Selection Process

When the defined search query was applied in selected databases, 3642 papers were initially retrieved. The first screening phase was applied on the basis of the retrieved paper's title. The screening process based on the title was carried out by two authors, resulting in 756 papers. After the selection of 756 papers, we removed the duplicate papers by applying an abstract-based screening process and a total of 523 papers were selected. A high number of papers were excluded because they were not relevant to our research topic. For example, a number of papers were related to SDN security issues in the cloud, IoT, and blockchain. However, the focus of our research is to review the security issues and challenges on SDN planes (application plane, control plane, and data plane). In the last selection procedure, two authors read all selected papers and excluded 30 papers due to their focus on specific SDN security issue domains. Two authors read the final selected papers, which did not lead to any exclusions. The selection process results are presented in Figure 2.

3.7. Quality Assessment Criteria

QA is considered one of the most critical phases to evaluate the quality of selected studies. QA questions aim to assess the internal and external validity of reviewed articles and measure the scope that these articles address. We define the four QA questions as follows:

QA1: Explicit discussion about data analysis and the possible answer will be: "Quantitative = +1" "Qualitative = +0.5" or "No analysis performed = +0".
QA2: Discussed research concerns are valid according to the defined methodology and the topic of interest: "Yes = +1", "Partial = +0.5" and "No = +0".
QA3: Discussion about challenges and advantages of selected topic: "Yes = +1", "Partial = +0.5" and "No = +0".
QA4: Number of citations and source reliability:

(+2) if sum of the citations $\geq$50;
(+1.5) if sum of the citations $\geq$10 and $\leq$49;
(+1) if sum of the citations $\geq$1 and $\leq$9;
(+0) if the sum of the citations =0.

4. Analysis

Regarding the objective of this research, the findings of SLR are explained in this section. After the screening and selection processes, selected studies were used to identify the answers of each research question. This analysis formulates a fundamental contribution to identifying the security challenges in SDN along with their proposed solutions to make the network system more secure in the future.

4.1. Results Selection

The state-of-the-art analysis of multiple security threats on SDN planes is a key challenge due to their integration into multiple technologies, including cloud computing, IoT, big data, machine learning, blockchain, etc. However, according to the defined RQs, we focused only on three SDN planes (application plane, control plane, and data plane) and gathered 86 studies. After carrying out a deep analysis of the selected articles, we addressed all RQs according to the extracted information. The classification results of the selected studies and quality assessment scores are shown in Table 3.

Table 3. Classification and quality assessment score of security challenges on SDN planes.

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[48]	2017	Proposed system	Control plane	Distributed Denial of Service (DDoS) Attacks	The attack occurs due to weak authentication and malicious flow rule.	Proposed SDN-Guard system to detect and mitigate the attacks against SDN rootkits.	1	1	0.5	1	3.5
[49]	2016	Proposed system	Control plane		An attack happens due to malicious traffic, flow rules, and flow timeouts.	SDN-Guard is a novel scheme developed to protect the network from DDoS attacks.	1	1	0	2	4
[50]	2016	Proposed system	Data plane		Unsteadiness between security features and network performance generates DDoS attacks.	A security design approach has been proposed to prevent the controller and network from attacks	0.5	1	0.5	1.5	3.5
[14]	2018	Model	Control plane		DDoS attacks make the resources available to an unauthorized user.	A hybrid machine learning model has been proposed to secure the SDN controller from attacks.	1	1	0.5	1	3.5
[21]	2020	Review	Control plane		Attackers scan the network for vulnerable hosts and exploit them with malicious programs to generate DDoS attacks.	Presented state-of-the-art review to identify the techniques and methods that have been proposed to detect DDoS attacks.	0.5	1	1	1	3.5
[16]	2019	Proposed solution	Control plane, data plane		Attackers use multiple potential vulnerabilities due to the centralized nature of SDN.	The proposed solution protects both the data plane and control plane from malicious attacks.	1	1	0.5	1.5	4
[17]	2020	Framework	Application plane		Attack detection in the application plane is difficult due to their stealthy nature and a potential increase in DDoS attacks	A robust self-protection framework has been proposed which mitigates DDoS attacks on the application layer.	1	1	0.5	0	2.5
[51]	2016	Proposed solution	Control plane		Attacks are generated because of bottlenecks and hindrances in the on-demand network's capability in the control plane.	To accommodate the workload surges on the control plane, an SDN shield solution has been proposed.	1	1	1	1.5	4.5
[52]	2019	Proposed solution	Control plane		The centralized nature of the controller makes it more vulnerable to launching DDoS attacks.	Safe-guarding schemes decrease DDoS attacks by using an anomaly traffic detection module and controller dynamicdefense.	1	1	1	1.5	4.5
[53]	2020	Proposed solution	Application plane		Slow TCAM exhaustion attacks and saturation attacks originate the DDoS attacks.	A solution has been designed to slow down DDoS attacks on the SDN application plane.	1	1	1	1	4
[54]	2019	Proposed method	Data plane		Malicious packets overload the secure channel, software control agent, and controller to inject DDoS.	Investigated DDoS attack methods and developed a DosDefender to filter malicious packets.	1	1	1	1.5	4.5
[55]	2016	Model	All		Centralized controller configuration originates security vulnerabilities.	Proposed a STRIDE threat model to analyze the security model.	1	0.5	0.5	1.5	3.5
[56]	2016	Proposed method	All		DDoS attacks are hard to trace and mitigate once they have started.	Solutions provided by this research include attack detection, trigger, traceability, and mitigation.	1	1	1	2	5

Table 3. *Cont.*

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[57]	2017	Framework	All		Heavy reliance on other software modules instigates DDoS attacks.	A DDoS attack mitigation framework called ArOMA is proposed.	1	1	1	1.5	4.5
[58]	2016	Proposed solution	Control plane		As the controller is considered the brain of the computer that manages the whole network, it is attractive for attackers.	To protect the central controller from DoS attacks, adaptive suspicious prevention solutions have been proposed.	1	1	1	1.5	4.5
[18]	2019	Proposed system	Control plane		The attack is generated by a flooding request messages to the victim.	The proposed system uses an unsupervised deep belief network algorithm.	1	1	1	0	3
[59]	2015	Proposed system	Data plane, control plane		Vulnerabilities of the controller are higher because it is a single point of failure.	In order to defend the blind DDoS attacks, a system called “moving target defense” is proposed.	1	1	0.5	1.5	4
[60]	2019	Proposed method	All		Identified techniques used to mitigate DDoS attacks are expensive and security threats are difficult to detect.	By using the lion optimization algorithm DDoS attacks can be detected.	1	1	1	1.5	4.5
[61]	2019	Proposed method	All		A number of DDoS attacks, such as UDP flooding and SYN flooding, are due to the legitimate activities of a user.	An advanced support vector machine method is proposed to detect the two types of DDoS attacks.	1	1	1	1	4
[62]	2018	Framework	Control plane		Due to the easy exploitation of centralized controllers, it has become the central point of vulnerable attacks.	Proposed a defense framework called OverWatch for DDoS attack detection.	1	1	1	1.5	4.5
[19]	2018	Proposed solution	Control plane		The separation of the control plane and data plane is a major reason behind DDoS attacks.	Controller-to-controller protocol was proposed for the mitigation of DDoS attacks.	1	1	1	1.5	4.5
[63]	2019	Proposed system	All		In SDN architecture, network devices create multiple vulnerabilities to generate DDoS attacks.	Developed an intrusion detection system to detect multiple types of DDoS attacks.	1	1	1	1.5	4.5
[64]	2020	Framework	Control layer		The DDoS attack causes the controller services to be unavailable for the authenticated user.	In order to detect the DDoS attack against SDN, a framework has been proposed.	1	1	1	1	4
[65]	2016	Proposed solution	Data plane		Vulnerabilities exposed by the data plane generate DDoS flooding attacks.	Proposed a solution to mitigate the DDoS attacks on the data plane by maximizing performance degradation.	1	1	1	1	4
[66]	2021	Proposed Method	All planes		The existing limitations in the current DDoS detection method depend on the network topology, which it is hard to detect all DDoS attacks.	A detection method has been proposed that consists of classification and entropy-based methods and uses three collectors to detect the DDoS attack accurately.	0.5	1	0.5	1.5	3.5

Table 3. *Cont.*

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[20]	2021	Proposed System	Control and data plane		DDoS attacks occur due to the separation of the control and data plane.	A detection and defense system has been proposed that utilizes a generative adversary network for the detection of DDoS attack	1	1	1	1.5	4.5
[67]	2021	Proposed frame-work	Data plane		Due to heavy network traffic, severe challenges related to network activities happen which causes DDoS attacks	Designed a novel framework to detect DDoS attacks and intrusion detection systems on data layers.	1	1	0.5	1	3.5
[68]	2022	Proposed frame-work	All planes		Due to the expensive and unscaleable solutions of existing enterprises, smart networks are facing immense attacks.	DDoS attack detection and mitigation framework have been proposed based on machine learning techniques.	1	1	0.5	1	3.5
[69]	2021	Proposed solution	Data plane, control plane		Rapidly increasing network devices challenge the access layers functionalities and generate new security concerns.	Designed a secure control and data plane algorithm that can resist DDoS attacks through real-time monitoring.	0.5	1	0.5	1	3
[70]	2021	Proposed frame-work	All		Due to the low transportation rate and flash crowd nature, SDN planes are targeted by LDoS attacks.	Proposed a framework based on histogram-based gradient boosting and finding peaks (HGB-FP) algorithm.	1	1	0.5	1	3.5
[71]	2022	Proposed solution	Control plane		Due to unknown traffic analysis in machine learning, DoS saturation attacks occur.	The research presents an extension for the detection of DDoS saturation attacks.	1	1	1	0	4
[72]	2021	Proposed solution	All planes		DoS or DDoS attacks occur due to the linking of control and data plane.	A novel technique is proposed by using information theory metrics to detect DDoS attacks	1	0.5	0.5	1	3
[73]	2022	Proposed frame-work	Control and data plane		Separation of control and data planes enhances the flexibility and programmability of networks which leads to severe threats.	A blockchain-based DDoS defense framework to overcome these growing attacks on SDN planes.	1	0.5	1	0	2.5
[74]	2021	Proposed method	Control plane		In a DDoS attack, the attacker manipulates an entire network by sending a huge amount of malicious traffic.	Designed a network gate shield to detect DDoS attacks.	1	0.5	0.5	1	3
[75]	2022	Proposed frame-work	All planes		Low-rate DoS attacks destroy an entire network's security and generate huge losses.	To mitigate the low rate of DoS attacks an online attack mitigation system is proposed.	1	1	1	0	3
[76]	2022	Proposed method	Control plane		DDoS attacks bring down specific parts of a network in a very short span.	Two techniques PortMergIP and port mapping are proposed to mitigate the attack.	1	1	1	0	3

Table 3. *Cont.*

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[77]	2018	Proposed system	Data plane	Address Resolution Protocol (ARP) Spoofing Attacks	ARP spoofing and configuration-based attacks occur due to scalability challenges and threat vectors.	Proposed OFMTL-SEC, a novel security system that provides protection against attacks on the data plane.	1	0.5	0.5	1	3
[78]	2017	Architecture	Application plane		Attackers launch spoofing attacks due to the negligence of the source IP address of the packet.	Proposed IP source address validation architecture to cover the intra- and inter-domain spoofing attacks.	1	1	1	1.5	4.5
[79]	2015	Architecture	Application plane		CDN networks have created new security challenges by optimizing network topology.	In order to detect spoofed IP attacks, a defense architecture has been proposed.	0.5	0.5	0.5	1.5	3
[80]	2017	Proposed method	Data plane		A lack of flow table rules and switch controllers enhances the spoofing attacks.	Proposed multiple solutions to mitigate the spoofing attacks.	1	1	1	1.5	4.5
[81]	2020	Proposed solution	Control plane		ARP spoofing attacks come from the vulnerabilities of address resolution protocol (ARP)	Proposed a Bayes-based protocol to detect the attackers.	0.5	1	0.5	0	2
[82]	2017	Proposed solution	All		Neighbor discovery protocol messages are easily spoofed due to malicious nodes.	Proposed an authentication mechanism to secure the NDP.	1	1	1	1.5	4.5
[83]	2019	Survey	All		ARP poisoning attacks are further used by attackers for other malicious attacks such as DDoS.	Presented a comprehensive survey on multiple security attacks along with their proposed solutions to solve these threats.	0.5	1	1	1	3.5
[83]	2021	Proposed architec-ture	Control plane, data plane		Due to the vertical integration of the control plane and data plane, ARP spoofing attacks are most common on networks.	Designed and developed an architecture to handle the spoofing attacks on SDN planes.	1	1	1	1	4
[84]	2019	Architecture	Data plane	Flow Rule Conflicts	Malicious end hosts forward flow requests to the SDN controller which generates severe attacks.	Security architecture is proposed to manage the applications running in the SDN controller.	1	1	1	1	4
[85]	2015	Proposed solution	Application plane		Isolation of classifiers decreases the classification speed as well as delivering less accurate results.	Application plane classification is shown by applying machine learning and deep packet inspection approaches.	1	1	1	1.5	4.5
[86]	2015	Proposed system	Application plane		De facto implementation limits the use of control intelligence at the application plane.	In order to identify the attacks on application layer, the content parser system called COPY is proposed.	1	1	1	1	4
[87]	2018	Proposed method	All		A comprehensive and fine-grained rule collision detection technique is needed.	Proposed a deep detection method to detect the collision of flow rules.	1	0.5	0.5	0	2
[88]	2020	Survey	All		The interconnectivity of a large number of devices may result in flow rule conflicts.	Presented a comprehensive survey of existing SDN security issues in order to make sure the security standard.	0.5	1	1	1	3.5

Table 3. *Cont.*

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[89]	2018	Framework	Control plane	Weak Authentication and Communication Attacks	The control plane is targeted by attacks due to less support for fault tolerance and consistency checks.	Proposed a distributed SDN architecture for SDN control layer security.	1	1	0.5	1	3.5
[90]	2019	Proposed solution	Data plane		The addition of multiple wireless technologies in the data plane creates communication errors.	A secure distributed communication solution is proposed for the data plane and southbound API.	1	1	0.5	1	3.5
[91]	2018	Proposed solution	All		Adversaries monitor all network traffic to exploit whole information.	U-TRI technique has been proposed to detect the information leakage identifier.	1	1	1	1	4
[92]	2016	Architecture	Control plane, data plane		The weak authentication of cryptography generates illegal requests from malicious attackers.	Proposed a robust architecture to protect the network from malicious attacks.	1	1	0.5	1.5	4
[91]	2018	Architecture	Data plane		In terms of security and scalability, SDN architecture has brought new challenges due to the separation of the control plane and data plane.	Carrier-grade network requirements may be obtained by the proposed IEEE 802.1X port-based authentication architecture.	1	1	1	1	4
[93]	2019	Proposed solution	Data plane, control plane	Flooding Attacks	The flooding of TCP SYN packets from data to control plane launch saturation and buffer overflow attacks.	A novel solution called SAFETY has been proposed for the mitigation and detection of TCP flooding.	1	1	1	1.5	4.5
[94]	2016	Proposed solution	Control plane		Increases the response time overhead and degrades the controller performance.	SLICOTS solution has been proposed to mitigate the SYN flooding attack.	1	1	1	2	5
[95]	2018	Proposed method	Control plane		Centralized controller incurs new security vulnerabilities, such as flooding attacks.	To mitigate the UDP flooding, a lightweight countermeasure is proposed.	1	0.5	0.5	1.5	3.5
[96]	2017	Survey	All		Link-flooding attack congests critical network links and isolates the victim networks.	Presented a comprehensive survey on link flooding attacks on all layers/planes of SDN.	0.5	1	1	0	2.5
[97]	2016	Proposed solution	All		The centralized controller in SDN architecture makes it a more vulnerable attack target.	To overcome the flooding attacks, a self-organizing map application is developed.	1	1	1	1.5	4.5
[98]	2016	Proposed method	Data plane, control plane		When the controller is disabled, flooding attacks overload the controller.	In order to protect the controller from flooding attacks, a security-enhanced open vSwitch is proposed.	1	1	0.5	1	3.5
[99]	2018	Proposed method	Control layer		Breakdown of the controller may disrupt the whole network, which creates a packet-in messages flooding attack.	An effective detection method has been proposed to detect packet-in messages flooding attacks.	1	1	1	1	4

Table 3. *Cont.*

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[100]	2015	Framework	Control plane, data plane	Saturation Attacks	Due to the overloads in the infrastructure of SDN networks, saturation attacks occur.	A lightweight and independent framework called FloodGuard is proposed to mitigate security threats.	1	1	1	2	5
[101]	2015	Proposed solution	Control plane		Saturation attacks happen due to the bottleneck during the communication between the control and data plane.	LineSwitch solution is proposed to tackle the saturation attack on the control plane.	1	1	1	2	5
[84]	2019	Architecture	Control plane	Information Disclosure Attacks	Counteracting attacks take place via security architecture compromise over security management applications.	Designed architecture detects flow requests from malicious end hosts.	1	1	1	1	4
[102]	2018	Proposed method	Data plane, control plane		Hijacking SDN controllers and switches causes new security challenges.	A real-time method is proposed to detect compromised devices in SDN architecture.	1	1	1	1.5	4.5
[103]	2020	Proposed method	Application plane		Lack of availability of datasets for anomaly detection in SDN networks.	Proposed a comprehensive dataset to authenticate the performance of intrusion detection in SDN.	1	1	1	1	4
[104]	2017	Survey	All		Lack of programmability and centralized management causes fault management challenges.	A comprehensive survey on fault management issues on SDN layers/planes.	0.5	1	1	1.5	4
[105]	2016	Survey	All		Topology discovery is challenging due to the lack of authentication, integration, scarcity of SDN standards, and frequent migration of virtual machines.	A comprehensive survey on network topology discovery has been presented along with its security implications on SDN planes.	0.5	1	1	2	4.5
[106]	2018	Proposed method	Control plane	Tempering Attacks	Switch migration problems increase resource utilization and scalability issues on the control plane.	The switch migration method protects the controller in the network.	0.5	1	1	1	3.5
[107]	2019	Proposed solution	Control plane		Due to passive defense defects, the control plane suffers a tampering attack.	Proposed a security scheduling mechanism to provide diversity for improved security.	0.5	0.5	0.5	0	1.5
[108]	2017	Proposed system	Control plane, data plane		Attackers affect the network services and applications by injecting manipulated packets	A lightweight extension module called PackedChecker is presented to defend against this attack	1	1	0.5	1.5	4
[109]	2017	Proposed solution	Control plane, data plane		Due to the implementation of new devices on the control layer and data layer, it has become a greater attack targeted.	Authentication on the basis of a fingerprint resolves cryptographic security issues.	0.5	1	0.5	1.5	3
[110]	2019	Framework	Data plane	Scanning Attacks	Higher processing power costs and memory requirements weak the security in SDN-based data enters.	A collaborative security framework is proposed to address the security problems in SDN-enabled data enters.	0.5	1	1	1	3.5
[111]	2019	Survey	All		Multiple security causes have been identified which create malicious attacks on all planes of SDN.	Presented a comprehensive survey on the security perspective of the application of SDNs.	0.5	1	1	1.5	4

Table 3. *Cont.*

Reference	Classification						Quality Assessment				
	P. Year	Research Approach	Target Layers	Attack Type	Attack Causes	Attack Solution	QA1	QA2	QA3	QA4	Score
[112]	2015	Survey	All		Integration of new network applications in SDN architecture creates more security challenges.	Presented a comprehensive survey on different SDN attacks.	0.5	1	1	2	4.5
[113]	2016	Proposed method	All	Man-in-the -middle (MIM) attack	Powerful adversaries poison the topology information and security protocols.	Proposed a method to detect fake links and the existence of adversaries.	1	0.5	0.5	1	3
[114]	2016	Proposed solution	All	Cache Poisoning Attacks	Sniffer attacks capture and analyze the data to exploit the whole attack network.	SDN-based double-hoping communication method has been proposed to resolve the sniffer attack.	1	1	1	1.5	4.5
[115]	2017	Proposed method	All		Fingerprinting attacks occur on the operating system to obtain the system information for future attacks.	The fingerprinting method is proposed to secure SDN planes from fingerprinting attacks.	1	1	1	1.5	4.5
[116]	2020	Proposed solution	Control plane		Attackers use the open flow discovery protocol to inject malicious attacks including MIM, DDoS, etc.	Correlation-based anomaly detection techniques mitigate topology discovery attacks.	1	1	0.5	1	3.5
[117]	2019	Architecture	Control plane		Forwarding devices in the controller overload some controllers whereas some remains are unutilized.	ASLB architecture provides the solution for packet-in processing latency.	1	1	1	0	3
[118]	2016	Proposed solution	Control plane, data plane	Control Channel Hijacking Attacks	Threats in this scenario have a high impact on drivers' behavior as well as their quality of life.	Presented a comprehensive discussion about vehicular ad hoc network security threats.	1	1	1	1.5	4.5
[119]	2016	Architecture	All		Poor integration of SDN with servers and networks generates multiple security threats.	FS-OpenSecurity architecture is beneficial for administrators who want to handle security threats individually.	1	1	1	1.5	4.5
[120]	2020	Proposed scheme	All	Cyber Attacks	In a network-harvesting attack, an attacker steals the network credentials of any user.	The detection and defense scheme was designed by the author to detect and mitigate the network harvesting attack.	1	1	1	1	4
[121]	2022	Proposed frame-work	Control layer		Reliability of the control layer diverts all network traffic control due to which security attacks occur.	Proposed a novel technique called the adversarial path to identify the attack-targeted paths.	1	1	1	1	4
[122]	2021	Proposed solution	Control plane		Malicious cyber attacks occur due to weak network security, which affects their connectivity and continuity.	Proposed an optimized approach for SDN network operators to control critical attacks.	1	1	0.5	1	3.5

4.1.1. Assessment of RQ1: How Has the Frequency of Research Approaches Changed with the Passage of Time in the Field of SDN Security?

The yearly distribution of selected primary studies is shown in Figure 3. We selected the articles published between 2015 and 2020, according to their publication channels.

Interestingly, all selected studies related to SDN security issues were published after the year 2014. This indicates that security issues in SDN as a research field are a recent and challenging area. A closer overview of the yearly distribution of selected studies shows that 6 papers (8%) were published in 2015, 16 papers (23%) in 2016, 11 papers (16%) in 2017, 11 papers (16%) in 2018, 17 papers in 2019 (24%), and 6 papers (13%) were published in 2020. This shows the growing demand for SDN each year and an increasing number of security challenges on SDN planes. It can be seen that few papers were published in 2020 because the initial search process was performed in November 2020.

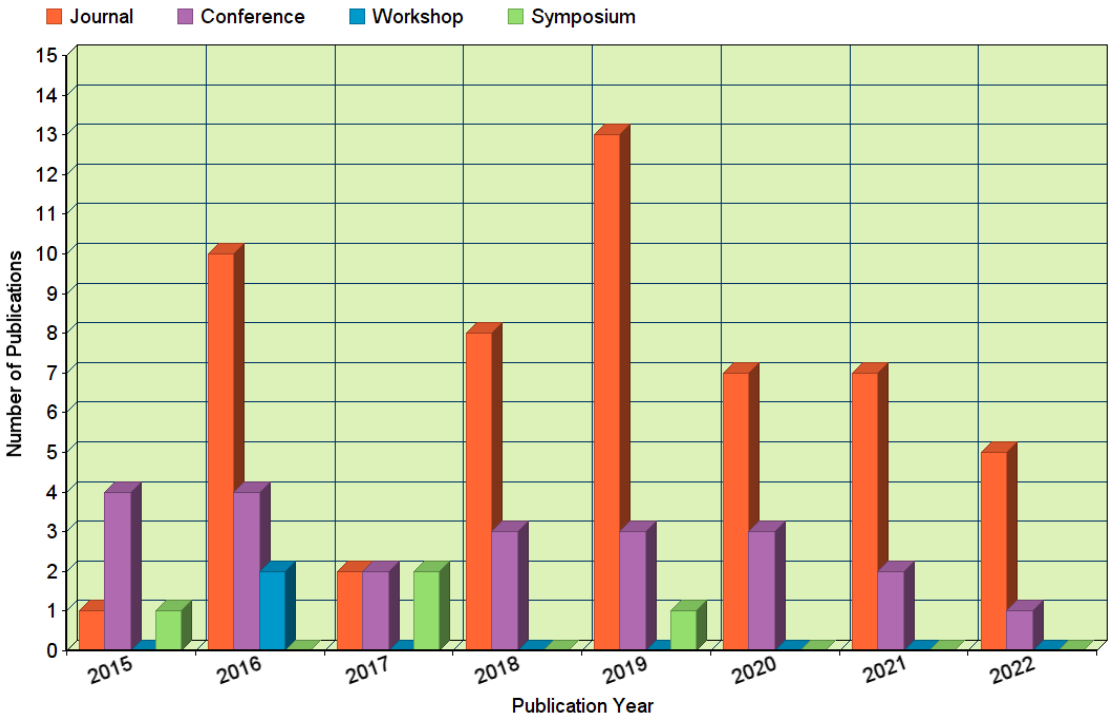

Figure 3. Yearly distribution of selected studies.

4.1.2. Assessment of RQ2: What Are the Primary Publication Channels for Identified Security Attacks and Solutions on SDN Planes?

Figure 4 shows the publication sources of selected studies, where we have identified seven different databases. The possible sources for selected studies are IEEE, MDPI, Springer, Science Direct, Wiley, Plos, and Hindawi. Furthermore, the selected channels for this SLR include journals, conferences, workshops, and symposiums. Our results showed that 66% of papers were published in journals, 27% in conferences, 3% in workshops, and 4% in symposia. Note that 44% of articles were published in IEEE journals, conferences, symposia, and workshops; however, 56% of the selected studies were published through other channels.

4.1.3. Assessment of RQ3: What Research Approaches Have Been Used by Researchers to Identify the Security and Privacy Issues in SDN Planes?

In this systematic literature review, we identified nine types of research approaches, including proposed solutions (26), proposed methods (17), proposed systems (10), architectures (9), frameworks (13), surveys (7), models (2), reviews (1), and schemes (1), as shown in Figure 5. The categorization of these approaches is shown in Table 3, according to their planes/layers. Moreover, identified approaches are discussed in this section:

Proposed Solution

Researchers have proposed a number of security solutions for SDN planes to identify, investigate, and mitigate the security threats in their actual contexts [85,86,90,95,96,101].

OF the multiple security threats in SDN, there is a considerable growth in DDoS attacks due to the isolation of the control plane and data plane [16,19,51–53,58,65]. Celesova et al. [16] proposed a specter solution to mitigate DoS/DDoS attacks by using machine learning approaches and open flow possibilities, which cover a wide area of security on the control plane and data plane.

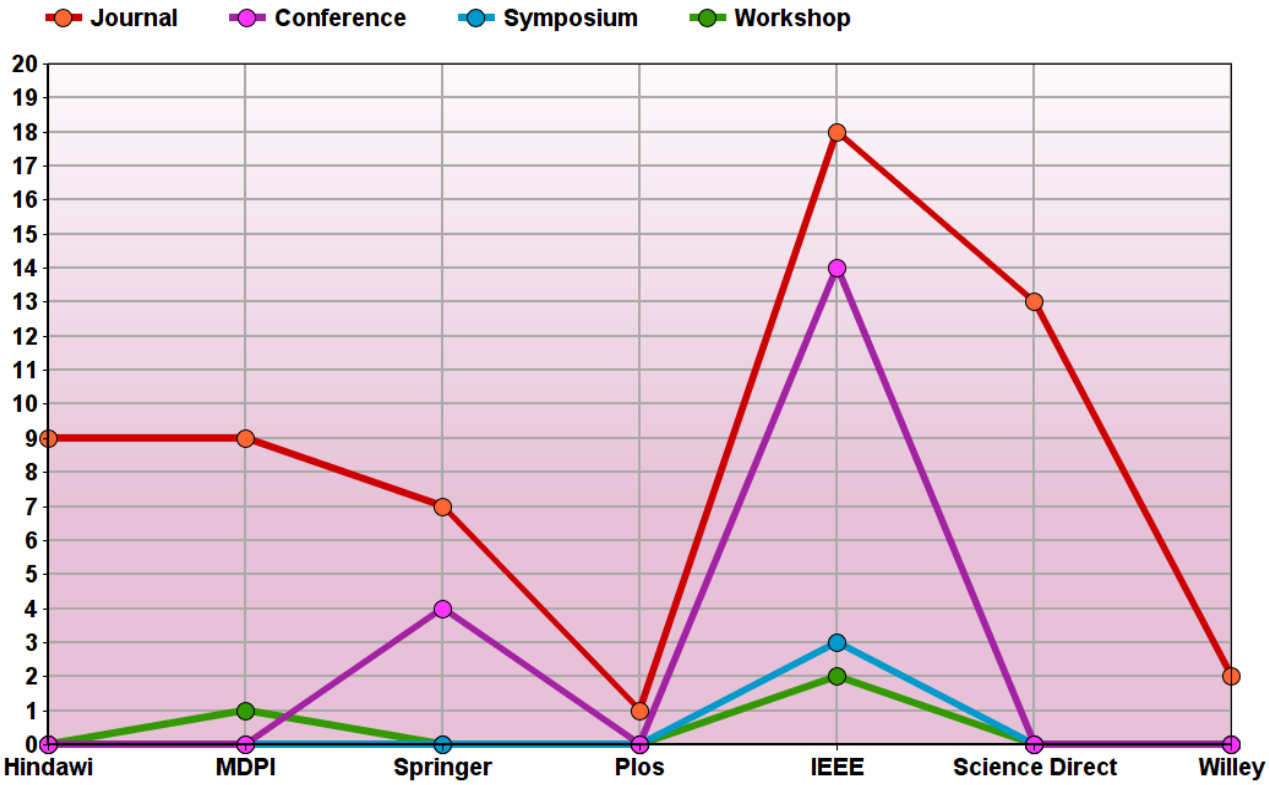

Figure 4. Frequency of publications from each source.

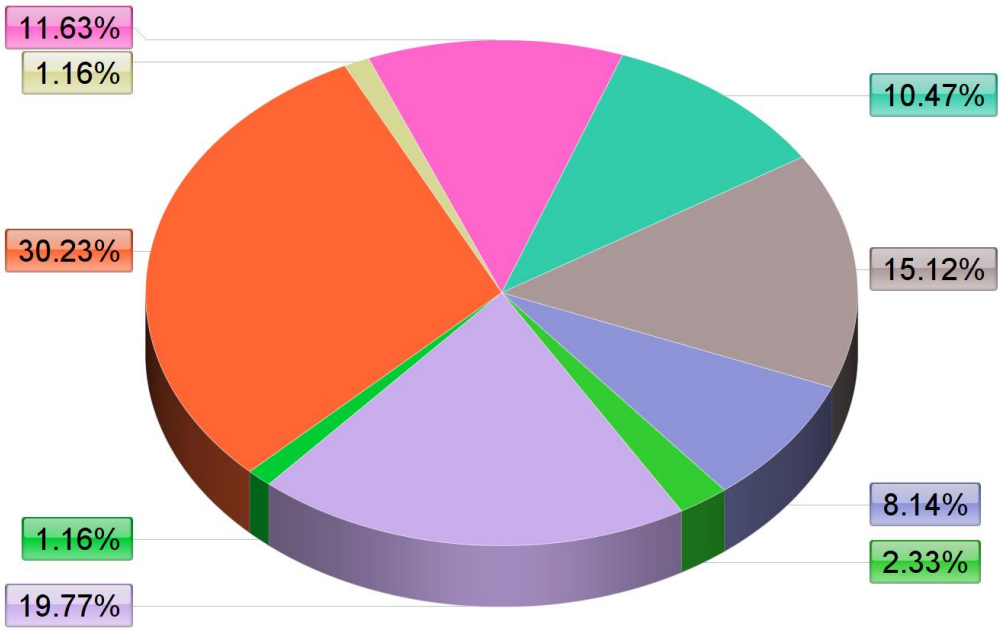

Figure 5. Used approaches in selected studies.

Proposed Method and System

Different methods and systems have been investigated to detect and mitigate security attacks on SDN planes. As the decoupling of control and data plane provides centralized control, programmability, and flexibility, on the other hand, there are also many vulnerabilities due to communication conflicts between these planes. These vulnerabilities are leveraged due to the saturation of the control plane and buffer overflow attacks, which are introduced via TCP SYN flooding. To detect and mitigate the TCP SYN flooding attack, the SAFETY solution is presented, which determines the flow data randomness [95]. Additionally, another countermeasure, SLICOTS, also mitigates the TCP SYN flooding attack at the control plane by taking the advantage of the SDN's dynamic programmability nature [14].

Architecture and Framework

Researchers proposed a number of security architectures and frameworks to make the SDN architecture secure [57,119]. In this research, we identified the proposed architectures [78,84,93,93,94,94,100,117] and frameworks [62,64,89] in regard to each SDN plane (application plane, control plane, data plane). Mowla et al. [79] proposed an SDN-based CDNI network architecture to detect spoofed IPs and developed a defense to mitigate spoofing attacks. It also optimizes network services by evaluating packet handling, topology, and traffic paths to overcome IP spoofing attacks. Further, the application layer is protected from DDoS attacks by leveraging SDN and deep learning enablers. The proposed framework empowers fully autonomous attack detection and mitigation on the application layer.

Survey and Review

By using qualitative methods, researchers have presented comprehensive surveys on multiple security issues in SDN, such as DDoS attacks, link flooding attacks, and ARP poisoning attacks, along with their proposed solutions [88,104,105,111,112]. Shah et al. [83] presented a comprehensive survey on the solutions to ARP cache poisoning attacks. The identified solutions are divided into three categories, i.e., solutions on the basis of traffic patterns, IP-MAC address bindings, and flow graphs. Moreover, these solutions were also analyzed with respect to ARP response time, attack detection time, and delay calculation at the controller. Further, an extensive review of DDoS attack detection techniques is presented, which categorizes these detection techniques at a high level according to the methods used [21].

Model and Scheme

In the model research approach, researchers accomplished a provisional understanding of SDN security challenges by using controlled testing techniques to investigate the major causes and threats on SDN planes. Deepa et al. [14] proposed a hybrid machine learning model to detect and mitigate DDoS attacks on the controller. Further, to overcome the isolation attacks in SDN, two defense schemes were proposed, i.e., SpoofDefender and RSDetector [120]. The RSDetector scheme detects the rouge switches in the network and the SpoofDefender scheme protects the SDN planes from spoofing attacks.

4.1.4. Assessment of RQ4: What Are the Major Attacks Targeting Planes in SDNs Addressed in the Literature?

SDNs have become some of the most promising and hottest technologies in terms of flexibility, scalability, and effectiveness due to their wide implementation in different areas, such as data centers, smart homes, smart grids, wireless LAN, etc. However, the architecture is not as mature as required and its centralized control nature has caused a wide variety of new security challenges. Table 3 shows that attack on different planes are different due to the multi-layer architecture of SDNs. Therefore, we classified the security issues on each plane along with their proposed solutions. Attack frequency on the application plane is 21.48%; on the control plane, it is 46.31%; and 32.21% on the data plane,

as shown in Figure 6. Among the security threats on SDN planes, we noticed that there has been a significant rise in DDoS attacks. The frequency of DDoS attacks is higher in the control plane when compared to other planes because the controller is only responsible for packet-forwarding decision making in SDN architecture [14,16,21,48,49,51,52].

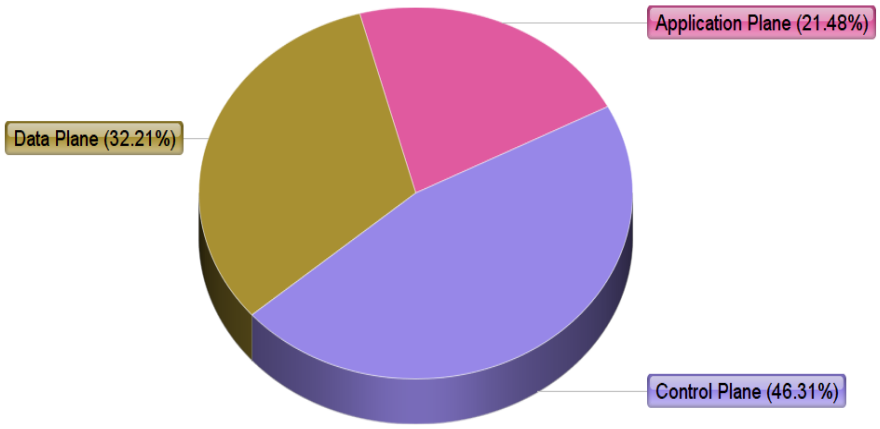

Figure 6. Number of selected studies for each plane.

4.1.5. Assessment of RQ5: What Are the Major Types of Attacks on SDN Planes Identified by Researchers?

In this section, we discuss the security threats that threaten SDN architecture. Moreover, Table 3 presents the classification of major security attacks that affect the SDN planes, i.e., application plane, control plane, and data plane. For more clarity on the taxonomy of attacks on SDN planes, see Figure 7.

DDoS Attacks

DoS/DDoS is the most challenging attack for SDNs. These attacks are intentional attempts to make network resources available to illegitimate users. In SDN architecture, the controller plays a vital role to determine the overall functionalities of the SDN network; therefore, the controller has become the central target for DoS/DDoS attacks. Some controllers that attract DoS/DDoS attacks have been identified, such as flood light-based controllers [123,124]. Shin et al. [125] described a DoS attack that exploits the separation logic of the control and data planes. Moreover, Fonseca et al. [126] suggested a secondary controller to deal with this problem, but the secondary controller is also vulnerable to these threats. Thus, multi-controller implementation is not a solution for DDoS because it leads to the failure of all controllers if the single controller fails [127].

Flooding Attack

Whenever an unmatched flow arrives at the virtual switch, it will send a request to the centralized controller to generate a forwarding rule for the new flow through the southbound interface. Attackers will send multiple packets to the virtual switch with spoofed IPs and compel the virtual switch to forward packet-in messages to the controller. The overflow of these fake flow requests overloads the controller and makes it inaccessible to legitimate users [128].

Flow Rule Conflicts

When open flow sends a request to the controller for a new flow rule, in response, the controller sends a new flow rule that is stored in the flow table of the switch. There is a time-out value for every flow rule and after that specified time, the switch evicts the old rules from the flow table [129]. Ternary content addressable memory has a very limited capacity to store flow table entries due to its high cost and power consumption [130]. Attackers overcome the switch by taking the advantage of this feature and sending fake flows to the

switch. These fake flows cause the flow table of the switch to run out of memory and store only fake rules, which erase the legitimate entries and degrade the performance.

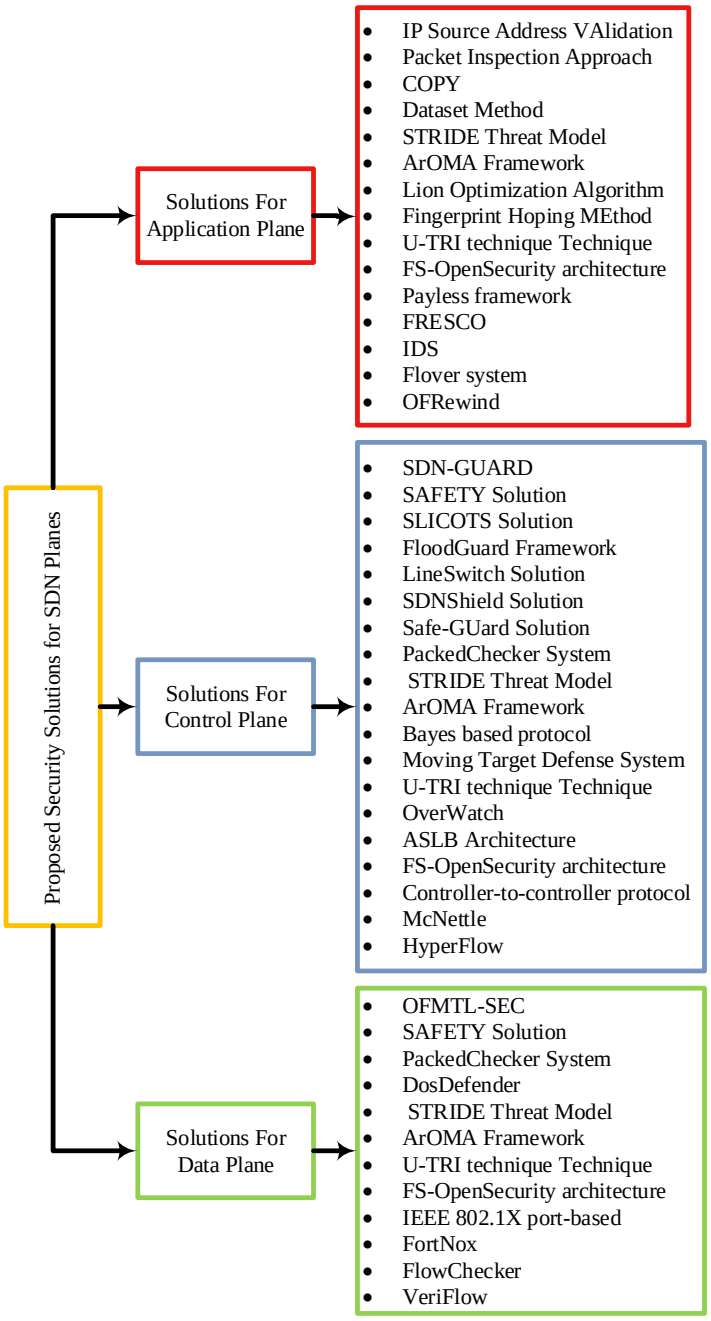

Figure 7. Proposed solutions for SDN planes.

Saturation Attacks

Saturation attacks degrade the network performance through controller saturation and buffer saturation:

Controller Saturation: When packet-in requests (i.e., fake flows) arrive at the controller, then the controller makes a queue to handle the packet-in requests. Moreover, if fake packets arrive in bulk, the controller will be busy handling fake requests, which eventually downgrades the performance of SDN-based networks [131].

Buffer Saturation: When a packet-in message is transmitted by the switch to the controller, the maximum part of that packet is stored in the buffer memory. An attacker takes advantage of this feature and sends fake packet-in messages to the switch, and this

leads to a buffer overflow. In this way, legitimate users are not able to process the flow requests and the attacker degrades the network performance without any difficulty.

Spoofing Attacks

In spoofing attacks, the actual identity of an attacker or traffic originator is hidden by forged network information, i.e., ARP, IP, MAC, etc. A user may use a spoofed addresses in order to access the network resources, which can be a part of a botnet designed to launch DDoS attacks. The most common spoofing threats are IP spoofing and ARP spoofing, which introduce larger attacks, such as smurf and TCP SYN flooding attacks [96,97].

ARP Spoofing: In ARP spoofing, the IP address of a legitimate user is linked to the attacker's MAC address. In this type of spoofing, traffic is hijacked from the originally anticipated receiver and knocked out the legitimate host or user from the network. Further, if SSL encryption is not specified in open flow, ARP poisoning may occur between switches and controllers. In ARP cache poisoning, the attacker exists in the same subnet of the victim network and uses the scanner to detect the network traffic among network components.

IP Spoofing: IP spoofing is creating many other security challenges on SDN planes, such as amplification and tampering. As the DNS directory associates an IP address with a domain name, by taking the advantage of this association, an attacker manipulates the DNS directory to reroute the traffic to illegal websites and launch flooding attacks.

Switch Spoofing: The IP address and control messages of the switch can be transmitted through the spoofed switch with a modified address. When a switch starts communicating with the controller by establishing a connection, at the same time, another malicious switch will activate and establish a connection with the controller. In this way, the controller will drop the connection with the legitimate switch and start communication with the malicious switch. This malicious communication can create fake requests and downgrade the performance of the network [124].

Tampering Attacks

Tampering is the unauthorized destruction or modification of network information, such as policies, access lists, flow rules, and topology. A malicious user may inject firewall rules or flow rules that will prohibit legitimate users and allow illegitimate users. Furthermore, attackers tamper with the topology information, which resultantly hijacks the traffic. Porras et al. [132] have described dynamic flow tunneling security challenges associated with conflicts in flow rules. These flows of tunneling security threats occur due to the evaluation of rules one by one. Attackers may organize multiple rules to violate the firewall rules because a single flow cannot violate the firewall rules.

Information Disclosure

Information disclosure security attacks never disrupt or destroy the network directly, they spy on network-sensitive information. Therefore, first of all, attackers will try to steal network information, such as communication between nodes or topology. The controller is the central location in SDN networks to control all switches of the entire network. Therefore, hackers can achieve a huge level of network access by invading the controller. Klöti et al. [133] described an attack scenario in which the attacker creates information about active flow rules. This is accomplished by defining the time between two connections. If the attempt of the second connection is faster when compared to the first, then an attacker may act as though the new flow rules. In this way, the attacker exploits the aggregation of flow in order to discover the flow table's content by observing the variances in the response time of the controller.

Scanning Attacks

An intruder is able to scan the SDN network remotely by sending probes to network IP addresses. When an adversary receives a response from its target point, it can be attacked or identified for a malicious purpose [21].

Cache Poisoning Attacks

A cache poisoning attack occurs due to the modification of flow rules in the flow table by an unauthorized user. Adversaries can add fake flow rules, which may lead to network failure. Hong et al. [7] presented an attack scenario in which an attacker produces fake LLDP packets by injecting a forged LLDP injection. Fake LLDP packets are produced into the open flow network to clarify the internal bogus links between two switches. By identifying the flow traffic from open flow switches, an attacker can access the actual LLDP packet.

Control Channel Hijacking

Hijacking occurs when a compromised open flow switch isolates itself from the authorized controller and connects with a malicious controller [134]. After connecting with the malicious controller, it redirects the traffic of the control channel to the malicious controller and spoofs messages to the legitimate controller.

Weak Authentication and Communication Attacks

Weak authentication and lack of trust between the controller and applications generate man-in-the-middle attacks and spoofing attacks at northbound and southbound APIs. Furthermore, inappropriate authentication may lead to malicious or unauthorized access to different applications. In this way, an attacker can easily eavesdrop on flows in order to analyze what kind of flows are implemented and what type of traffic is allowed through the network [89].

Cyber Attacks

SDN is one of the most exciting domains for cyber-attacks due to the programmability and centralized nature of these networks. Therefore, cyber security has become a major challenge in the SDN environment for ensuring the continuity of required service [120–122].

4.1.6. Assessment of RQ6: What Are the Major Causes of Attacks on SDN Planes Addressed in the Literature?

Researchers and security experts believe that typical SDN security threats involved are controller vulnerabilities, malicious applications, data leakage and modification, legality and consistency of flow rules, scalability, and configuration issues. We analyze the identified security threats and attack causes. The results of security analysis with respect to their affected SDN planes are shown in Table 3. From classification Table 3, it can be noted that the attack challenges of different planes are different. In this section, we discuss the causes of major security issues that exist on SDN planes.

Authorization and Authentication

Due to the centralized control nature of SDN architecture, the authentication and authorization of applications are major security issues in programmable networks. In open flow, control plane functionalities depend upon the applications running on the controller, which are developed by a third party rather than controller vendors. These applications restrict access to network resources and manipulate network behavior through malicious activities [135]. In this way, an attacker can easily gain access to network resources and impersonate the application/controller to influence network operations.

Data Modification

As previously discussed, the controller can reprogram the network devices in order to control the traffic flow in SDN. Therefore, an attacker can control the whole network by hijacking the controller and modifying or inserting flow rules in network devices. In this regard, the attacker generates a man-in-the-middle attack by intercepting messages between two victims to inject or alter messages into a communication channel. The FlowVisor

approach was identified by Jarschel et al. [136], which allows the attacker to modify data on communication entities.

Threats from Malicious Applications

In SDN architectures, applications are implemented on top of the control plane, which opens the control plane to detrimental security attacks. Controller security is very challenging due to the integration of third-party applications in SDN architecture [137]. Therefore, malicious applications have a serious impact on the network as a compromised controller. Moreover, a buggy or poorly designed application also generates new security threats to the system.

Scalability Issues

Jarschel et al. [136] investigated whether implementations of the new controller are able to handle the high volume of flows during the integration of open flow with high-speed networks. Therefore, a lack of scalability introduces saturation attacks on the control plane, which is more detrimental in SDN architecture when compared to traditional networks [137].

Side-Channel Attack

As SDNs constantly carry confidential as well as private information, where each execution has different attributes, such as the time attribute [138]. At this level, an attacker may attack by using side-channel attacks to infer network state-related information. Side-channel attacks not only affect the integrity, confidentiality, or availability of data but also trigger further attacks. Another open security challenge in the SDN framework is the security of stored credentials, such as certificates and keys. In the past, cross-virtual machine side-channel attacks occurred in cloud computing where malicious virtual machine identified a vulnerable virtual machine to extract secure information [139].

Configuration Issues

In order to detect network vulnerabilities, a number of security protocols and policies were developed. These policies and protocols were implemented on SDN planes for security purposes. If these security policies were implemented without understanding the security rules of the deployment scenario, there will be configuration issues.

Legality and Consistency of Flow Rules

Consistency and legality of flow rules is an incorrect or malicious flow rule injection at the data layer. In flow rules, consistency consists of three processes, i.e., generation process, the release process, and the update process [139]. The generation process overrides the flow rules by using multiple applications, whereas in the release process, malicious tampering or transmission delays cause the flow rules to be inconsistent between switches and controllers. Moreover, the update process originates the synchronization of flow rules among switches.

Data Leakage

A variety of possible actions in open flow switches has been described for packet handling including dropping, forwarding, and sending to the controller. An attacker analyzes the applied action of the specific packet through the timing analysis of packet processing. For example, the processing time of a redirected packet toward the controller will be longer when compared to a packet that is directly passed from an input port to an output port. Therefore, an attacker can discover the reactive/proactive configuration of the switch. After discovering the packet type, the attacker generates a fake or malicious flow request, which leads to a DoS attack [140].

Poor Deployment of the Controller

Network attacks overload the control layer by dominating it with malicious activities. Due to single points of failure and inefficiency in large-scale networks the researchers utilize multi-controller implementation schemes. However, in multi-controller implementation, the load of a failed controller is reassigned to another active controller [141]. Additionally, in a multi-controller scheme, the whole network is divided into multiple sub-networks, which leads to network privacy problems and consistency issues. An attacker hijacks the controller by using the admin station and rejects the user's legitimate request. However, attackers can use logical or physical methods to introduce severe attacks, which destroy the entire network. SDN security administrators can manage and configure the network using top-most layer applications, but this configuration introduces a new threat interface for unauthorized applications. Due to the centralized nature of the controller, SDN attacks will spread quickly to disrupt the entire network [142].

Fault and Power Analysis Attacks

A fault attack is the intentional manipulation of an integrated circuit with the objective of generating an attack within integrated circuits to push the device into an unintended state [143]. The objective of a fault attack is to access critical information and disable the internal mechanism of protection. However, power analysis is the form of a side-channel attack that enables the attacker to the consume power of a hardware device, which is cryptographic. Sun et al. [144] proposed the on-chip hybrid voltage scheme to enhance the cryptographic security of a circuit in order to overcome the power analysis attack by reducing the performance overhead. Furthermore, multiple fault detection architectures have been proposed for Ring-LWE by implementing FPGA [145–147].

4.1.7. Assessment of RQ7: What Proposed Solutions Have Been Implemented to Address Security Attacks on SDN Planes?

SDNs provide a dynamic security rule to define the security policy, implement the defined policy to network elements, and minimize the misconfiguration chances and policy conflicts. Due to the nature of global network visibility, multiple security systems, such as intrusion detection/prevention systems and firewalls, are implemented on specified traffic. In this section, we discussed the security measures, platforms, architectures, and proposed solutions for the application plane, control plane, and data plane. The identified security solutions from selected studies are shown in Figure 7.

Security Solutions on the Application Plane

In SDN architecture, the controller creates an intermediate layer among applications and network hardware to hide the complexity of the network from applications. The post-quantum cryptographic (PQC) method is one of the best solutions to secure communication devices in different technologies like IoT and SDN [147]. Post-quantum networking provides an advanced-level impact on communication networks for security and evaluates the existing protocols by performing the enhancement. Furthermore, it also evaluates the performance of PQ algorithms and overcomes through new designs [148–152]. Payless is an efficient, low cost, and flexible application-monitoring framework for SDN architecture [152]. The Payless framework is more efficient due to its proposed algorithm for flow statistics collection. The proposed algorithm uses a variable frequency technique for flow statistics collection to decrease the polling frequency for flows. This approach creates stability among network overheads and accuracy of statistics.

An open-flow security application framework FRESCO has been proposed to facilitate the modular composition of open flow-based security services [153]. The modular composition of this framework comes from challenges in security services, information deficiency, and attack response translation. FRESCO has the ability to reprogram the network infrastructure in order to secure the network from developing attacks. It has a scripting API that allows the developers to insert intrusion detection and monitoring algorithms, such as

libraries. These algorithms can be integrated with other components to create more complex security applications. An intrusion detection system was proposed by Seeber et al. [154] to countermeasure the malicious traffic over the network. On the basis of required flow rules, SDN switches perform as lightweight intrusion detection systems. These switches collect information and report it to the controller for further analysis to detect malicious traffic. Nygren et al. [155] also proposed an architecture called anomaly-based intrusion, which is integrated as an open flow switch for malicious attack detection.

Moreover, a cloud-based solution is proposed for the detection and mitigation of DDoS attacks, which discussed alert generation and auto-correlation methods [156]. The Flover system was proposed for checking open flows, which authenticates whether flow policies are conflicting with the network's security policies [155]. This system is deployed as an open flow application on the controller to measure the consistency of new flow rules with an existing set of specified attributes. Further, the ndb framework acts as a debugging tool to detect the root causes of bugs in SDN architecture [157]. Another framework, OFRewind, traces the network anomalies by recording and replaying the selected traffic [158]. OFRewind and ndb frameworks can also be used to detect faulty applications that induce malicious security threats.

The proposed architectures, platforms, and frameworks discussed above help in improving and developing more security applications as well as providing robust security to the application plane from malicious attacks. However, the literature shows that there is only a very small effort being made to enhance the security of application data. Apart from this, there is no proper mechanism for a distinction between user applications, third-party applications, and network applications. Moreover, an accountability and access control mechanism for nested applications in SDNs has not yet been defined.

Security Solutions on Control Plane

In SDN architecture, applications obtain network resources and information through the control plane; therefore, the security of the control plane is highly recommended. Further, the control plane must be secure from faulty or malicious applications and ensure the access of legitimate applications with respect to their functional requirements. An extended version of the floodlight controller called the security-enhanced (SE) floodlight controller provides exceptional security on the control plane [159]. The SE-floodlight controller adds a secure northbound API into the controller, which acts as an arbitrator between the data plane and applications. The SE controller has introduced an open flow verification module for the application, which integrates the class modules to produce the flow rules. In the open flow standard, the controller installs unique rules for every client connection, which leads to the installation of a large number of flows in switches and a high load on the controller. Therefore, multiple solutions and techniques are suggested to reduce the load on the controller or enhance the memory and processing power of the controller. The McNettle controller has multiple central processing unit cores to support and scale the control algorithm [160]. McNettle can be enhanced by implementing a high-level programming language. However, to maximize the processing performance of the controller in order to achieve higher availability and scalability, parallelism has been proposed through multi-core processors [161]. Moreover, to provide the functionalities of the control plane to overlay heterogeneous, distributed networks, a DISCO solution has been presented [162]. It consists of two components, i.e., inter-domain and intra-domain. The inter-domain module manages the communication between controllers, which consists of agents and a messenger, whereas the intra-domain components enable the network monitoring for the controller to calculate paths of priority flows.

HyperFlow is a logically centralized and physically distributed control platform that enables network operators to implement multiple controllers [163]. HyperFlow has the capability of local decision making, which enhances control scalability and reduces the flow setup time. DDoS or DoS attacks can be alleviated by analyzing the flow behavior and statistics accumulated in open flow switches. Braga et al. [164] use three components,

i.e., classifier, feature extractor, and flow collector. The flow controller collects flow entries from flow tables during fixed intervals. The fixture extractor component extracts those features that are necessary for DDoS attacks and forwards them to the classifier. Then, the classifier analyzes the extracted features by using SOM to classify traffic either normal traffic or malicious traffic [165].

The literature reviewed in this research has demonstrated that separation of the control plane and the data plane is not practical when it is essential to implement security services on a large scale and in cloud environments and datacenters due to specific shortcomings. Furthermore, all kinds of flows from switches must be forwarded to the controller in large-scale networks such as high-volume traffic that can cause a performance bottleneck. Additionally, multiple security applications, including a stateful firewall, requires historical network information in order to avoid communication overheads.

Security Solutions on Data Plane

In the data plane, malicious applications can install, modify, or change the flow rules in the data path; therefore, it must be protected from such malevolent applications. However, fine-grained mechanisms such as authorization and authentication are implemented to protect data from those applications that can modify or alter flow rules. FortNox is a platform that activates the NOX open flow controllers to evaluate the flow rule conflicts and authorize the applications [166]. FortNox provides security constraints through software extension and role-based authentication via digital signatures in a FortNox open flow controller. Moreover, the FlowChecker tool identifies inconsistencies in open flow rules inside the single switch or within data path elements [167]. FlowChecker is also deployed as a master controller or as an open flow application to analyze and validate the end-to-end configurations during run time. The VeriFlow tool is utilized to identify the faulty rules injected by malicious applications and to prevent the network from anomalous network behavior [25]. Further, the open flow protocol also provides the flexibility to configure the secondary connection along with the backup controller if the first controller fails. Zhang et al. [168] demonstrated that the length of the path between the controller and switch is directly proportional to the connectivity loss. Hence, the length of switches and controllers must be short enough to enhance the performance of the system, to improve the availability of content, and to make the security analysis fast. In addition, the SPHINX framework also detects known and unknown attacks on the data plane without assuming that forwarding devices are trusted or not [169]. It identifies and alleviates the attacks originating from malicious devices by isolating network operations with flow graphs and pre-defined security rules defined by the administrator.

Proper segmentation and network planning help to enhance the resilience of switches and increase the connectivity of controllers on the data plane. Different studies investigated whether the consistent connectivity among the controller and OpenFlow switch can protect the network from saturation attacks. Hence, this connectivity not only improves the performance of the system but also implements fast restoration and security analysis.

4.2. Quality Assessment Score

The quality assessment score is presented in Table 4, where 4% of selected articles hold an average score, whereas 93% are above average. Only 3% of the selected papers are below average. The defined quality assessment criteria can help researchers and security analysts to select SDN-related research to make their network system more secure and robust.

Table 4. Quality assessment score.

Reference	Score	Total
[56,96,97,100,101]	5	4
[19,20,51,52,54,57,58,60,63,78,80,82,85,95,102,105,112,114,115,118,119]	4.5	21
[16,49,53,64,65,71,83,84,86,91,93,94,99,103,104,108,111,120,121]	4	20
[14,21,48,50,55,66–68,70,83,88–90,92,95,98,106,110,116,117,122]	3.5	24
[69,72,74–77,79,90,109,113,117]	3	11
[17,73,96]	2.5	3
[81,87]	2	2
[107]	1.5	1

5. SDN Security Threat Model

The decoupling of data and control plane offers a defense solution to mitigate the multiple security threats via its programmability features. Still, there are many new security attacks that arise during the implementation of SDN. Furthermore, SDNs are frequently attacked due to their huge dependency on software/programs. Therefore, it jeopardizes the entire network system by making it easy for attackers to enter into the system. As we already discussed, SDNs are vertically divided into three planes, which are vulnerable to different security attacks, the most common of which are DDoS attacks. These vulnerabilities are discussed in classification Table 3. However, in this section, we propose a security model by identifying the location of typical security attack occurrences on SDN planes, as shown in Figure 8.

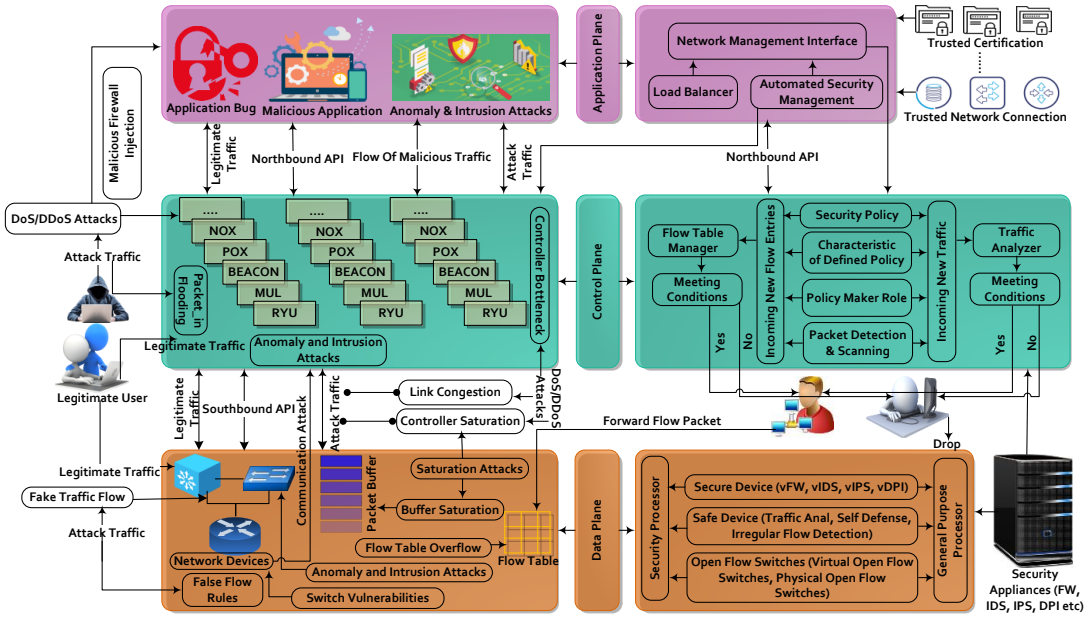

Figure 8. A collaborative security model for SDN planes.

5.1. Attack Scenario on SDN Planes

The explanation of each security threat shown in Figure 8 is given below. In SDN architecture, the entire network is controlled via software and a centralized controller, which is the basis of multiple security challenges. The most common software-related security attacks that occur on the application plane are intrusion attacks, anomaly attacks, bugs, and malicious application injection. The reason behind these security threats is the lack of open APIs standards for applications to manage and control the network services and functionalities through the control plane [128,170]. Moreover, the applications that are

deployed on this plane are developed by third parties that have the privilege of accessing and manipulating network resources [171]. Hence, in order to make the network secure, it is necessary to authenticate every request, which is very challenging due to the large number of applications.

Furthermore, the control plane is a highly targeted point for attackers due to its centralized control nature. Generally, the controller on this plane is responsible for the authentication of applications as well as the authorization of resources required for applications [135]. Therefore, it is necessary to separate the applications with respect to their security checks before allocating access to resources. To fulfill the responsibility of application authorization, there is a need for customized security checks. Such customization has not yet been developed [172]. Therefore, these applications introduce malicious threats to the control plane, for example anomaly and intrusion attacks. Additionally, the centralized controller also needs to implement the flow rules for all new flows in the data path.

In this way, the controller becomes a bottleneck due to a large number of new flows, which minimizes the controller scalability and maximizes the DoS/DDoS attacks [37]. The responsibility of network switches on the data plane is packet-forwarding, whereas rulemaking is performed on the control plane. Therefore, the identification of legitimate flow rules is the foremost security issue for the data plane. Every switch maintains the flow rules, but due to the limited memory, there are a large number of flows that lead to saturation attacks, for example buffer saturation.

Moreover, flow rules stay in the flow table for a specific length of time, which leads to the launch of DDoS attacks [173]. Due to this delay, the attacker sends multiple fake packets to the switch and the switch requests a new rule for every packet. In this way, new entries replace all old entries in the flow table and the memory becomes full of fake flow rules. Thus, legitimate entries are dropped due to space unavailability in the flow table. Furthermore, the switch also requests new flow rules for new traffic flow by using the packet-in message and receives a reply from the controller by using the packet-out message, which generates high traffic on the link and introduces link congestion. On the other hand, the virtual switch receives multiple packets with spoofed IPs and compels the virtual switch to forward packet-in messages, which overload the controller to generate packet-in flooding attacks. The defense solution for these attacks is shown in Figure 8 in the section below.

5.2. Search Strategy Proposed Security Solution against Attacks on SDN Planes

After reviewing and identifying the SDN security challenges on the application plane, control plane, and data plane in Section 2, we can see that SDN security challenges are not single-tier problems. Hence, the security issues in SDN architecture involve each plane of the entire system. Therefore, we need to propose a security solution that covers all aspects of security on each plane. In this regard, we proposed an ideal security model to mitigate these attacks. An overview of the proposed SDN security model is presented in Figure 8.

5.2.1. Solution against Attacks on the Application Plane

The application plane should set up a trusted network connection to achieve identity authentication on multiple components, i.e., platforms and terminals. After authentication verification, it evaluates the confidentiality level of the terminal platform. If evaluated results meet the security requirement, then it is acceptable for the terminal to access the network.

5.2.2. Solution against Attacks on the Control Plane

In SDN architecture, the controller receives the predefined network policies from logical functions, such as load balancer and automated security management, through the network management interface. Generally, the network management interface is an internal interface that may be more authenticated and trustable when compared to the northbound API. An automated security management function detects and mitigates the

security attacks on the SDN controller. To provide security services at the control plane, a number of security components, such as the security policy component, the characteristics of defined policy components, the policy maker role component, packet scanning, and the detection component are implemented. The security policy component is configured to collect multiple policies from the application plane through the northbound interface and the network management interface. In order to confirm the security of received policies, these policies are verified through the authorization and authentication module of the SDN controller. The characteristic of the defined policy component is configured to evaluate whether the policy will be sent from the controller to switch or not.

The policymaker role component is designed to identify the role of the policymaker. Policymakers may be security administrators, general administrators, users, or agents. The packet scanning and detection component are configured to detect whether flows meet the conditions configured by applications or administrators. The flow table manager and traffic analyzer are configured to manage the searching, updating, adding, and deleting of the new traffic and flow entries from flow tables. The flow table manager and traffic analyzer investigate whether incoming flow entries are inserted correctly into flow tables. For example, the flow table manager identifies the conflict between old and new flow entries before inserting them into the flow table. If yes, then only new flow entries are allowed to replace the previous conflicting flow entries by following the policymaker's rules. In this way, the flow table stores flow entries for SDN switches.

5.2.3. Solution against Attacks on the Data Plane

In the data plane, it is necessary to implement an independently developed security processor for the encryption and decryption process. To ensure the confidentiality of data, we encrypt the data transmission method and utilize a hardware decoupling procedure to build a separate area in the memory. The utilized hardware consists of the secure device (vFW, vIDS, vIPS, vDPI) and a safe device (traffic control, self-defense, irregular flow detection) open flow switches (virtual open flow switches, physical open flow switches).

6. Discussion

After an extensive review, we identified multiple research gaps and suggested future research directions by considering these research gaps.

6.1. Research Gaps

In order to secure SDN architecture, a number of security solutions have been proposed by researchers to secure the networks, but there are still many security challenges in SDNs that need to be addressed. In this section, we discussed the identified research gaps shown in Table 5.

Table 5. Research gaps.

Reference	Identified Gaps
[14,16,18,19,48,49,51,52,58,62,64,74,76]	After conducting a comprehensive review of SDN security issues, it has been demonstrated that most researchers provide security solutions for the control plane against DDoS attacks. Only a few studies presented security solutions for the data plane and application plane. Therefore, the provision of security solutions for other planes and SDN switches is an open research gap.
[70]	Although flow tables are easy to use to mitigate low-rate DoS attacks, they influence the effectiveness of those strategies, which are implemented to overcome low-rate DoS attacks. So, the efficient implementation of flow rules is a significant research gap.
[71]	Khamaiseh et al. [71] identified saturation attacks by using supervised and unsupervised classifiers with a single controller. However, the approach is unrealistic for detecting known and unknown attacks for multiple controllers where the SDN environment is on a large scale.
[75]	Due to the continuous growth of SDNs, applications also grow rapidly, which puts increasing pressure on controllers. This leads to the development of load time as well as the CPU utilization of controllers. Therefore, a programmable switch is required to decrease the load of the controller and increase the throughput.

Table 5. *Cont.*

Reference	Identified Gaps
[174,175]	Many researchers implemented a single controller in their proposed solution. However, if the deployed controller is not stable or secure, this may lead to the failure of the entire network. To overcome this challenge, multi-controller implementation could be the best choice.
[176]	There are only a few studies in which flash event traffic is identified. Jiang et al. [176] identified this security threat by using a small network topology that consists of 11 hosts and making their proposed technique unrealistic.
[177–179]	The information-based theory uses the predefined values of the threshold for anomaly detection. SDN is not yet implemented publicly; therefore, it is challenging to measure the behavior of the correct baseline.
[180]	Many researchers utilize virtual environments to authenticate their proposed defense approaches by using the Mninet emulator. This virtualization process drastically affects the results because the utilization of simulation tools with modeling of the internet is very complicated. To date, no solution has been proposed which represents internet behavior.
[181–183]	Researchers embedded many security modules to increase the performance of SDN switches. Although these security solutions reduce the communication overhead between SDN planes and minimize the controller computation overheads, they increase the cost as well as the complexity of network devices. Hence, the implementation of efficient security solutions in switches to reduce communication overheads is a considerable research gap.
[184]	Wang et al. [184] proposed an architecture by using multiple controllers to validate their proposed defense solution. Although the proposed solution enhances the performance of the network, the synchronization overheads of the controller are not considered. Therefore, efficient synchronization also an open research gap.
[52,183]	Many researchers used network traffic elements for DDoS attack detection in SDN and use open statistic techniques to collect network features, whereas the method of collection of network features by using open flow statistics increases the processing overheads for the control plane. Therefore, network statistics collection with minimal overheads is a challenging research gap.
[56,185]	The unavailability of benchmarked data to model the attack traffic and normal traffic is also another open research issue. However, experts use multiple traffic generator tools to represent normal traffic, but they are not productive. Hence, the prediction of an accurate baseline is also an open research gap in existing solutions.
[64,80,81,83–85,90,95,104,116]	In many studies, a single virtual machine (VM) has been used to perform the experimental setup. However, results obtained from simulation or emulation environments will be different from real scenarios due to the limited availability of resources in a single VM. Although some researchers have implemented real switches, they have adopted small topologies to perform the experiment. Hence, real-time implementation to validate the research results is a challenging research gap.
[67,69,70,74]	In many studies, we investigated whether a local or an artificial traffic generator tool was used to generate normal traffic and attack. On the other hand, in some studies, a real dataset was used, containing only malevolent or malicious traffic and normal traffic was generated through other tools. These tools are not capable of generating traffic in real ways. So, testing with an actual dataset that contains malicious traffic is another considerable research gap.
[75]	There is a need to propose an efficient architecture or framework for the detection of network intrusion-detection systems by using complex deep learning algorithms. In this way, intruders will be easily detected on SDN planes by using intrusion-detection frameworks. There is a need to apply machine learning in a constrained scenario, and the use of multicast routing could be highly valuable. Moreover, it is necessary to consider network topologies, traffic patterns, and networking changes for future analysis.

6.2. Future Directions

Although a number of security solutions have been proposed to make SDN architecture more secure, but further extension of proposed solutions is possible. In this section, we recommended the directions for future work.

After presenting a detailed discussion on threats to SDNs' security in Section 4, we investigated whether the security problem in SDN is a single-tier problem. Security threats exist on each layer in SDN architecture; therefore, to perform research on each layer individually is far from enough. Hence, a perspective system is needed to make the entire network robust and secure.

The frequency of DDoS attacks in SDN architecture is at the top of all other security challenges. Therefore, it is highly recommended to develop a solution for DDoS attack detection and mitigation to minimize the overall overheads of a centralized controller.

Transport layer security (TLS) alleviates multiple security threats with mutual verification on the control switch. Therefore, TLS specification is mandatory between controllers and switches to secure transmitted data and links.

The integrity verification of software applications protects the network from malicious software attacks. Additionally, for integrity verifications, unambiguous malware detection approaches must also be developed. Moreover, malicious applications also introduce significant security risks. Hence, third-party applications must be scanned to protect the network from malicious code and other vulnerabilities. Thus, security solutions for third-party applications are an ongoing challenging research area.

7. Conclusions

A systematic review has been presented by providing a comprehensive discussion based on collected studies available in the field of SDN security and privacy issues. In this research, we highlighted the security issues of the application plane, control plane, and data plane of SDNs and presented a taxonomy of attacks. We also identified the causes of attacks on the basis of their impacts. Thereafter, we summarized the existing security solutions for these planes that were proposed by researchers. Based on the identified security issues and solutions, a collaborative security model was proposed. Then, we presented some ongoing security challenges and gaps on SDN planes. Lastly, we provided suggestions for future research that may be beneficial for researchers to mitigate security attacks on SDN layers. Such an extensive systematic review will surely help researchers and policymakers to provide more reliable and robust security solutions in SDNs.

Author Contributions: M.S.F. led this research work and statistically analyzed the whole article. This manuscript presents the attacks taxonomy and security model that are designed to investigate the security attacks, challenges and proposed solutions for SDN planes. S.R. has supported for the problem statement refinement with state-of-the-art literature review. Moreover, S.R. and A.A. has contributed in the selection of frequently used open-source software in the field of SDN. M.S.F. and A.A. have made contribution in designing methodology and proof reading of manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: This is a review study; therefore, the data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors gratefully acknowledge the continuous support of senior research fellows for their expertise and assistance throughout all aspects of our study and for their help in writing the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Raghavan, B.; Casado, M.; Koponen, T.; Ratnasamy, S.; Ghodsi, A.; Shenker, S. Software-defined internet architecture: Decoupling architecture from infrastructure. In Proceedings of the 11th ACM Workshop on Hot Topics in Networks, Redmond, WA, USA, 29–30 October 2012; pp. 43–48.
2. Jain, S.; Kumar, A.; Mandal, S.; Ong, J.; Poutievski, L.; Singh, A.; Venkata, S.; Wanderer, J.; Zhou, J.; Zhu, M.; et al. B4: Experience with a globally-deployed software defined WAN. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 3–14. [CrossRef]
3. Natarajan, S.; Ramaiah, A.; Mathen, M. A software defined cloud-gateway automation system using OpenFlow. In Proceedings of the 2013 IEEE 2nd International Conference on Cloud Networking (CloudNet), San Francisco, CA, USA, 11–13 November 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 219–226.
4. Patel, P.; Bansal, D.; Yuan, L.; Murthy, A.; Greenberg, A.; Maltz, D.A.; Kern, R.; Kumar, H.; Zikos, M.; Wu, H.; et al. Ananta: Cloud scale load balancing. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 207–218. [CrossRef]
5. Yungaicela-Naula, N.M.; Vargas-Rosales, C.; Pérez-Díaz, J.A.; Zareei, M. Towards security automation in software defined networks. *Comput. Commun.* **2022**, *183*, 64–82. [CrossRef]
6. Jammal, M.; Singh, T.; Shami, A.; Asal, R.; Li, Y. Software defined networking: State of the art and research challenges. *Comput. Netw.* **2014**, *72*, 74–98. [CrossRef]
7. Hong, S.; Xu, L.; Wang, H.; Gu, G. Poisoning network visibility in software-defined networks: New attacks and countermeasures. In Proceedings of the NDSS, San Diego, CA, USA, 8–11 February 2015; Volume 15, pp. 8–11.
8. Kreutz, D.; Ramos, F.M.; Verissimo, P.E.; Rothenberg, C.E.; Azodolmolky, S.; Uhlig, S. *Software-Defined Networking: A Comprehensive Survey*; IEEE: Piscataway, NJ, USA, 2014; Volume 103, pp. 14–76.

9. Lee, S.; Yoon, C.; Lee, C.; Shin, S.; Yegneswaran, V.; Porras, P.A. DELTA: A Security Assessment Framework for Software-Defined Networks. In Proceedings of the NDSS, San Diego, CA, USA, 26 February–1 March 2017.
10. Lee, S.; Kim, J.; Woo, S.; Yoon, C.; Scott-Hayward, S.; Yegneswaran, V.; Porras, P.; Shin, S. A comprehensive security assessment framework for software-defined networks. *Comput. Secur.* **2020**, *91*, 101720. [CrossRef]
11. Voellmy, A.; Kim, H.; Feamster, N. Procera: A language for high-level reactive network control. In Proceedings of the First Workshop on Hot Topics in Software Defined Networks, Helsinki, Finland, 13 August 2012; pp. 43–48.
12. Dhamecha, K.; Trivedi, B. SDN Issues A Survey. 2013. Available online: https://www.researchgate.net/publication/269667437_SDN_Issues_A_Survey (accessed on 5 June 2023).
13. Kreutz, D.; Ramos, F.M.; Verissimo, P. Towards secure and dependable software-defined networks. In Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking, Hong Kong, China, 16 August 2013; pp. 55–60.
14. Deepa, V.; Sudar, K.M.; Deepalakshmi, P. Detection of DDoS attack on SDN control plane using Hybrid Machine Learning Techniques. In Proceedings of the 2018 International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 13–14 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 299–303.
15. Aladaileh, M.A.; Anbar, M.; Hasbullah, I.H.; Chong, Y.W.; Sanjalawe, Y.K. Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking Controller—A Review. *IEEE Access* **2020**, *8*, 143985–143995. [CrossRef]
16. Celesova, B.; Val'ko, J.; Grezo, R.; Helebrandt, P. Enhancing security of SDN focusing on control plane and data plane. In Proceedings of the 2019 7th International Symposium on Digital Forensics and Security (ISDFS), Barcelos, Portugal, 10–12 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6.
17. Benzaïd, C.; Boukhalfa, M.; Taleb, T. Robust Self-Protection Against Application-Layer (D)DoS Attacks in SDN Environment. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea, 25–28 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–6.
18. Priya, P.M.; Manjula, K.R. Cog-SDN: Mitigation Mechanism for Distributed Denial of Service Attacks in Software Defined Networks. In Proceedings of the International Conference on Applications and Techniques in Information Security, Tamil Nadu, India, 22–24 November 2019; Springer: Singapore, 2019; pp. 202–215.
19. Hameed, S.; Ahmed Khan, H. SDN based collaborative scheme for mitigation of DDoS attacks. *Future Internet* **2018**, *10*, 23. [CrossRef]
20. Novaes, M.P.; Carvalho, L.F.; Lloret, J.; Proença, M.L., Jr. Adversarial Deep Learning approach detection and defense against DDoS attacks in SDN environments. *Future Gener. Comput. Syst.* **2021**, *125*, 156–167. [CrossRef]
21. Schehlmann, L.; Abt, S.; Baier, H. Blessing or curse? Revisiting security aspects of Software-Defined Networking. In Proceedings of the 10th International Conference on Network and Service Management (CNSM) and Workshop, Rio de Janeiro, Brazil, 17–21 November 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 382–387.
22. Abdulkarem, H.S.; Dawod, A. DDoS Attack Detection and Mitigation at SDN Data Plane Layer. In Proceedings of the 2020 2nd Global Power, Energy and Communication Conference (GPECOM), Izmir, Turkey, 20–23 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 322–326.
23. Pradhan, A.; Mathew, R. Solutions to Vulnerabilities and Threats in Software Defined Networking (SDN). *Procedia Comput. Sci.* **2020**, *171*, 2581–2589. [CrossRef]
24. Hu, T.; Guo, Z.; Yi, P.; Baker, T.; Lan, J. Multi-controller based software-defined networking: A survey. *IEEE Access* **2018**, *6*, 15980–15996. [CrossRef]
25. Al-Shaer, E.; Al-Haj, S. FlowChecker: Configuration analysis and verification of federated OpenFlow infrastructures. In Proceedings of the 3rd ACM Workshop on Assurable and Usable Security Configuration, Chicago, IL, USA, 4 October 2010; pp. 37–44.
26. Nara, R.; Satoh, K.; Yanagisawa, M.; Ohtsuki, T.; Togawa, N. Scan-based side-channel attack against RSA cryptosystems using scan signatures. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2010**, *93*, 2481–2489. [CrossRef]
27. Ristenpart, T.; Tromer, E.; Shacham, H.; Savage, S. Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds. In Proceedings of the 16th ACM Conference on Computer and Communications Security, Chicago, IL, USA, 9–13 November 2009; pp. 199–212.
28. Xu, J.; Hong, H.; Lin, G.; Sun, Z. A New Inter-Domain Information Sharing Smart System Based on ABSES in SDN. *IEEE Access* **2018**, *6*, 12790–12799. [CrossRef]
29. Canto, A.C.; Kaur, J.; Kermani, M.M.; Azarderakhsh, R. Algorithmic Security is Insufficient: A Comprehensive Survey on Implementation Attacks Haunting Post-Quantum Security. *arXiv* **2023**, arXiv:2305.13544.
30. Oktian, Y.E.; Lee, S.; Lee, H.; Lam, J. Secure your northbound SDN API. In Proceedings of the 2015 Seventh International Conference on Ubiquitous and Future Networks, Sapporo, Japan, 7–10 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 919–920.
31. Vasconcelos, C.R.; Gomes, R.C.; Costa, A.F.; da Silva, D.D. Enabling high-level network programming: A northbound API for Software-Defined Networks. In Proceedings of the 2017 International Conference on Information Networking (ICOIN), Da Nang, Vietnam, 11–13 January 2017; pp. 662–667.
32. Feng, M.; Xu, Z.; Wang, C. SDN-based Satellite Networks and Southbound Interface Protocol Extension. *Radio Commun. Technol.* **2017**, *43*, 19–23.
33. Hyun, S.; Kim, J.; Kim, H.; Jeong, J.; Hares, S.; Dunbar, L.; Farrel, A. Interface to network security functions for cloud-based security services. *IEEE Commun. Mag.* **2018**, *56*, 171–178. [CrossRef]

34. Giesen, F.; Kohlar, F.; Stebila, D. On the security of TLS renegotiation. In Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, Berlin, Germany, 4–8 November 2013; pp. 387–398.
35. Tschofenig, H.; Fossati, T. Transport layer security (tls)/datagram transport layer security (dtls) profiles for the internet of things. In *RFC 7925*; Internet Engineering Task Force: Fremont, CA, USA, 2016.
36. Scott-Hayward, S.; Natarajan, S.; Sezer, S. A survey of security in software defined networks. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 623–654. [CrossRef]
37. Ahmad, I.; Namal, S.; Ylianttila, M.; Gurtov, A. Security in software defined networks: A survey. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2317–2346. [CrossRef]
38. Shaghaghi, A.; Kaafar, M.A.; Buyya, R.; Jha, S. Software-defined network (SDN) data plane security: Issues, solutions, and future directions. In *Handbook of Computer Networks and Cyber Security: Principles and Paradigms*; Springer: Cham, Switzerland, 2020; pp. 341–387.
39. Rahouti, M.; Xiong, K.; Xin, Y.; Jagatheesaperumal, S.K.; Ayyash, M.; Shaheed, M. SDN Security Review: Threat Taxonomy, Implications, and Open Challenges. *IEEE Access* **2022**, *10*, 45820–45854. [CrossRef]
40. Alsmadi, I.M.; AlAzzam, I.; Akour, M. A systematic literature review on software-defined networking. In *Information Fusion for Cyber-Security Analytics*; Springer: Cham, Switzerland, 2017; pp. 333–369.
41. Ali, T.E.; Chong, Y.W.; Manickam, S. Machine Learning Techniques to Detect a DDoS Attack in SDN: A Systematic Review. *Appl. Sci.* **2023**, *13*, 3183. [CrossRef]
42. Singh, J.; Behal, S. Detection and mitigation of DDoS attacks in SDN: A comprehensive review, research challenges and future directions. *Comput. Sci. Rev.* **2020**, *37*, 100279. [CrossRef]
43. Alhaj, A.N.; Dutta, N. Analysis of security attacks in SDN network: A comprehensive survey. In *Contemporary Issues in Communication, Cloud and Big Data Analytics*; Springer: Singapore, 2022; pp. 27–37.
44. Zhang, H.; Cai, Z.; Liu, Q.; Xiao, Q.; Li, Y.; Cheang, C.F. A survey on security-aware measurement in SDN. *Secur. Commun. Netw.* **2018**, *2018*, 2459154. [CrossRef]
45. Hussein, A.; Chadad, L.; Adalian, N.; Chehab, A.; Elhajj, I.H.; Kayssi, A. Software-Defined Networking (SDN): The security review. *J. Cyber Secur. Technol.* **2020**, *4*, 1–66. [CrossRef]
46. bin Salleh, R.; Koubaa, A.; Khan, Z.; Khan, M.K.; Ali, I. Data plane failure and its recovery techniques in SDN: A systematic literature review. *J. King Saud Univ.-Comput. Inf. Sci.* **2023**, *35*, 176–201.
47. Dybå, T.; Dingsøyr, T. Empirical studies of agile software development: A systematic review. *Inf. Softw. Technol.* **2008**, *50*, 833–859. [CrossRef]
48. Tatang, D.; Quinkert, F.; Frank, J.; Röpke, C.; Holz, T. SDN-Guard: Protecting SDN controllers against SDN rootkits. In Proceedings of the 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Berlin, Germany, 6–8 November 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 297–302.
49. Dridi, L.; Zhani, M.F. SDN-Guard: DoS Attacks Mitigation in SDN Networks. In Proceedings of the 2016 5th IEEE International Conference on Cloud Networking (Cloudnet), Pisa, Italy, 3–5 October 2016; pp. 212–217.
50. Hussein, A.; Elhajj, I.H.; Chehab, A.; Kayssi, A. SDN Security Plane: An Architecture for Resilient Security Services. In Proceedings of the 2016 IEEE International Conference on Cloud Engineering Workshop (IC2EW), Berlin, Germany, 4–8 April 2016; pp. 54–59.
51. Chen, K.Y.; Junuthula, A.R.; Siddhrau, I.K.; Xu, Y.; Chao, H.J. SDNShield: Towards more comprehensive defense against DDoS attacks on SDN control plane. In Proceedings of the 2016 IEEE Conference on Communications and Network Security (CNS), Philadelphia, PA, USA, 17–19 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 28–36.
52. Wang, Y.; Hu, T.; Tang, G.; Xie, J.; Lu, J. SGS: Safe-guard scheme for protecting control plane against DDoS attacks in software-defined networking. *IEEE Access* **2019**, *7*, 34699–34710. [CrossRef]
53. Pascoal, T.A.; Fonseca, I.E.; Nigam, V. Slow denial-of-service attacks on software defined networks. *Comput. Netw.* **2020**, *173*, 107223. [CrossRef]
54. Deng, S.; Gao, X.; Lu, Z.; Li, Z.; Gao, X. Dos vulnerabilities and mitigation strategies in software-defined networks. *J. Netw. Comput. Appl.* **2019**, *125*, 209–219. [CrossRef]
55. Jantila, S.; Chaipah, K. A security analysis of a hybrid mechanism to defend DDoS attacks in SDN. *Procedia Comput. Sci.* **2016**, *86*, 437–440. [CrossRef]
56. Cui, Y.; Yan, L.; Li, S.; Xing, H.; Pan, W.; Zhu, J.; Zheng, X. SD-Anti-DDoS: Fast and efficient DDoS defense in software-defined networks. *J. Netw. Comput. Appl.* **2016**, *68*, 65–79. [CrossRef]
57. Sahay, R.; Blanc, G.; Zhang, Z.; Debar, H. ArOMA: An SDN based autonomic DDoS mitigation framework. *Comput. Secur.* **2017**, *70*, 482–499. [CrossRef]
58. Dao, N.N.; Kim, J.; Park, M.; Cho, S. Adaptive suspicious prevention for defending DoS attacks in SDN-based convergent networks. *PLoS ONE* **2016**, *11*, e0160375. [CrossRef]
59. Ma, D.; Xu, Z.; Lin, D. Defending blind DDoS attack on SDN based on moving target defense. In *International Conference on Security and Privacy in Communication Networks*; Springer: Cham, Switzerland, 2014; pp. 463–480.
60. Arivudainambi, D.; KA, V.K.; Chakkaravarthy, S.S. LION IDS: A meta-heuristics approach to detect DDoS attacks against Software-Defined Networks. *Neural Comput. Appl.* **2019**, *31*, 1491–1501. [CrossRef]

61. Myint Oo, M.; Kamolphiwong, S.; Kamolphiwong, T.; Vasupongayya, S. Advanced support vector machine-(ASVM-) based detection for distributed denial of service (DDoS) attack on software defined networking (sdn). *J. Comput. Netw. Commun.* **2019**, *2019*, 8012568. [CrossRef]
62. Han, B.; Yang, X.; Sun, Z.; Huang, J.; Su, J. OverWatch: A cross-plane DDoS attack defense framework with collaborative intelligence in SDN. *Secur. Commun. Netw.* **2018**, *2018*, 9649643. [CrossRef]
63. Manso, P.; Moura, J.; Serrão, C. SDN-based intrusion detection system for early detection and mitigation of DDoS attacks. *Information* **2019**, *10*, 106. [CrossRef]
64. Swami, R.; Dave, M.; Ranga, V. Voting-based intrusion detection framework for securing software-defined networks. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5927. [CrossRef]
65. Wu, X.; Liu, M.; Dou, W.; Yu, S. DDoS attacks on data plane of software-defined network: Are they possible? *Secur. Commun. Netw.* **2016**, *9*, 5444–5459. [CrossRef]
66. Banitalebi Dehkordi, A.; Soltanaghaei, M.; Boroujeni, F.Z. The DDoS attacks detection through machine learning and statistical methods in SDN. *J. Supercomput.* **2021**, *77*, 2383–2415. [CrossRef]
67. Varghese, J.E.; Muniyal, B. An efficient ids framework for ddos attacks in sdn environment. *IEEE Access* **2021**, *9*, 69680–69699. [CrossRef]
68. Aslam, M.; Ye, D.; Tariq, A.; Asad, M.; Hanif, M.; Ndzi, D.; Chelloug, S.A.; Elaziz, M.A.; Al-Qaness, M.A.; Jilani, S.F. Adaptive Machine Learning Based Distributed Denial-of-Services Attacks Detection and Mitigation System for SDN-Enabled IoT. *Sensors* **2022**, *22*, 2697. [CrossRef]
69. Wang, S.; Gomez, K.; Sithamparanathan, K.; Asghar, M.R.; Russello, G.; Zanna, P. Mitigating ddos attacks in sdn-based iot networks leveraging secure control and data plane algorithm. *Appl. Sci.* **2021**, *11*, 929. [CrossRef]
70. Tang, D.; Zhang, S.; Yan, Y.; Chen, J.; Qin, Z. Real-time Detection and Mitigation of LDoS Attacks in the SDN Using the HGB-FP Algorithm. *IEEE Trans. Serv. Comput.* **2021**, *15*, 3471–3484. [CrossRef]
71. Khamaiseh, S.; Al-Alaj, A.; Adnan, M.; Alomari, H.W. The Robustness of Detecting Known and Unknown DDoS Saturation Attacks in SDN via the Integration of Supervised and Semi-Supervised Classifiers. *Future Internet* **2022**, *14*, 164. [CrossRef]
72. Singh, J.; Behal, S. A novel approach for the detection of DDoS attacks in SDN using information theory metric. In Proceedings of the 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 17–19 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 512–516.
73. Jiang, S.; Yang, L.; Gao, X.; Zhou, Y.; Feng, T.; Song, Y.; Liu, K.; Cheng, G. BSD-Guard: A Collaborative Blockchain-Based Approach for Detection and Mitigation of SDN-Targeted DDoS Attacks. *Secur. Commun. Netw.* **2022**, *2022*, 1608689. [CrossRef]
74. Dalati, M.S.; Meng, W.; Chiu, W.Y. NGS: Mitigating DDoS Attacks using SDN-based Network Gate Shield. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6.
75. Ahmed, N.; Ngadi, A.b.; Sharif, J.M.; Hussain, S.; Uddin, M.; Rathore, M.S.; Iqbal, J.; Abdelhaq, M.; Alsaqour, R.; Ullah, S.S.; et al. Network Threat Detection Using Machine/Deep Learning in SDN-Based Platforms: A Comprehensive Analysis of State-of-the-Art Solutions, Discussion, Challenges, and Future Research Direction. *Sensors* **2022**, *22*, 7896. [CrossRef] [PubMed]
76. Amin, R.; Rojas, E.; Aqdus, A.; Ramzan, S.; Casillas-Perez, D.; Arco, J.M. A survey on machine learning techniques for routing optimization in SDN. *IEEE Access* **2021**, *9*, 104582–104611. [CrossRef]
77. Scott-Hayward, S.; Arumugam, T. OFMTL-SEC: State-based security for software defined networks. In Proceedings of the 2018 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Verona, Italy, 27–29 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–7.
78. Zhang, C.; Hu, G.; Chen, G.; Sangaiah, A.K.; Zhang, P.A.; Yan, X.; Jiang, W. Towards a SDN-based integrated architecture for mitigating IP spoofing attack. *IEEE Access* **2017**, *6*, 22764–22777. [CrossRef]
79. Mowla, N.I.; Doh, I.; Chae, K. An efficient defense mechanism for spoofed IP attack in SDN based CDNi. In Proceedings of the 2015 International Conference on Information Networking (ICOIN), Cambodia, 12–14 January 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 92–97.
80. Afek, Y.; Bremler-Barr, A.; Shafir, L. Network anti-spoofing with SDN data plane. In Proceedings of the IEEE INFOCOM 2017—IEEE Conference on Computer Communications, Atlanta, GA, USA, 1–4 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–9.
81. Tchendji, V.K.; Mvah, F.; Djamegni, C.T.; Yankam, Y.F. E2BaSeP: Efficient Bayes Based Security Protocol Against ARP Spoofing Attacks in SDN Architectures. *J. Hardw. Syst. Secur.* **2020**, *5*, 58–74. [CrossRef]
82. Lu, Y.; Wang, M.; Huang, P. An SDN-based authentication mechanism for securing neighbor discovery protocol in IPv6. *Secur. Commun. Netw.* **2017**, *2017*, 5838657. [CrossRef]
83. Shah, Z.; Cosgrove, S. Mitigating ARP Cache Poisoning Attack in Software-Defined Networking (SDN): A Survey. *Electronics* **2019**, *8*, 1095. [CrossRef]
84. Varadharajan, V.; Tupakula, U. Counteracting attacks from malicious end hosts in software defined networks. *IEEE Trans. Netw. Serv. Manag.* **2019**, *17*, 160–174. [CrossRef]
85. Li, Y.; Li, J. MultiClassifier: A combination of DPI and ML for application-layer classification in SDN. In Proceedings of the 2014 2nd International Conference on Systems and Informatics (ICSAI 2014), Shanghai, China, 15–17 November 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 682–686.

86. Li, H.; Hu, C.; Hong, J.; Chen, X.; Jiang, Y. Parsing application layer protocol with commodity hardware for SDN. In Proceedings of the 2015 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), Oakland, CA, USA, 7–8 May 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 51–61.
87. Xiaochen, Q.; Shihui, Z.; Lize, G.; Yongmei, C. A Fine-Grained Detection Mechanism for SDN Rule Collision. In *International Conference on Advanced Hybrid Information Processing*; Springer: Cham, Switzerland, 2018; pp. 549–559.
88. Isyaku, B.; Mohd Zahid, M.S.; Bte Kamat, M.; Abu Bakar, K.; Ghaleb, F.A. Software Defined Networking Flow Table Management of OpenFlow Switches Performance and Security Challenges: A Survey. *Future Internet* **2020**, *12*, 147. [CrossRef]
89. Abdou, A.R.; van Oorschot, P.C.; Wan, T. Comparative Analysis of Control Plane Security of SDN and Conventional Networks. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 3542–3559. [CrossRef]
90. Mahboob, T.; Arshad, I.; Batool, A.; Nawaz, M. Authentication Mechanism to Secure Communication between Wireless SDN Planes. In Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 8–12 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 582–588.
91. Wang, Y.; Yi, J.; Guo, J.; Qiao, Y.; Qi, M.; Chen, Q. A Semistructured Random Identifier Protocol for Anonymous Communication in SDN Network. *Secur. Commun. Netw.* **2018**, *2018*, 2916356. [CrossRef]
92. Lam, J.; Lee, S.G.; Lee, H.J.; Oktian, Y.E. Securing SDN southbound and data plane communication with IBC. *Mob. Inf. Syst.* **2016**, *2016*, 1708970. [CrossRef]
93. Yao, J.; Han, Z.; Sohail, M.; Wang, L. A robust security architecture for SDN-based 5G networks. *Future Internet* **2019**, *11*, 85. [CrossRef]
94. Benzekki, K.; El Fergougui, A.; El Belrhiti El Alaoui, A. Devolving IEEE 802.1 X authentication capability to data plane in software-defined networking (SDN) architecture. *Secur. Commun. Netw.* **2016**, *9*, 4369–4377. [CrossRef]
95. Kumar, P.; Tripathi, M.; Nehra, A.; Conti, M.; Lal, C. Safety: Early detection and mitigation of tcp syn flood utilizing entropy in sdn. *IEEE Trans. Netw. Serv. Manag.* **2018**, *15*, 1545–1559. [CrossRef]
96. Mohammadi, R.; Javidan, R.; Conti, M. Slicots: An sdn-based lightweight countermeasure for tcp syn flooding attacks. *IEEE Trans. Netw. Serv. Manag.* **2017**, *14*, 487–497. [CrossRef]
97. Wei, H.C.; Tung, Y.H.; Yu, C.M. Counteracting UDP flooding attacks in SDN. In Proceedings of the 2016 IEEE NetSoft Conference and Workshops (NetSoft), Seoul, Republic of Korea, 6–10 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 367–371.
98. Liu, X.; Cho, B.; Kim, J. Sd-ovs: Syn flooding attack defending open vswitch for sdn. In *International Workshop on Information Security Applications*; Springer: Cham, Switzerland, 2016; pp. 29–41.
99. Gao, D.; Liu, Z.; Liu, Y.; Foh, C.H.; Zhi, T.; Chao, H.C. Defending against Packet-In messages flooding attack under SDN context. *Soft Comput.* **2018**, *22*, 6797–6809. [CrossRef]
100. Wang, H.; Xu, L.; Gu, G. Floodguard: A dos attack prevention extension in software-defined networks. In Proceedings of the 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, Rio de Janeiro, Brazil, 22–25 June 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 239–250.
101. Ambrosin, M.; Conti, M.; De Gaspari, F.; Poovendran, R. Lineswitch: Tackling control plane saturation attacks in software-defined networking. *IEEE/ACM Trans. Netw.* **2016**, *25*, 1206–1219. [CrossRef]
102. Zhou, H.; Wu, C.; Yang, C.; Wang, P.; Yang, Q.; Lu, Z.; Cheng, Q. SDN-RDCD: A real-time and reliable method for detecting compromised SDN devices. *IEEE/ACM Trans. Netw.* **2018**, *26*, 2048–2061. [CrossRef]
103. Elsayed, M.S.; Le-Khac, N.A.; Jurcut, A.D. InSDN: A Novel SDN Intrusion Dataset. *IEEE Access* **2020**, *8*, 165263–165284. [CrossRef]
104. Fonseca, P.C.; Mota, E.S. A Survey on Fault Management in Software-Defined Networks. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2284–2321. [CrossRef]
105. Khan, S.; Gani, A.; Wahab, A.W.A.; Guizani, M.; Khan, M.K. Topology Discovery in Software Defined Networks: Threats, Taxonomy, and State-of-the-Art. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 303–324. [CrossRef]
106. Zhou, Y.; Zheng, K.; Ni, W.; Liu, R.P. Elastic switch migration for control plane load balancing in SDN. *IEEE Access* **2018**, *6*, 3909–3919. [CrossRef]
107. Zeyu, G.; Xingming, Z.; Qing, M. MDSA: Security Scheduling Mechanism for a Reliable SDN Control Layer Based on Mimic Defense. In *Recent Developments in Intelligent Computing, Communication and Devices*; Springer: Singapore, 2019; pp. 249–258.
108. Deng, S.; Gao, X.; Lu, Z.; Gao, X. Packet injection attack and its defense in software-defined networks. *IEEE Trans. Inf. Secur.* **2017**, *13*, 695–705. [CrossRef]
109. Gray, N.; Zinner, T.; Tran-Gia, P. Enhancing SDN security by device fingerprinting. In Proceedings of the 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), Lisbon, Portugal, 8–12 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 879–880.
110. Krishnan, P.; Duttagupta, S.; Achuthan, K. VARMAN: Multi-plane security framework for software defined networks. *Comput. Commun.* **2019**, *148*, 215–239. [CrossRef]
111. Sahay, R.; Meng, W.; Jensen, C.D. The application of Software Defined Networking on securing computer networks: A survey. *J. Netw. Comput. Appl.* **2019**, *131*, 89–108. [CrossRef]
112. Farhady, H.; Lee, H.; Nakao, A. Software-defined networking: A survey. *Comput. Netw.* **2015**, *81*, 79–95. [CrossRef]
113. Wang, X.; Gao, N.; Zhang, L.; Liu, Z.; Wang, L. Novel mitm attacks on security protocols in sdn: A feasibility study. In *International Conference on Information and Communications Security*; Springer: Cham, Switzerland, 2016; pp. 455–465.

114. Zhao, Z.; Gong, D.; Lu, B.; Liu, F.; Zhang, C. SDN-based Double Hopping Communication against sniffer attack. *Math. Probl. Eng.* **2016**, *2016*, 8927169. [CrossRef]
115. Zhao, Z.; Liu, F.; Gong, D. An SDN-based fingerprint hopping method to prevent fingerprinting attacks. *Secur. Commun. Netw.* **2017**, *2017*, 1560594. [CrossRef]
116. Chou, L.D.; Liu, C.C.; Lai, M.S.; Chiu, K.C.; Tu, H.H.; Su, S.; Lai, C.L.; Yen, C.K.; Tsai, W.H. Behavior anomaly detection in SDN control plane: A case study of topology discovery attacks. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 8898949. [CrossRef]
117. Yu, H.; Li, K.; Qi, H. An Active Controller Selection Scheme for Minimizing Packet-In Processing Latency in SDN. *Secur. Commun. Netw.* **2019**, *2019*, 1949343. [CrossRef]
118. Di Maio, A.; Palattella, M.R.; Soua, R.; Lamorte, L.; Vilajosana, X.; Alonso-Zarate, J.; Engel, T. Enabling SDN in VANETs: What is the impact on security? *Sensors* **2016**, *16*, 2077. [CrossRef]
119. Sung, Y.; Sharma, P.K.; Lopez, E.M.; Park, J.H. FS-OpenSecurity: A taxonomic modeling of security threats in SDN for future sustainable computing. *Sustainability* **2016**, *8*, 919. [CrossRef]
120. Yu, Z.; Zhu, H.; Xiao, R.; Song, C.; Dong, J.; Li, H. Detection and defense against network isolation attacks in software-defined networks. *Trans. Emerg. Telecommun. Technol.* **2020**, *32*, e3895. [CrossRef]
121. Xie, R.; Cao, J.; Li, Q.; Sun, K.; Gu, G.; Xu, M.; Yang, Y. Disrupting the SDN Control Channel via Shared Links: Attacks and Countermeasures. *IEEE/ACM Trans. Netw.* **2022**, *30*, 2158–2172. [CrossRef]
122. Calle, E.; Martínez, D.; Mycek, M.; Pióro, M. Resilient backup controller placement in distributed SDN under critical targeted attacks. *Int. J. Crit. Infrastruct. Prot.* **2021**, *33*, 100422. [CrossRef]
123. Ambrosin, M.; Conti, M.; De Gaspari, F.; Poovendran, R. Lineswitch: Efficiently managing switch flow in software-defined networking while effectively tackling dos attacks. In Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, Singapore, 14 April–17 March 2015; pp. 639–644.
124. Dover, J.M. *A Denial of Service Attack against the Open Floodlight SDN Controller*; Dover Networks LCC: Edgewater, MD, USA, 2013.
125. Shin, S.; Gu, G. Attacking software-defined networks: A first feasibility study. In Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking, Hong Kong, China, 16 August 2013; pp. 165–166.
126. Fonseca, P.; Bennesby, R.; Mota, E.; Passito, A. A replication component for resilient OpenFlow-based networking. In Proceedings of the 2012 IEEE Network Operations and Management Symposium, Maui, HI, USA, 16–20 April 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 933–939.
127. Yao, G.; Bi, J.; Guo, L. On the cascading failures of multi-controllers in software defined networks. In Proceedings of the 2013 21st IEEE International Conference on Network Protocols (ICNP), Goettingen, Germany, 7–10 October 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1–2.
128. Akhunzada, A.; Ahmed, E.; Gani, A.; Khan, M.K.; Imran, M.; Guizani, S. Securing software defined networks: Taxonomy, requirements, and open issues. *IEEE Commun. Mag.* **2015**, *53*, 36–44. [CrossRef]
129. Kandoi, R.; Antikainen, M. Denial-of-service attacks in OpenFlow SDN networks. In Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), Ottawa, ON, Canada, 11–15 May 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1322–1326.
130. David, E.S.; Taylor, D.; Turner, J. Packet classification using extended TCAMs. In Proceedings of the 11th IEEE International Conference on Network Protocols, Atlanta, GA, USA, 4–7 November 2003.
131. Zhang, P.; Wang, H.; Hu, C.; Lin, C. On denial of service attacks in software defined networks. *IEEE Netw.* **2016**, *30*, 28–33. [CrossRef]
132. Porras, P.; Shin, S.; Yegneswaran, V.; Fong, M.; Tyson, M.; Gu, G. *A Framework for Enabling Security Controls in OpenFlow Networks*; ACM: New York, NY, USA, 2012.
133. Klöti, R.; Kotronis, V.; Smith, P. OpenFlow: A security analysis. In Proceedings of the 2013 21st IEEE International Conference on Network Protocols (ICNP), Goettingen, Germany, 7–10 October 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1–6.
134. Antikainen, M.; Aura, T.; Särelä, M. Spook in your network: Attacking an sdn with a compromised openflow switch. In *Nordic Conference on Secure IT Systems*; Springer: Cham, Switzerland, 2014; pp. 229–244.
135. Wen, X.; Chen, Y.; Hu, C.; Shi, C.; Wang, Y. Towards a secure controller platform for openflow applications. In Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking, Hong Kong, China, 16 August 2013; pp. 171–172.
136. Jarschel, M.; Oechsner, S.; Schlosser, D.; Pries, R.; Goll, S.; Tran-Gia, P. Modeling and performance evaluation of an OpenFlow architecture. In Proceedings of the 2011 23rd International Teletraffic Congress (ITC), San Francisco, CA, USA, 6–9 September 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1–7.
137. Thomas, R.M.; James, D. DDOS detection and denial using third party application in SDN. In Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 3892–3897.
138. Nandhini, M. An Implementation of Public Key Infrastructure Using Wireless Communication Networks. *Int. J. Grid Distrib. Comput.* **2015**, *8*, 35–42.
139. Wang, M.M.; Liu, J.W.; Chen, J.; Mao, J.; Mao, K.F. Software defined networking: Security model, threats and mechanism. *J. Softw.* **2016**, *27*, 969–992.

140. Al-Shabibi, A.; De Leenheer, M.; Gerola, M.; Koshibe, A.; Parulkar, G.; Salvadori, E.; Snow, B. OpenVirteX: Make your virtual SDNs programmable. In Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, Chicago, IL, USA, 22 August 2014; pp. 25–30.
141. Hu, T.; Yi, P.; Zhang, J.; Lan, J. Reliable and load balance-aware multi-controller deployment in SDN. *China Commun.* **2018**, *15*, 184–198. [CrossRef]
142. Li, H.; Li, P.; Guo, S.; Nayak, A. Byzantine-resilient secure software-defined networks with multiple controllers in cloud. *IEEE Trans. Cloud Comput.* **2014**, *2*, 436–447. [CrossRef]
143. Aghaie, A.; Kermani, M.M.; Azarderakhsh, R. Fault diagnosis schemes for secure lightweight cryptographic block cipher RECTANGLE benchmarked on FPGA. In Proceedings of the 2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS), Monte Carlo, Monaco, 11–14 December 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 768–771.
144. Sun, N.; Cheng, J.; Liu, W.; Peng, Z.; Sun, C.; Wang, C.; Sha, H.; Wang, Y.; Yu, W. A novel on-chip linear and switching mixed regulation against power analysis attacks. *Integration* **2023**, *93*, 102049. [CrossRef]
145. Sarker, A.; Kermani, M.M.; Azarderakhsh, R. Fault detection architectures for inverted binary ring-LWE construction benchmarked on FPGA. *IEEE Trans. Circuits Syst. II: Express Briefs* **2020**, *68*, 1403–1407. [CrossRef]
146. He, P.; Bao, T.; Xie, J.; Amin, M. FPGA Implementation of Compact Hardware Accelerators for Ring-Binary-LWE based Post-Quantum Cryptography. *ACM Trans. Reconfigurable Technol. Syst.* **2022**, *16*, 1–23. [CrossRef]
147. Anastasova, M.; Azarderakhsh, R.; Kermani, M.M.; Beshaj, L. Time-Efficient Finite Field Microarchitecture Design for Curve448 and Ed448 on Cortex-M4. In *International Conference on Information Security and Cryptology*; Springer Nature: Cham, Switzerland, 2022; pp. 292–314.
148. Anastasova, M.; Azarderakhsh, R.; Kermani, M.M. Fast strategies for the implementation of SIKE round 3 on ARM Cortex-M4. *IEEE Trans. Circuits Syst. I: Regul. Pap.* **2021**, *68*, 4129–4141. [CrossRef]
149. Bisheh-Niasar, M.; Azarderakhsh, R.; Mozaffari-Kermani, M. Cryptographic accelerators for digital signature based on Ed25519. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2021**, *29*, 1297–1305. [CrossRef]
150. Mozaffari-Kermani, M.; Azarderakhsh, R.; Aghaie, A. Reliable and error detection architectures of Pomaranch for false-alarm-sensitive cryptographic applications. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2015**, *23*, 2804–2812. [CrossRef]
151. Mozaffari-Kermani, M.; Reyhani-Masoleh, A. Reliable hardware architectures for the third-round SHA-3 finalist Grostl benchmarked on FPGA platform. In Proceedings of the 2011 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems, Vancouver, BC, Canada, 3–5 October 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 325–331.
152. Aghaie, A.; Kermani, M.M.; Azarderakhsh, R. Fault diagnosis schemes for low-energy block cipher Midori benchmarked on FPGA. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2016**, *25*, 1528–1536. [CrossRef]
153. Sanal, P.; Karagoz, E.; Seo, H.; Azarderakhsh, R.; Mozaffari-Kermani, M. Kyber on ARM64: Compact implementations of Kyber on 64-bit ARM Cortex-A processors. In Proceedings of the Security and Privacy in Communication Networks: 17th EAI International Conference, SecureComm 2021, Virtual, 6–9 September 2021; Springer International Publishing: Cham, Switzerland, 2021; pp. 424–440.
154. Shin, S.W.; Porras, P.; Yegneswara, V.; Fong, M.; Gu, G.; Tyson, M. Fresco: Modular composable security services for software-defined networks. In Proceedings of the 20th Annual Network & Distributed System Security Symposium, San Diego, CA, USA, 27 February–3 March 2023.
155. Seeber, S.; Stiemert, L.; Rodosek, G.D. Towards an SDN-enabled IDS environment. In Proceedings of the 2015 IEEE Conference on Communications and Network Security (CNS), Florence, Italy, 28–30 September 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 751–752.
156. Nygren, A.; Pfaff, B.; Lantz, B.; Heller, B.; Barker, C.; Beckmann, C.; Cohn, D.; Malek, D.; Talayco, D.; Erickson, D. *Openflow Switch Specification*; Version 1.5. 1.; Technical Report; Open Networking Foundation: Palo Alto, CA, USA, 2015.
157. Akila, J.; Vetripriya, M.; Brigetta, A.; Magesh Kumar, K. Dynamic network security protection on cloud computing. *Int. Educ. Res. J. (IERJ)* **2016**, *2*.
158. Brooks, M.; Yang, B. A Man-in-the-Middle attack against OpenDayLight SDN controller. In Proceedings of the 4th Annual ACM Conference on Research in Information Technology, Chicago, IL, USA, 30 September–3 October 2015; pp. 45–49.
159. Scott-Hayward, S.; O'Callaghan, G.; Sezer, S. SDN security: A survey. In *2013 IEEE SDN for Future Networks and Services (SDN4FNS)*; IEEE: Piscataway, NJ, USA, 2013.
160. Switch, B. Developing Floodlight Modules. Floodlight OpenFlow Controller. 2012. Available online: https://scholar.google.com.hk/scholar?hl=zh-CN&as_sdt=0%2C5&q=Switch%2C+B.+Developing+floodlight+modules.+Floodlight+OpenFlow+controller%2C%E2%80%9D+2012.&btnG=#d=gs_cit&t=1689313192518&u=%2Fscholar%3Fq%3Dinfo%3AnBUnnVPlp5YJ%3Ascholar.google.com%2F%26output%3Dcite%26scirp%3D0%26hl%3Dzh-CN (accessed on 13 July 2023).
161. Voellmy, A.; Wang, J. Scalable software defined network controllers. In Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Helsinki, Finland, 13–17 August 2012; pp. 289–290.
162. Cai, Z.; Cox, A.L.; Maestro, T.E.N. *Maestro: A System for Scalable OpenFlow Control*; Technical Report TR10-08; Rice University: Houston, TX, USA, 2010.
163. Phemius, K.; Bouet, M.; Leguay, J. Disco: Distributed multi-domain sdn controllers. In Proceedings of the 2014 IEEE Network Operations and Management Symposium (NOMS), Krakow, Poland, 5–9 May 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 1–4.

164. Tootoonchian, A.; Ganjali, Y. Hyperflow: A distributed control plane for openflow. In Proceedings of the 2010 Internet Network Management Conference on Research on Enterprise Networking, San Jose, CA, USA, 27 April 2010; Volume 3.
165. Braga, R.; Mota, E.; Passito, A. Lightweight DDoS flooding attack detection using NOX/OpenFlow. In Proceedings of the IEEE Local Computer Network Conference, Denver, CO, USA, 10–14 October 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 408–415.
166. Kohonen, T. *The Self-Organizing Map*; IEEE: Piscataway, NJ, USA, 1990; Volume 78, pp. 1464–1480.
167. Porras, P.; Shin, S.; Yegneswaran, V.; Fong, M.; Tyson, M.; Gu, G. A security enforcement kernel for OpenFlow networks. In Proceedings of the First Workshop on Hot Topics in Software Defined Networks, Helsinki, Finland, 13 August 2012; pp. 121–126.
168. Khurshid, A.; Zou, X.; Zhou, W.; Caesar, M.; Godfrey, P.B. Veriflow: Verifying network-wide invariants in real time. In Proceedings of the First Workshop on Hot Topics in Software Defined Networks, Helsinki, Finland, 13 August 2012; pp. 15–27.
169. Zhang, Y.; Beheshti, N.; Tatipamula, M. On resilience of split-architecture networks. In Proceedings of the 2011 IEEE Global Telecommunications Conference—GLOBECOM 2011, Houston, TX, USA, 5–9 December 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1–6.
170. Xu, Y.; Liu, Y. DDoS attack detection under SDN context. In Proceedings of the IEEE INFOCOM 2016—The 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–14 April 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–9.
171. Liu, Z.; Campbell, R.H.; Mickunas, M.D. Active security support for active networks. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2023**, *33*, 432–445.
172. Hartman, S.; Wasserman, M.; Zhang, D. Security Requirements in the Software Defined Networking Model. Internet Engineering Task Force, Internet-Draft draft-hartman-sdnsec-requirements-01. 2013. Available online: https://datatracker.ietf.org/doc/html/draft-hartman-sdnsec-requirements-01 (accessed on 9 July 2023).
173. Naous, J.; Erickson, D.; Covington, G.A.; Appenzeller, G.; McKeown, N. Implementing an OpenFlow switch on the NetFPGA platform. In Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, San Jose, CA, USA, 6–7 November 2008; pp. 1–9.
174. Dhawan, M.; Poddar, R.; Mahajan, K.; Mann, V. SPHINX: Detecting Security Attacks in Software-Defined Networks. *Ndss* **2015**, *15*, 8–11.
175. Hong, G.C.; Lee, C.N.; Lee, M.F. Dynamic Threshold for DDoS Mitigation in SDN Environment. In Proceedings of the 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Lanzhou, China, 18–21 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–7.
176. Deepa, V.; Sudar, K.M.; Deepalakshmi, P. Design of Ensemble Learning Methods for DDoS Detection in SDN Environment. In Proceedings of the 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, India, 30–31 March 2019.
177. Jiang, Y.; Zhang, X.; Zhou, Q.; Cheng, Z. An entropy-based DDoS Defense mechanism in software defined networks. In *International Conference on Communicatins and Networking in China*; Springer: Cham, Switzerland, 2016; pp. 169–178.
178. Giotis, K.; Argyropoulos, C.; Androulidakis, G.; Kalogeras, D.; Maglaris, V. Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments. *Comput. Netw.* **2014**, *62*, 122–136. [CrossRef]
179. Mousavi, S.M.; St-Hilaire, M. Early detection of DDoS attacks against SDN controllers. In Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC), Garden Grove, CA, USA, 16–19 February 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 77–81.
180. Boite, J.; Nardin, P.A.; Rebecchi, F.; Bouet, M.; Conan, V. Statesec: Stateful monitoring for DDoS protection in software defined networks. In Proceedings of the 2017 IEEE Conference on Network Softwarization (NetSoft), Bologna, Italy, 3–7 July 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–9.
181. Floyd, S.; Paxson, V. Difficulties in simulating the Internet. *IEEE/ACm Trans. Netw.* **2001**, *9*, 392–403. [CrossRef]
182. Conti, M.; Lal, C.; Mohammadi, R.; Rawat, U. Lightweight solutions to counter DDoS attacks in software defined networking. *Wirel. Netw.* **2019**, *25*, 2751–2768. [CrossRef]
183. Piedrahita, A.F.M.; Rueda, S.; Mattos, D.M.; Duarte, O.C.M. FlowFence: A denial of service defense system for software defined networking. In Proceedings of the 2015 Global Information Infrastructure and Networking Symposium (GIIS), Guadalajara, Mexico, 28–30 October 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.
184. Li, C.; Wu, Y.; Yuan, X.; Sun, Z.; Wang, W.; Li, X.; Gong, L. Detection and defense of DDoS attack–based on deep learning in OpenFlow-based SDN. *Int. J. Commun. Syst.* **2018**, *31*, e3497. [CrossRef]
185. Dotcenko, S.; Vladyko, A.; Letenko, I. A fuzzy logic-based information security management for software-defined networks. In Proceedings of the 16th International Conference on Advanced Communication Technology, Pyeongchang, Korea, 16–19 February 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 167–171.