REVIEW OF SECURITY ATTACKS ON SOFTWARE DEFINED NETWORKING

A. Suleman[1], A. Mustafa[2], H. U. R. Kayani[3], M. A. Raza[4] and A. Saleem[5]

[12345] Department of Computer Science & IT - The University of Lahore, Sargodha Campus
*Corresponding Author Email: msits07223001@gmail.com

ABSTRACT: Now a days use of software defined networking increases in industry and in different enterprises due to its capabilities like centralized architecture, Data plan and control plan separation, different available controllers in different languages, very helpful in monitoring the network flow and other type of working behavior and security measures can be taken in SDN. In this article we will provide a brief overview of SDN and then we try to elaborate each and every thing to do our best like related existing work, architecture of SDN its security threats and also try to describe the SDN security attacks defense mechanism existing work with references that will be very helpful for readers to understand an SDN attacks and solutions. In short SDN becoming popular in future and also being used for many security measures to solve the security issues because SDN is also a technique that can be used as a part of security solution that is a very helpful in future. We also give a future direction at the end that is a really a novel research problem and must be solved to secure the SDN network.

Keywords: Software defined networking, control plan, data plan, SDN security, SDN attacks, SDN layers.

INTRODUCTION

Software defined networking (SDN) has recently gained huge fame as a way to address the lack of programmability in networking topologies and speed up network evolution. SDN's separation of the data plane (where the network is managed) and control plane (which deals with routing decisions, etc.) makes software development much easier. This allows for complex networking applications to be implemented on cost-effective hardware that can then be managed by programs utilizing standardized interfaces—allowing us to take advantage of all kinds of cool network configurations! The network itself should remain flexible enough to incorporate new features through the use of network applications. [1] The concept of running apps on mobile OS, such as iOS and android, is a widely recognized example of this phenomenon. With the exponential growth in both app-related internet traffic (bandwidth) and the number of IoT devices, the demand for resources from service providers, such as ISPs, to support these applications in homes and businesses is also increasing. As of 2019, there were 26.66 billion active internet-connected devices worldwide, and it is projected that global network traffic flow will reach 77.5 exabytes per month by 2022.. [2]. A sizable percentage of the data produced by Internet of Things (IoT) devices is managed by data centers, which provide consumers a wide range of cloud services. In traditional networking, maintaining internet infrastructure and data centers has grown more crucial due to the increased need for new applications like real-time processing. The creation of software-defined networking (SDN), a revolutionary networking design that divides the network control (management) plane from the network data plane, has solved this issue. The controller, also known as the "SDN controller," is in charge of determining the forwarding rules that should be applied to network devices in order to regulate user-initiated data flow. [3]. With the proliferation of dynamic applications, services, physical objects, and devices communicating worldwide over the Internet, there is a growing need for advanced network traffic control and orchestration systems. These systems need to efficiently manage the ever-changing variations in connection usage, bandwidth distribution, delay, power consumption, and variability across various and varied networks. [3] As multi-tenant data centers (DC) and the Internet of Things (IoT) continue to grow, network intricacy and traffic increase as well. Regrettably, implementing precise, Quality of Service (QoS)-conscious traffic management throughout the network proves challenging because of the design of contemporary networks. SDN is presently being developed and deployed in a range of devices, and it is getting closer to being a reality. Cloud computing and virtualization technologies are two areas where the academically well-researched combination of network programmability with distinct control and data plane capabilities has found practical application. [4]

Both the business sector and the academic community have rapidly displayed significant interest in Software Defined Networking (SDN). Off-the-shelf hardware and a no-cost, open-source Network Operating System (NOS) enable the replacement of costly, proprietary hardware and firmware. An open, impartial control-data plane connection, such as OpenFlow, also allows the unfettered development of network equipment

and software. Through the administration of network assets and the provision of high-level abstractions and APIs for interaction, administration, monitoring, and the programming of network switches, the NOS presents an open foundation that simplifies the creation of state-of-the-art and advantageous network applications and services that function across various hardware platforms. [5] Fig.1 explain an SDN in a very simple manner.

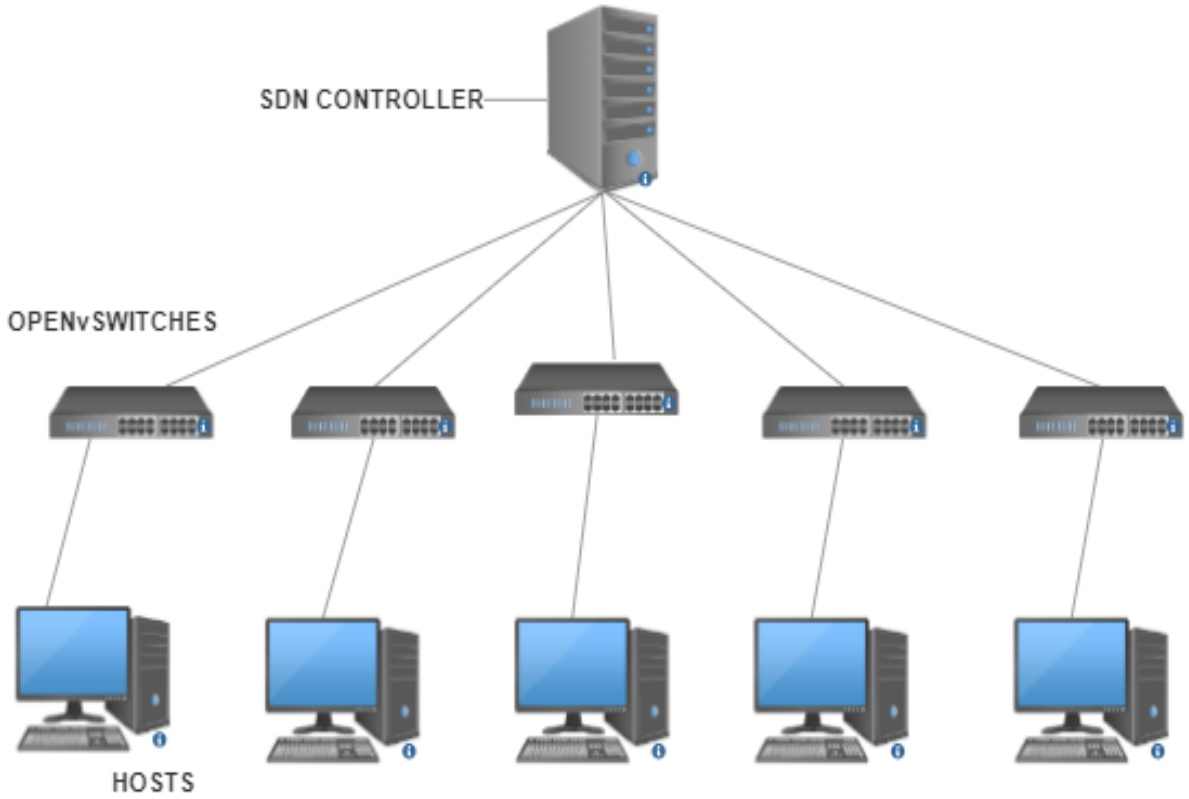

Figure 1: SDN NETWORK

The domain of networking technology that has garnered the greatest attention in recent years is Software Defined Networking (SDN). It offers a plethora of additional capabilities and enables the resolution of numerous complex problems in legacy networks. The SDN concept asserts that network intelligence should be transferred from packet-switching devices to a centrally situated controller in a logical manner. The monitored switches only execute the decisions made in the controller, which is where the forwarding decisions are decided. [6] We gain a variety of benefits from this, such as the capacity to manage and monitor the entire network at once, which is helpful for automating network operations and enhancing server and network utilization. recognizing all the benefits Microsoft and Google have switched their data centers over to SDN.

SDN architecture: SDN cannot function without application programming interfaces (APIs), which enable communication across the management, control, and data planes. Southbound APIs (SBI), Northbound APIs (NBI), and East/Westbound APIs are the three well-known APIs in the context of Distributed Controllers. These APIs, which are SDN architectural elements, are used to configure network applications or forwarding devices. The layered architecture of SDN, which includes APIs, is shown in Figure..[7].Fig.2 explains the SDN architecture in detail.

North bound API's: The application plane, which offers a collection of programs (applications/abstractions) necessary to meet the needs of the system itself, enables the controller to create or respond to demands of the SDN environment. The controller creator or other parties may offer one of the various applications that are now available, including firewalls, routing rules, protocols, etc. Each controller provides its own programming languages and specialized API since the northbound interface, which is used to enable communication between the application plane and the controller, is not standard. [8] While REST currently stands as the most widely employed API for industry applications, the ONF remains proactive and does not rule out the potential for standardization.

South bound API's: Through the southbound interface, the control plane and data plane communicate with each

other utilizing various protocols, including OpenFlow, OVSDB, OpFlex, NETCONF, and ForCes. The approach, referred to as OpenFlow, is currently under examination and is considered a standard. This article will focus on this method. The OF Switch and the OF Controller can optionally establish secure communication links employing TLS or plain TCP for the primary connections and TLS, DTLS, TCP, or UDP for the secondary connections. Security features are not inherently embedded in OpenFlow. Despite these challenges, the ONF recommends the use of TLS starting from the version.

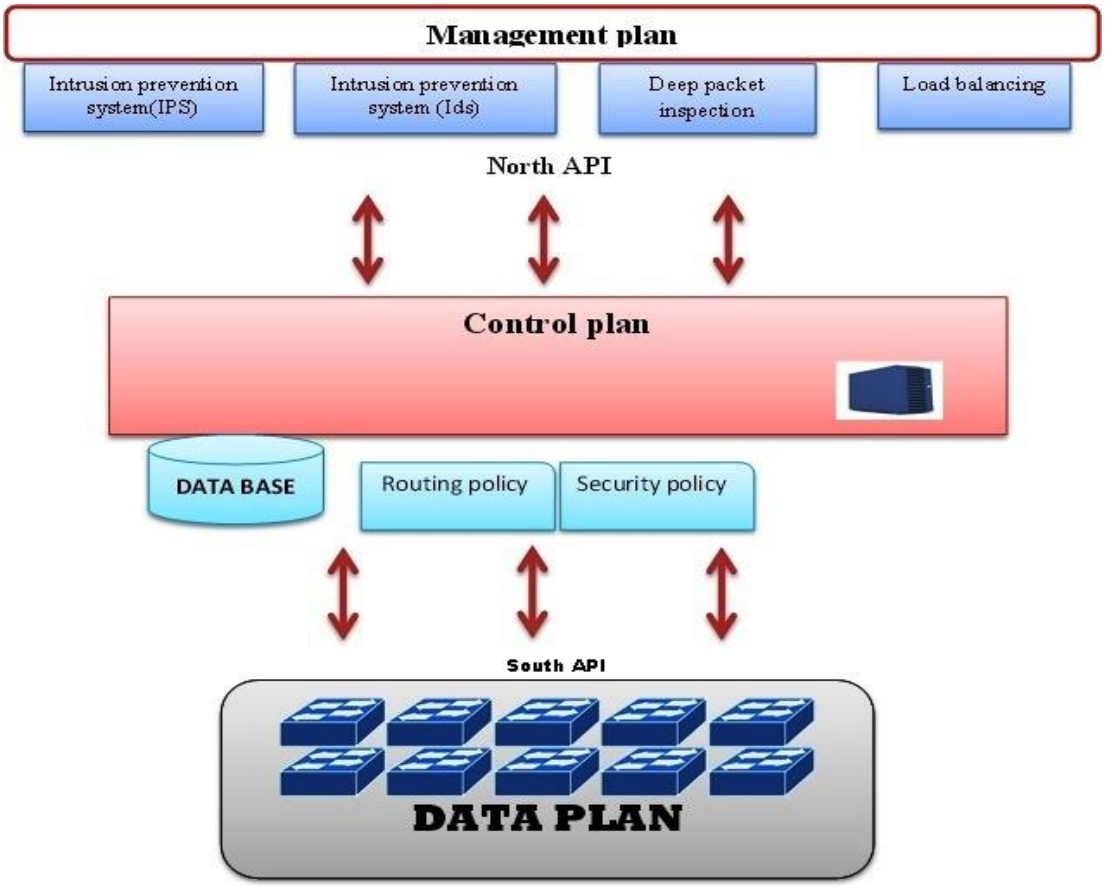

Figure 2: SDN Architecture

SDN layers: The control layer and data layer must be maintained distinct, according to the separation principle. SDN is a contemporary approach to network programmability that enables network activity to be governed and controlled via programming [10]. The SDN architecture allows centralised data route control regardless of the method used to connect this network equipment, which may be purchased from a variety of suppliers. The central control device, which also creates all the data, retains a comprehensive network perspective of the data path components and the links joining them. The core idea of SDN, a relatively new technology, is to remove intelligence from networking hardware and utilise a centralised controller to manage the operation of the whole network Fig. 2 illustrates.

a) ***Application layer:*** While the application plane handles the services, the data and control planes receive requests from applications for network operations. In common network topologies, this layer is where devices are monitored and controlled. Even though the devices' tasks are similar to those of SDN networks, the delivery methods are typically virtualized, centralised, and abstracted. Network information regarding the topology of the device and the appliances is needed for a broad variety of efficient end-to-end SDN-enabled services, and this information is defined at this layer together with the characteristics, services, and rules. Through this layer, programs may also quickly decide how to react to changes in the network.[9] . Overall, security enforcement is part of the application layer's remit in addition to load balancing, traffic engineering, and access restrictions. The management plane is the initial group of network programs that manages the control logic of a software-defined network. SDN-enabled networks

leverage programmability rather than a command line interface to make implementing new technologies more flexible and convenient.[10]. Applications and services from service providers include load balancing, routing, and custom applications for policy enforcement. It also makes use of pre-existing APIs to provide network orchestration and automation.

b) ***Control plan layer:*** In contrast to the typical network design, which incorporates SDN, a decoupled architecture might include a discrete control plane that specifies traffic routing and network management/control. The command aircraft, a layer, is responsible for managing, controlling, and customizing flow forwarding decisions and other forwarding activities throughout the network stack. SDN promotes agility in the automation, monitoring, administration, maintenance, expansion, deployment, and troubleshooting of network infrastructure by separating control from the data plane. Based on centralised control, the controller transmits the necessary and suitable data to the forwarding devices (such an OpenFlow switch) over the control plane as flow rules for efficient decision-making. The explicit forwarding information base (FIB) and MAC programming are used to achieve this[11]. It is made up of a centralised control plane where the controller specifies the business logic for managing and accessing various sorts of network data, including status information, topology information, statistics information, etc. The logic of the control plane is implemented centrally by controllers. In this controller, a single server oversees all control plane operations. Such Controllers offer the administrative and simplicity benefits since they offer a single point of control. However, due to each server's limited ability to manage data plane devices, they experience scaling challenges.[12]

SDN Controllers: A physically centralised control plane with a single controller overseeing the whole network is the optimal approach in terms of simplicity, according to theory. However, a single controller system might not be able to keep up with the network's growth. It is likely to get overloaded (controller bottleneck) as it struggles to maintain the same performance promises while managing an increase in the number of requests. Undoubtedly, a vast real-world network system cannot be served by a centralised SDN controller.[13]. Data centers and service provider networks, for example, have a variety of challenges that require different controller designs. Such large-scale networks often have a variety of scalability and reliability concerns. In order to appropriately determine how well they scale given their unique qualities, we highly suggest readers to thoroughly examine the pertinent studies.[14]. By using varying numbers of threads, it can be seen that single threaded controllers, like Ethane and NOX, have very low throughput since they are unable to manage several flows. However, multi-threaded controllers, like Beacon, Maestro, McNettle, and NOX-MT, can manage a lot of flows per second. All control plane logic is implemented by centralised controllers in a single place. One server manages all control plane operations in such a controller. Since they offer a single point of control, the key advantages of such a controller are simplicity and manageability. However, they have scalability issues since each server can only handle a certain number of data plane devices.[12].Table.1 describe the different controllers in tabular form.

Table.1 SDN Controllers with smart review.

SDN Controller	Implementation	Developer	Open Source	Overview
Beacon[15]	JAVA	stantford	Yes	a modular, cross-platform OpenFlow controller for Java that works with threaded and event-based processes.
NOX[16]	Python/C++	Nicira	Yes	the first Python and C++ OpenFlow controller.
POX[17]	Python	Nicira	Yes	SDN controller written in Python
MUL[18]	C	Kulcloud	Yes	Using a C-based multi-threaded framework, the OpenFlow controller.
RouteFlow[19]	C++	CPqD	Yes	Specific purpose
Flowvisor[20]	C	Stanford	Yes	Specific purpose
ovs-controller[21]	C	Independent Developers	Yes	Very simple SDN open Flow controller
Node Flow[22]	Java script	Independent Developers	Yes	Written in java script
RYU[23]	python	OSRG,NTT	Yes	an SDN operating system with APIs for building new network management and control applications and logically centralised control.
SNAC[24]	C++	Nicira	No	a user-friendly policy manager that is web-

				based to administer the network, set up devices, and keep an eye on events.
Floodlight[25]	Java	Big switch	Yes	a Beacon-based Java-based OpenFlow controller that operates with both physical and virtual OpenFlow switches and supports version 1.3.
Helios[26]	C	NEC	No	OpenFlow controller with a C programming interface that offers a shell for integrated experimentation.
Trema[27]	Ruby/C	NEC	Yes	Ruby-based OpenFlow controller development framework
Maestro[28]	Java	Rice University	Yes	a Java-based network operating system that offers interfaces
DISCO[29]	Java	Kevin Phemius	No	Modern overlay networks and wide area networks have a scattered and heterogeneous character that requires an open and adaptable distributed SDN control plane.
Fleet[30]	Java	Stephanos Matsumoto	No	Our argument is that the Fleet controller's lower layer, put on top of switches, solves many of the issues associated with utilising multiple controllers in SDN
Rosemary[31]	C	Seungwon Shin	Yes	Launching apps individually inside a micro-NOS is the foundation of the controller.
ONOS[32]	Java	ONF	Yes	ONOS was created to satisfy the requirements of operators looking to provide carrier-grade

1.5 Data plan layer: It's also known as a forwarding aircraft. Traffic is directed to the next jump based on the chosen destination network, following the control plane's reasoning. The router examines the packets in the data plane. The routers and switches employ the control plane for the entry and exit of frames and packets. It is possible to attack this layer using the following attack types. consists of hardware that is connected to networks, either wired or wireless, and a network administrator has planned the functioning of each mechanism. In network foundations, linked forwarding devices request direct data interchange on the data plane. By using certain direction sets, forwarding devices may perform activities. These devices take a logical approach while sending flows or packets. [33] Southbound connections provide a description of certain particular kinds of guidelines. The southbound connection is utilized to create a connection between the control and data planes. Switches, composing the data plane, are mainly tasked with guiding incoming streams through the routes indicated in flow tables to reach their ultimate destinations. [34]

SDN Security Background: The rigidity of earlier networks served as networking's primary source of inspiration. The requirement for administrators to manually configure each crucial component (router/switch) slows down change-making significantly. The plethora of network device providers makes it more difficult to choose the right specialists and scale up the infrastructure as needed. A prime illustration is the transition from IPv4 to IPv6. The transition is still underway despite its long duration. Software Defined Networking (SDN) has fundamentally transformed the way computers and humans engage in communication. Currently, it is a crucial part of software defined architecture, enabling companies to build highly adaptable IT systems.[35]. Recent progress in software-defined networks (SDN) has unveiled fresh opportunities for enhancing network operations. This method offers network operators a straightforward conceptual framework by eliminating the intricacies of network architecture from the equation. It becomes feasible to construct and code networks with increased adaptability when hardware and software can be decoupled [36]. Employing the concept of dynamic and responsive network administration, the intricacy of switches within the network is further reduced. Thanks to a novel network framework known as SDN, network apparatus can now be programmed from a central manager. The SDN promotes the separation of the data plane and the control plane, thus streamlining and enhancing the overall network structure's adaptability. Users have the liberty to integrate the data and control planes in whichever manner they prefer, thanks to the backing of the SDN architecture by the Open Networking Foundation (ONF) [37]. Now that these two planes have been divided, SDN applications may utilise the SDN northbound interface to program and control the underlying network architecture.

Future Sustainability Computing (FSC) holds a range of commitments in information technology, such as

enhanced automation, streamlined design, increased flexibility, policy-driven governance, and the connection of network management to more accessible IT workflow systems, all of which SDN needs to fulfill. SDN must enhance collaboration and fusion with teams dedicated to networks, servers, and security to address sustainability issues. These challenges will impact how enterprises formulate, create, implement, and oversee their networks.

a) ***SDN as security solution provider:*** Products like SANE make it possible for the link layer to implement simple and natural access control policies that are independent of topologies while keeping topology and service information hidden from those who don't need to know. Three stages separate a server B and a client A in SANE's paradigm. Prior to engaging, each party must establish a connection with the controller. Second, B informs the controller of the kind of service it is willing to provide, and A asks what kind of service the controller needs. Highlights from the literature that was examined and determined to be relevant to the current investigation are included in the paragraphs that follow. Researchers have developed and implemented DDoS detection methods based on artificial intelligence, particularly machine learning, to address the problem of DDoS attacks in SDN. [38] Security can ensure data authenticity, confidentiality, and integrity. One of the driving motivations behind SDN is security, which has increased significantly as a result of sophisticated network attacks. An important difficulty for SDN, a young technology, is security. The design of SDN introduces extra security and data protection vulnerabilities due to its programmability and decoupling aspects. Although SDN provides benefits, it also increases the risk of network assaults. [39] This is due to the fact that security issues have not been sufficiently studied and that security was not considered in the original architecture. The SDN architecture based on OpenFlow is thought to be the main target of potential attackers.

Use of machine Learning for anomaly detection: To acquire a single, complete picture of the network, SDN removes the control flow management capabilities from the forwarding elements (FEs) and consolidates them into a logically centralised controller. Switches, routers, gateways, and access points are all FEs that SDN controllers manage using standardized network device programming. The OpenFlow (OF) protocol, a well-known communication standard that enables the controller to give flow level commands, is used in the SDN communication architecture. [40] ML algorithms have become a well-known method of problem-solving in many fields. Although many machine learning applications seem promising, deep learning may also offer fascinating insights from a different angle. However, because deep learning methods need a lot of training data, Parampottupadam and Moldovann argue that deep learning models might not necessarily perform better than other regular ML models in some cases. [41] The recent rapid expansion of intelligent gadgets (such as smart phones, smart cars, and smart home devices) and network technologies (such as cloud computing and network virtualization) has led to an increase in data traffic in our society. To manage a huge number of devices and maximise traffic distribution, networks are becoming increasingly diversified and smart. [42] A typical production network serves a variety of applications, makes use of several protocols, and is made up of several pieces of hardware. Diverse cell types with varying transmission coverage, power levels, and operational mechanisms (including macro-cells, pico-cells, femto-cells, Relays, and RRHs) have been deployed within wireless networks, along with a range of communication technologies such as ZigBee, WiMAX, IEEE 802.11 ac/ad, Bluetooth, and LTE. The programmable attributes of Software Defined Networking (SDN) technology have enabled effective detection and monitoring of security issues within the network. Machine learning (ML) methodologies have recently been integrated into SDN-based Network Intrusion Detection Systems (NIDS) to safeguard computer networks and address security challenges. [43] More particular, we assessed how deep learning methods were used to create SDN-based NIDS. Software defined networks (SDNs) were the main focus of Ashraf et al.'s study on distributed denial of service (DDoS) assaults and intrusions. [44] provided machine learning algorithms. The study examined the usage of SDN anomaly detection using support vector machines, Bayesian networks, fuzzy logic, evolutionary algorithms, and neural networks. The essay goes into great length on the advantages and disadvantages of various anomaly detection techniques. In their in-depth analysis of the application of SDNs to protect networks, et al. [45] For security as a service, the usage of SDNs is recommended. The paper addresses a range of issues and suggestions that have been addressed in the literature in order to take network threats into account. Furthermore, Astuto et al. [46] Describe programmable networks in general, concentrating on SDNs. The research of programmable network development in the article emphasises the SDN architecture. SDN technologies and potential OpenFlow standard alternatives are being tested as part of the project. Hu and companions. [47] In a study of SDN from an OpenFlow perspective, the core idea, applications, and security features of OpenFlow are explained. Abdu with his friends. [48] SSH brute force attack study using automation. The LongTail project's data was used to undertake in-depth analyses of the behaviour of attackers and the dynamics of attacks, including password dictionary sharing and coordinated attempts. [49] The study's findings can be used to instruct SSH users and

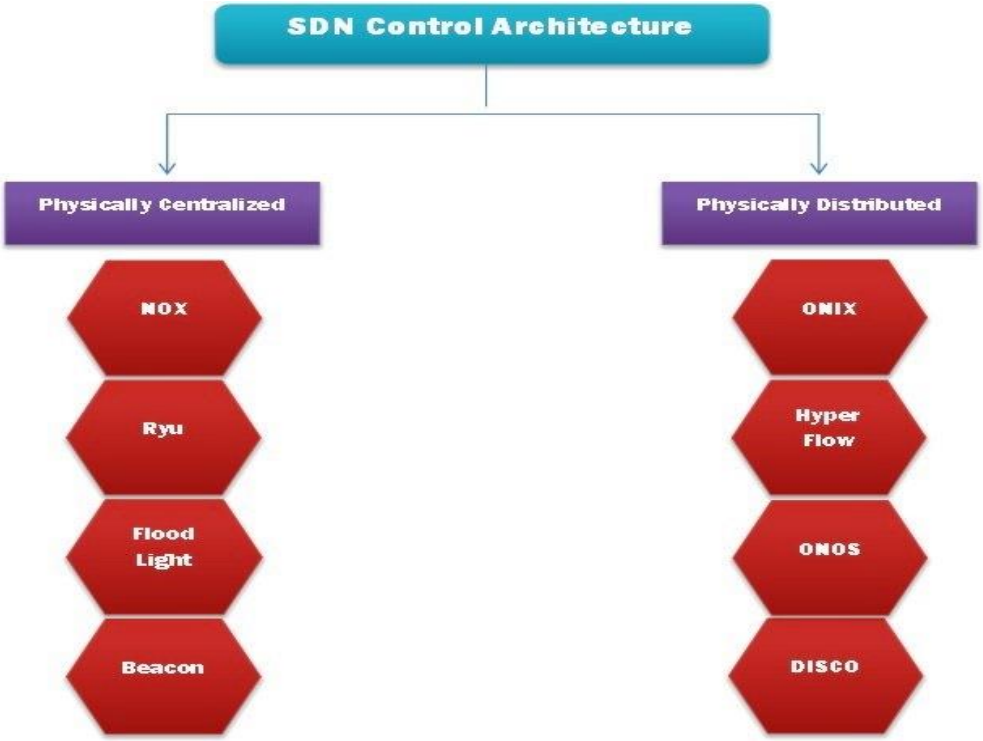

Figure 3 Physical classification of SDN control plane

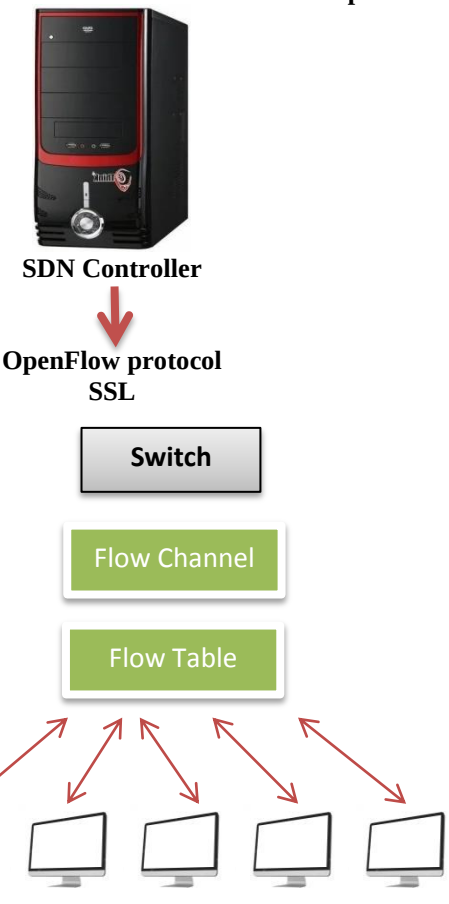

Figure 4: SDN Flow Protocol

network administrators. The following is highlighted by Sommer in his discussion on SDN anomaly detection techniques: k-Nearest Neighbours (kNN), Bayesian Networks, Support Vector Machines, and Expectation Maximisation. [50] There is a description of the various attack scenarios and prospective SDN programme implementations. Atlas is a cutting-edge framework that employs application-awareness in SDN, according to Qazi et al. [51]

b) **OpenFlow:** OpenFlow, according to the Open Networking Foundation (ONF), is the most widely used southbound interface. The OpenFlow protocol describes how to interact with one or more control servers and appropriate switches. An OpenFlow controller advises switches on how to route traffic by sending them entries in the flow table. So, for OpenFlow switches, controller setup is crucial. The OpenFlow protocol offers the ability to circumvent the limitations imposed by flow tables by allowing each flow entry to have an idle delay. These limitations prevent flow tables from overflowing by limiting the storage of flow tables to only the most recent entries and eliminating old idle timeout flow entries to create way for new ones. [3] The forwarding element builds a Packet In and delivers it to the controller whenever an unknown flow without a corresponding flow entry appears, asking for the rapid insertion of flow rules. Network security has become a nightmare as Bring-Your-Own-Device (BYOD) program become more and more common in educational institutions. By using their devices overnight or on the weekends, students and professors have the opportunity to bring malware into the network, necessitating hours of manual effort from IT specialists to manually discover and remove network risks. Even though we make every effort to safeguard consumers and maintain the network clean. [52] The foundation of OpenFlow is an Ethernet switch with an internal flow-table and a designated interface for adding and deleting flow entries. Switch devices from networking vendors are needed on college campuses' backbones and wire closets to enable OpenFlow. OpenFlow, in our opinion, provides a realistic alternative by freeing switch makers from having to provide details about their internal workings. Additionally, it enables researchers to do reliable tests at line speed and high port density on heterogeneous switches. [53] Researchers may investigate fresh ideas and test out novel applications by leveraging the special features of OpenFlow-based architectures. The OpenFlow-based apps might make managing and configuring networks easier, improve security, enable network and data center virtualization, and make it easier to deploy mobile devices. On top of networking operating systems like Nox, Beacon, Maestro, Floodlight, Trema, or Node, these apps operate. Larger-scale OpenFlow infrastructures have been built to allow the research community to carry out experiments and test their applications under more accurate circumstances. [54] The first OpenFlow standard required the use of TLS encryption in order to ensure the security of the control link between controllers and switches. The standards now make TLS encryption optional (up to v1.3.0). This is because a successful TLS installation requires many steps that provide operators more technical challenges than plaintext communication does. These tasks need the production and signing of a site-wide certificate, as well as certificates for the switches and the controller, using the site-wide private key. [55] Hardware and software manufacturers can implement the OpenFlow protocol, which enables network virtualization and can work alongside conventional routing technologies. [55, 56] A vast variety of modern dynamic applications, services, physical objects, and devices communicate globally through the Internet. Contemporary network technology referred to as "network traffic management and coordination" faces significant challenges in adapting to fluctuations in link utilization, bandwidth allocation, latency, energy consumption, and jitter within a diverse network environment.

The expansion of multi-user data centers (DC) and the burgeoning Internet of Things (IoT) elevate the complexity and volume of network traffic. Regrettably, the design of traditional networks hinders the implementation of detailed, Quality of Service (QoS)-aware traffic control across the entire network. [3] Managing network equipment individually becomes costly when dealing with temporal variations and multiple tenants in the data center.

c) **SDN applications:** In 2006, the Stanford University Clean Slate project and the American GENI project joined forces to produce Software-Defined Networking (SDN). This project attempted to reinvent the Internet and upgrade network architecture. Modern network architecture and network design methodologies are presented by the cutting-edge idea of SDN, which is backed by IT technologies. It promotes application-centric design, is built on software, and includes network softening capabilities. [57] Among the five primary SDN function categories, various applications have been developed, including Unified Network applications, Traffic Management applications, Data Center connectivity applications, Mobility and Wireless applications, and Network Security features. SDN adoption is presently progressing slowly, but it nevertheless provides network operators with a number of advantages in terms of flexibility and programmability. [58] Monitoring statistics show that well-known service providers like Google have switched their globally dispersed networks to an SDN infrastructure. Here is a list of all the fields where SDN is used.

- Service provider networks
- Campus and enterprise networks
- Internet of Things
- WAN
- Network security
- Data center networking
- Network Virtualization
- switching fabrics
- traffic engineering
- access networks
- Monitoring and measurement
- Big Data
- Routing
- Optical
- Wireless

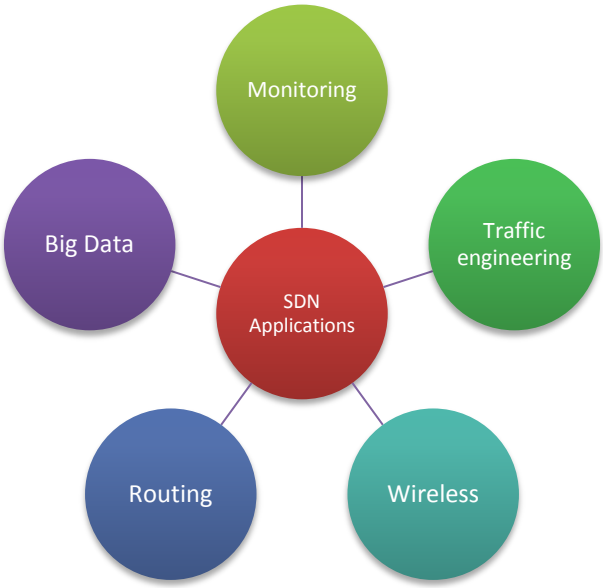

Figure 5: SDN Application

Related work: In recent years, the security aspects of the components comprising the SDN architecture, including availability, confidentiality, and integrity, have been extensively addressed through the development of security methods and solutions. Various authors have conducted reviews and discussions, highlighting several security issues associated with different levels of the SDN architecture, along with corresponding remedies. [8] [59] discusses a number of security issues related to SDN architectural levels, including issues and solutions. There are various security issues related to SDN architectural layers that are covered, along with potential remedies. The authors of [60] provides a comprehensive assessment of SDN, encompassing various aspects including architectural security. However, it does not extensively delve into the specific methods employed to tackle the aforementioned security challenges. Instead, it primarily focuses on providing an in-depth analysis and evaluation of SDN, shedding light on its architectural security concerns. [61] While recognizing that SDN has been designed without giving complete consideration to critical security aspects, such as architectural security and measures to thwart and identify attacks, it is imperative to rectify these shortcomings to bolster the overall security of SDN systems. [62] The paper explores the SDN approach to network architecture discovery and highlights potential security concerns associated with this process. It examines how SDN enables the discovery and mapping of network components, topology, and connections. Additionally, it delves into the security implications and vulnerabilities that may arise during the network architecture discovery phase in SDN. The authors of [61] aim to raise awareness about potential vulnerabilities in the stateful data plane of SDN. The paper specifically focuses on discussing various security concerns related to SDN controllers. It highlights the importance of addressing these concerns to ensure the overall security and reliability of SDN systems. [63], along with some mitigating strategies. [64] The study presents several security issues associated with the SDN architecture and offers a concise overview of various remedies. It highlights the significance of addressing these security challenges to ensure the robustness and resilience of SDN systems. The focus of research [65] is to analyze and address the security and privacy concerns specifically related to the implementation of 5G technology. The study aims to thoroughly examine the potential vulnerabilities and risks associated with 5G networks, as well as propose effective strategies and solutions to enhance the security and privacy measures in

the context of 5G. By concentrating on these aspects, the research aims to ensure the trustworthiness and confidentiality of 5G networks and the data transmitted through them. The author of [4] provide a comprehensive overview of the research conducted on SDN security. They summarize the key findings and advancements in the field of SDN security, highlighting the different approaches, methodologies, and techniques employed in various studies. The overview encompasses a wide range of topics, including threat analysis, vulnerability assessment, intrusion detection, access control, and data privacy in SDN environments. This compilation of research offers valuable insights into the current state of SDN security and serves as a foundation for further exploration and development in the field. The authors categorize their study into two main sections: (1) the architectural challenges of the SDN framework, and (2) the security benefits that SDN offers. They establish a correlation between the issues faced in SDN and the application, control, and data layers, as well as the interfaces between these layers. Additionally, they analyze relevant research efforts in SDN security, focusing on security analysis, enhancements, and solutions for problems related to the aforementioned layers and interfaces. The authors emphasize the significance of trust among the different levels of SDN and highlight the increased risk of denial-of-service (DoS) attacks due to SDN's centralization and limited flow-table capacity. Furthermore, they differentiate between widespread initiatives aimed at enhancing network security. The paper presents a technique that enables attackers to identify an SDN (Software-Defined Networking) system prior to launching a Denial-of-Service (DoS) attack. Additionally, the authors introduce a model verification system called Flover in the same study [66]. It is challenging to cover all area of SDN in a single poll because the field is complex. Numerous surveys Several surveys have been conducted to explore various aspects of the SDN paradigm, including its historical context. These surveys aim to provide an overview of the evolution of SDN, tracing its origins, development, and significant milestones. By examining the historical context of SDN, researchers gain valuable insights into the progression of this networking paradigm and its impact on the broader networking landscape. [46] Its architecture, design challenges, and applications its programming languages, fault management in SDN , traffic engineering with SDN , security concerns in SDN], and its applicability in a variety of domains . However, several articles and surveys have confined discussion to specific controllers and have considered only a few performance factors, despite the SDN control plane being an essential part of the SDN architecture. This confirms that the network's security policy is not violated by OpenFlow rules.

Table .2 Comparison of related work

*** Describe topic partially, ✓ Author coverd section, *X* describes that topic uncovered.**

References	Year	Background	Control plan	Data plane	Security Review	SDN attacks	SDN Attack solutions	Research directions
Sandra Scott-Hayward et el. [4]	2013	*	✓	✓	✓	*	*	✓
Kapil Dhamecha et el. [67]	2013	*	*	*	✓	*	*X*	✓
M Coughlin et.el. [68]	2014	✓	*X*	*X*	✓	*	*	*
S Scott-Hayward et el.[59]	2015	✓	*	*	✓	✓	✓	*
K Benzekki et el [69]	2016	✓	✓	✓	✓	✓	*	*
T Dargahi et el. [61]	2017	✓	✓	✓	✓	✓	*X*	✓
Iman Hassani et el. [70]	2018	✓	*	*	✓	*	*	*
Azath Mubarakali et el. [71]	2019	✓	✓	✓	✓	✓	*	*
Arash Shaghaghi et el. [72]	2020	✓	✓	✓	✓	✓	✓	✓
MARÍA B. JIMÉNEZ et el . [8]	2021	✓	✓	✓	✓	✓	✓	✓
MOHAMED RAHOUT et el. [9]	2022	✓	✓	✓	✓	✓	✓	✓
Abdullahi Hassan Yusuf et el. [73]	2023	✓	✓	✓	*	✓	*	*
Our work	2023	✓	✓	✓	✓	✓	✓	✓

METHODOLOGY

We utilized a qualitative technique to conduct this study by analyzing and synthesizing existing literature reviews. The focus was on identifying the fundamental requirements, risks, attacks, goals, challenges, and constraints specific to the security context of SDN. To initiate, we gathered pertinent articles from prominent scientific repositories and publishers, encompassing Web of Science, Google Scholar, Scopus, PubMed, Elsevier, IEEE, ACM, arXiv, and Springer. The exploration encompassed the timeframe starting from 2010 and extending forward to acquire the latest information available. We employed appropriate keywords related to SDN security attacks in each database and publisher to filter and select articles of interest, such as those discussing threats or security measures in SDN. Subsequently, we assessed and excluded works with limited information, categorizing all articles into various themes and categories.

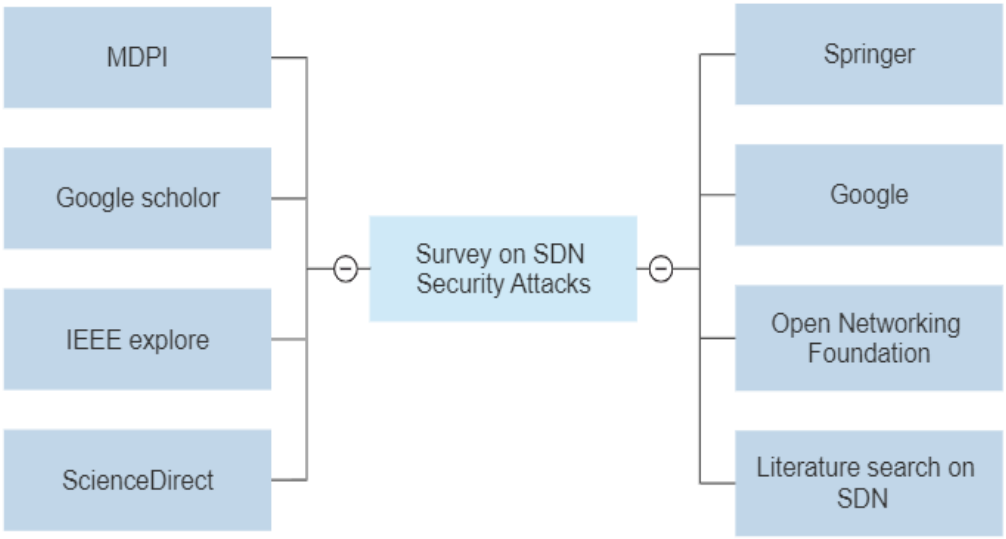

Figure 6: Survey Methodology

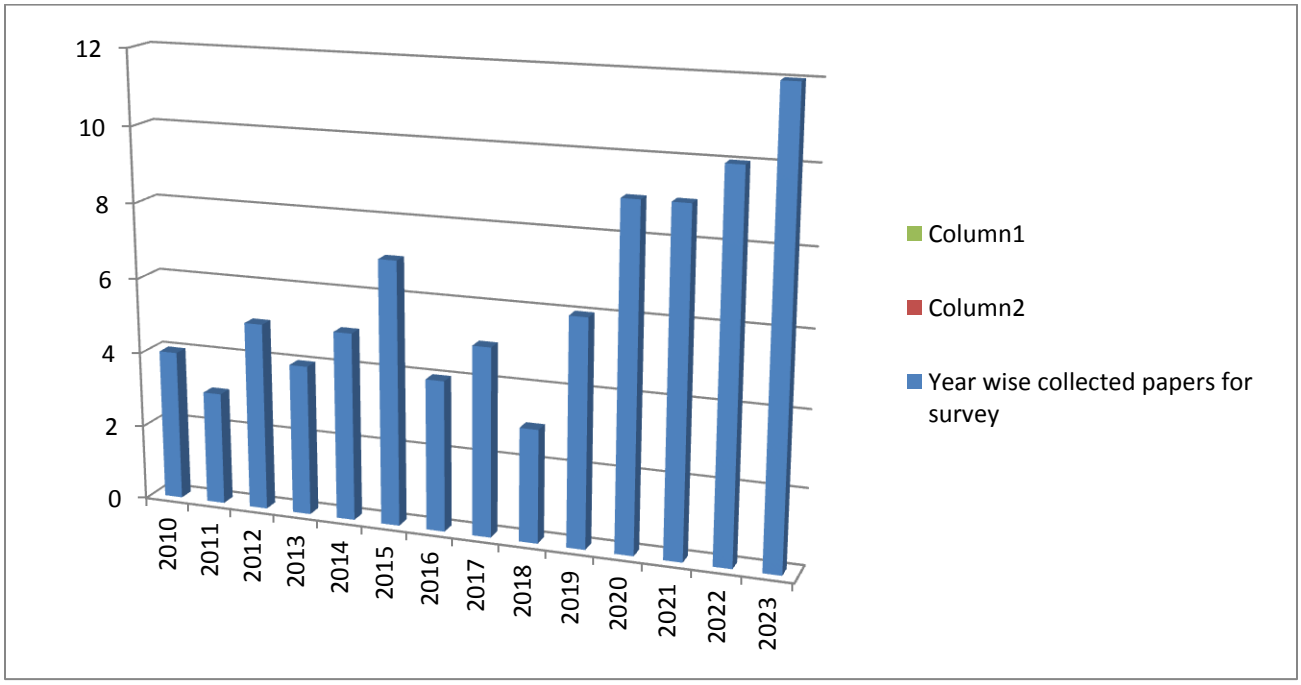

Figure 7: Year wise collected apers for survey of SDN attacks

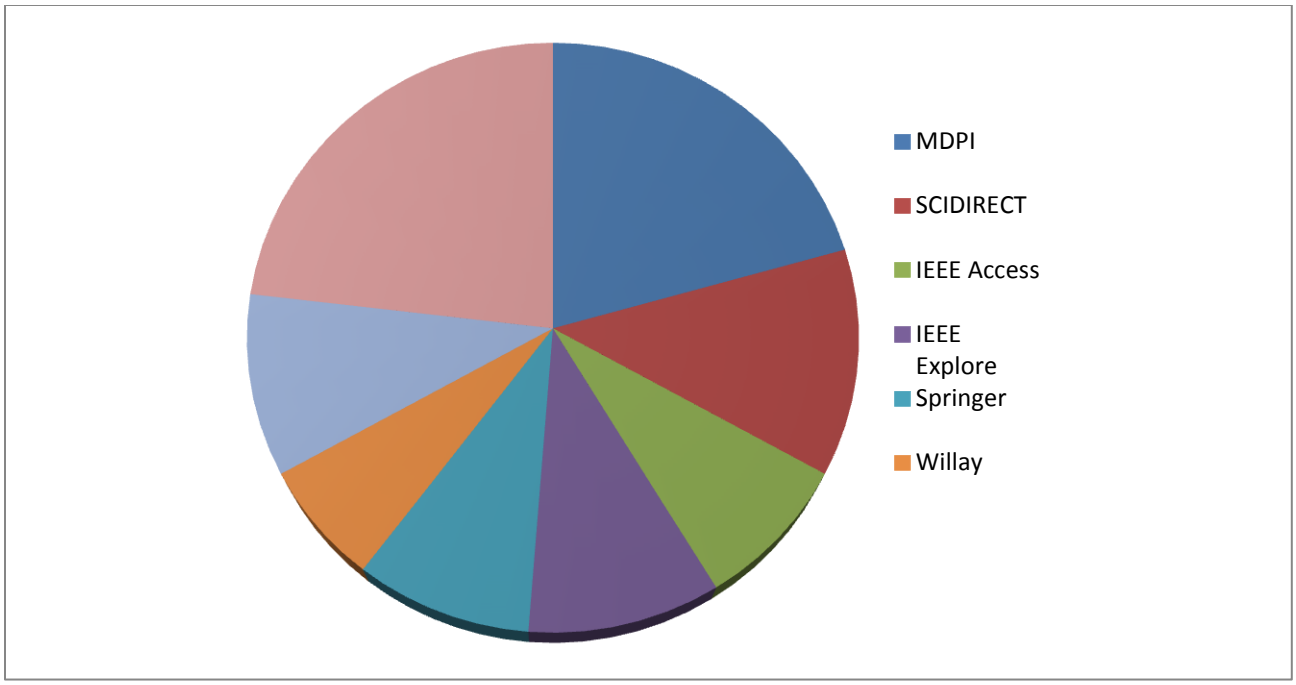

Figure 8:Publisher wise papers distribution

SDN Security Attacks Review: It is obvious that there are a number of ways the data flow to and from a controller might be abused by an outsider to undermine the controller's security. According to[74], when two controllers are situated in separate domains, data flows between them would encounter a number of security challenges, including as inter-controller trust and inter-domain trust. In this study, the analysis is conducted using a single controller model. The scope of this study does not include the security analysis of multi-controller and hierarchical controllers. The controllers have a collection of processes that implement and carry out crucial networking tasks including topology management, load balancing, access control, etc[75]. In order to evaluate the security of a process, security evaluations of these processes are essential. As more and more network security incidents occur nowadays, the challenges with network security grow more and more prevalent[76]. in succession. Traditional network security measures, however, are unable to successfully stop network intrusion and undiscovered vulnerability assaults that are becoming more sophisticated and intelligent[77]. As usual, hackers are able to defeat firewalls and intrusion detection systems (IDS), making it simple to penetrate an intranet[78]. Moving Target Defence (MTD), a cutting-edge technology that offers dynamic and proactive network defense, alters the laws of the game. This section will analyze the types of risks that an SDN network may be vulnerable to using the conventional STRIDE threats model Although the suggested model is based on conventional networks, the dangers mentioned below may be generic and apply to any networks. As an alternative, threats to SDN may be categorized according to the primary functional components of SDN that were previously outlined and the types of assaults that each component is susceptible to. Assaults on SDN can also be classified based on the type of resources or possessions that a standard SDN might possess. For example, attacks could focus on the flow tables of switches, which contain data pertaining to network management, switching, and routing. [79] A new network design called Software Defined Network (SDN) is based on centralised control and configures a network. [80]

a) ***SDN Attacks:*** A software-defined network could have more security holes than conventional networks due to five factors, which could be risky. Some of these features include a centralised controller, open programmable interfaces, the forwarding device management protocol, third-party network services, and virtualized logical networks. We briefly go over these considerations before talking about the security of SDNs. [72].

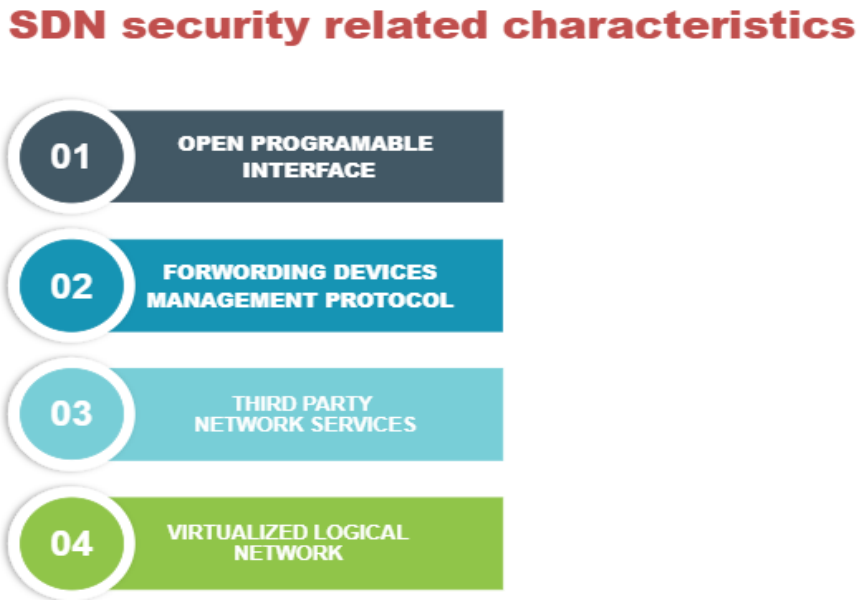

Figure 9:SDN Main Security-related Characteristics

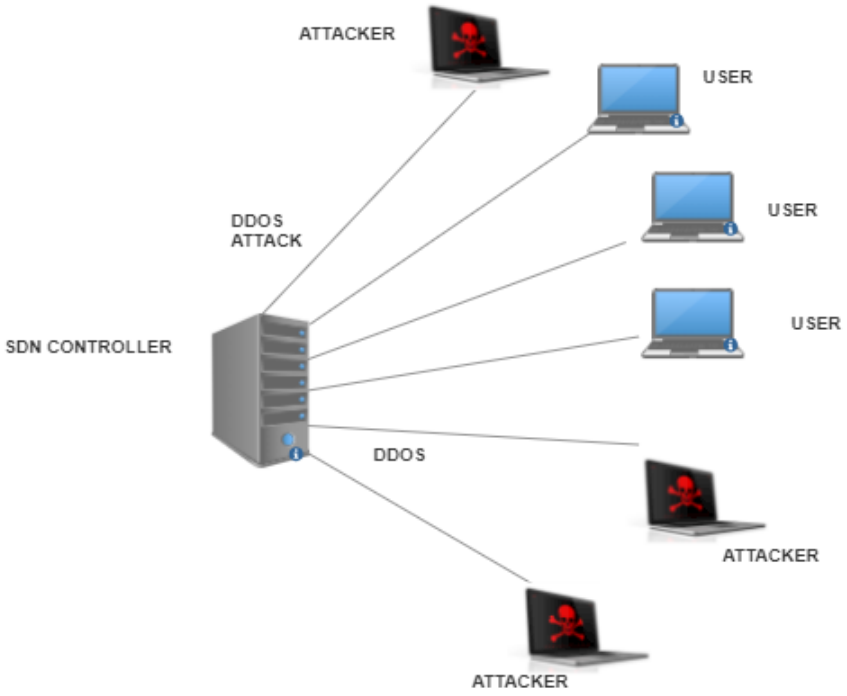

Figure 10: DDOS ATTACK

A protocol attack is any assault that targets the data plane of an SDN by taking advantage of network protocol flaws in the forwarding hardware (such as BGP attacks).

Other studies examined the security issue in SDN networks in the lack of an explicit security layer for SDN networks, classifying the many network attacks that may be launched. [81] More information on these attacks may be found in other studies, including. Numerous researchers have attempted to address different security assaults by putting forth alternative security models, as in the following studies. [82] In the area of networks, several relevant studies were sought out, including .

Current network systems and data centers are growing more and more feature-rich, complicated, and data-excessive as a result of the growth of computer networks, therefore because system designers frequently have to alter network software and coordinate the use of computers and networks based on the unique needs. [83] Traditional network designs, on the other hand, are

unable to meet the aforementioned needs of businesses, carriers, and end users. For instance, adding any new network devices or services to legacy networks is difficult due to the distribution of decision-making power among numerous network components. [24] As a result, network configuration and maintenance are becoming incredibly time-consuming and prone to mistake. Here are attacks on layers are different types .

- switch level Attack
- Controller Level Attack
- Channel Level Attack

This section will analyze the types of risks that an SDN network may be vulnerable. Although the suggested model is based on conventional networks, the dangers mentioned below may be generic and apply to any networks. As an alternative, threats to SDN may be categorized according to the primary functional components of SDN that were previously outlined and the types of assaults that each component is susceptible to [79] Attacks against SDN may also be categorized according to the kind of resources or assets that a typical SDN may have. Assaults might aim at switches' flow databases, which encompass details regarding network management, switching, and routing, as an example.

1. **Tampering:** Tampering involves intentional modification or deletion without authorization, resulting in the manipulation of data related to the network's structure, entries in flow tables, regulations, and permission listings. For instance, an intruder might endeavor to introduce flow regulations capable of causing network disruption. They might introduce firewall or flow table regulations that grant access to unauthorized hosts or deny access to authorized ones. Furthermore, cybercriminals may seek to modify topological information, potentially diverting traffic. [84] Different controllers interact with critical information while using SDN controller distribution. It is crucial to protect this communication route from being used improperly or tampered with.

2. **Spoofing:** When network data such as IP, MAC, ARP is purposefully altered to conceal the true identity of the traffic originator or attacker, this process is referred to as spoofing. For instance, people may access network resources using fake IP addresses. Spoofing frequently occurs in conjunction with other attacks such SYN flooding, Smurf, and DNS amplification. [85]

3. **Repudiation:** Repudiation refers to the disavowal of involvement in a communication segment by one of the involved parties. Non-repudiation, typically considered a legal concept rather than a technological one, aims to preempt such disavowals. [86] The sender must guarantee that packets sent to the accurate recipient are identified in the packet header, and the recipient must verify that packets originate from the authentic sender indicated in the packet header. Responsibility, which pertains to making individuals or entities answerable for their actions, is often associated with non-repudiation.

4. **Information disclosure:** Attacks focused on information disclosure primarily aim to collect data rather than directly causing harm or disruption to the network. In their initial stages, attackers will attempt to intercept network data, including details about the network's structure, node attributes, and communication exchanges between nodes, alongside the sensitive information they seek to acquire. [86] SDN architecture can have a variety of effects on scanning attacks.

5. **DoS:** Due to their negative effects on network speed, increased latency, and packet drops of legal packets, DoS assaults are among the most dangerous threats. They could even completely shut down the network or prevent it from operating. Because there is a constant flow between the controller and the switches in OpenFlow networks, DoS can be more damaging. Attackers may be tempted by the constant contact between the controller and switches to push flows between them and stop regular network operations. [87]

6. **Elevation of privilege:** Once inside the system, an intruder endeavors to enhance their level of authorization to gain entry to applications and system assets that require specific permissions. A robust and well-informed auditing process is essential to detect attempts aimed at elevating their privileges. For instance, Pedigree, a method to track executed programs by labelling them with unique identification, was proposed by [88]. Scalability is a significant issue with logging or auditing systems since they keep a lot of data, which may impact storage, memory, and bandwidth.

Attacks on SDN architecture: Every layer and interface within the SDN architecture is susceptible to a variety of attacks that can compromise network components residing within the same layer or target components in other layers. The SDN architecture consists of multiple tiers, each associated with its own set of potential threats. The subsequent paragraphs provide an overview of these architecture levels, along with the prevalent threats that have been identified at each tier. [64] Each attack is characterized by its source, which refers to the origin of the attack, such as the attack surface or threat vector.

Application layer: The abuse of fixed rights and privileges can lead to the termination of applications. When applications have unrestricted access to the network system, malicious third parties can manipulate system instructions to execute actions that primarily result in the disconnection or shutdown of critical network APIs or applications. [89] Service neutralization: Control packet handlers may be manipulated by

malicious software that has been installed successfully on top of the controller.

- Service neutralization
- Attacks to vulnerable northbound APIs
- Application termination by abusing fixed privileges and authority
- Attacks to vulnerable northbound APIs

1. ***Control layer:*** Compromised applications and malicious data-plane devices can exploit controller vulnerabilities and configuration errors to achieve different objectives. These may include executing system commands that result in the controller's termination, extracting sensitive data from internal network storage, or diverting data meant for legitimate devices. [90] The SDN controller transmits rules and instructions from the application layer to the lower layer. When necessary, the network controller gathers network information from various data layer devices and distributes them to higher level applications. Statistics cover network status and numerous

2. **Infrastructure layer:** By assuming the role of the controller, an attacker can isolate the target switch. The target switch creates a connection with the fake controller rather than the real one once the attacker has taken over the controller's functionality using an ARP poisoning attack. The switch is thus unplugged from the network. [91]

SDN Threats and vulnerabilities: A cyber threat refers to malicious efforts aimed at unauthorized access or control of computer networks, information technology (IT) devices (such as computers or smartphones), and operational technology (OT) equipment, including programmable logic controllers (PLCs), supervisory control and data acquisition (SCADA) systems, and other acquisition technologies. [92] With each passing day, new cyber threat vectors continue to emerge, posing challenges in implementing effective preventive and defensive measures. The STIX v1.2.1 standard [38] defines TTP (tactics, methods, and procedures) as the behaviours and resources that attackers employ to carry out their assaults. In the STIX v2.1 specification [8], the same notion is shown in a different way: In the previous iteration, TTP type was broken down into attack pattern, malware, and tool object categories. [93] The Internet of Things (IoT), a rapidly evolving communication technology, is having a significant influence on how traditional network communication approaches are changing. The range of IoT applications completely encapsulates our quality of life, and by combining with other technologies, this variety is increased. However, this legality also leaves IoT open to a number of significant security risks, necessitating the use of noteworthy countermeasures. In this scientific study, we suggested an intrusion detection system (IDS) that makes use of Software Defined Networking (SDN) and Deep Learning (DL) to protect against new cyber threats in the Internet of Things. In IoT communications, our suggested approach (DNNLSTM) can defend against a wide range of typical and rare cyber threats.

Table.3 SDN Attacks Review.

Attack Type on SDN	Attack goal	Existing Survey	Existing Work on Attack Detection/Defense	Year	Authors with Reference for Existing Survey
Social Engineering	social engineering is an attack on information security for accessing systems or networks.	Preventing Social Engineering Attacks	Surveying and Analyzing Social Engineering Defense Mechanisms and Information Security Policies [94]	2022	W Syafitri, Z Shukur, U Asma'Mokhtar[95]
Phishing	To steal credentials and credit card numbers.	Detecting and Mitigating Phishing Attacks: A Survey	Performance Analysis of Phishing Attack Prevention in Cyberspace using Software-Defined Networking and Deep Machine Learning with Cantina Approach (DMLCA): A System Study **[96]**	2018	Jibi Mariam Biju1, Anju J Prakash2 [97]
DDOS	make online services unavailable	A Systematic Review of Machine Learning Techniques for DDoS	Methods, Practices, and Solutions for DDoS Attack Detection and Mitigation using SDN [98]	2023	TE Ali, YW Chong, S Manickam [99]

		Attack Detection in SDN			
DOS	Disrupt the availability of a computer system	Threats, Challenges, and Potential Solutions for the Security of Low Power Wide Area Networks: A Survey	A Review of Detection Techniques for Distributed Denial of Service Attacks on Software-Defined Networking Controllers [100]	2020	KO Adefemi Alimi, K Ouahada, AM Abu-Mahfouz[101]
SQL injection	SQL query to obtain unauthorized access to a database	Surveying Cybersecurity Vulnerabilities , Attacks, and Solutions in the Medical Domain.	A Proposed Technique for Simultaneous Detection of DDoS and SQL Injection Attacks. [102]	2019	A Razaque, F Amsaad, MJ Khan, S Hariri[103]
MITM	to intercept and manipulate communication between two parties without their knowledge.	A Comprehensive Review of Multi-Channel Man-in-the-Middle Attacks against Protected Wi-Fi Networks.	CBNA-RF: A Machine Learning-Based MitM Detection and Defense Mechanism for Large-Scale SDN Environments. [104]	2022	M Thankappan, H Rifà-Pous, C Garrigues[105]
ARP spoofing		A Comprehensive Survey on Security Attacks in SDN Networks: Analysis and Findings.	A Flexible Software Architecture for Mitigating ARP Spoofing-Based Attacks in the SDN Data Plane Layer. [106]	2022	AN Alhaj, N Dutta [81]
side channel attack	information leaked through unintended channels	Side channel Attack-Survey	SDN security through system call learning[107]	2011	G Joy Persial, M Prabhu[108]
eavesdropping	unauthorized individual intercepts and listens to private or sensitive communications	SDN Security Problems and Solutions Analysis	Mitigating Eavesdropping Attacks in the Software-Defined Networking Data Plane. [109]	2015	A Feghali, R Kilany, M Chamoun [110]
Packet _In message Flooding	consuming SDN controller processing power, memory, and network bandwidth.	Analysis of Control Plane Packet-In Arrival Rate for Detection and Mitigation of Denial-of-Service Saturation Attacks in Software-	Mitigating Packet-In Message Flooding Attacks in the SDN Context: Defense Mechanisms and Strategies. [111]	2018	F Khellah [112]

		Defined Networks.			
Ransomware	an attacker encrypts files on a victim's computer or network and demands a ransom payment in exchange for restoring access to the encrypted data	"Exploring Situational Awareness of Ransomware Attacks: Parameters for Detection and Prevention - A Survey."	RDS3: A Stealthy Spare Space-Based Defense Strategy for Ransomware. [113]	2019	JAH Silva, LIB López, ÁLV Caraguay[114]
Sniffing	attacker intercepts and captures network traffic to extract sensitive information	Comprehensive Survey and Analysis of Security Attacks in SDN Networks.	Detection of Sniffing through Network Traffic Probing and Machine Learning. [115]	2022	Ali Nadim Alhaj and Nitul Dutta [81]
Network Manipulation	attacker attempting to intercept data transfering via the network and specifically targeting the controllers to inflict damage.	Analysis of Security Attacks in SDN Network: A Comprehensive Survey	Network Manipulation Techniques Utilizing Network Scanning in SDN. [116]	2019	Ali Nadim Alhaj and Nitul Dutta[81]
Forge controller Attack	Attacker will take over the charge of the whole network	**"A Survey on Security-Aware Measurement in SDN"**	**"No work Exist on this hardware base attack that we found in our survey"**	2018	H Zhang, Z Cai, Q Liu, Q Xiao, Y Li [117]

DISCUSSION

In this survey we have tried our best to take brief overview to every security attack for this purpose we downloaded a lot of research paper from different platforms. SDN have a great impact on tradition network due to its centralized monitoring and management system that improve the security system. Although SDN have great energetic mechanism to defend the threats and attacks due to its advanced systems but also have some vulnerabilities that makes SDN a piece of cake for hackers and attack. Why I am saying this because in different surveys and reviews we found a lot of vulnerabilities in software defined networking specially we found that a controller that control over the network and have lot of vulnerabilities and threats. Different attacks solutions exist but still we have hardware base challenges in SDN for example we found that if a controller and switch loss connection then this will become a major issue. Open v switch will come in Standalone mode or Fail secure mode and switch will try to reconnect with controller a forge controller attack may happen and attacker will take charge of whole network and network will be hijacked. So this should be solved a hardware base problem. We have a comparison survey also with existing surveys on SDN security and Attacks.

Conclusion: In this examination, we present a concise summary of security threats and various challenges associated with SDN. We've determined that considerable progress has been made, yet various vulnerabilities persist across multiple levels and strata. This paper has deliberated on security concerns pertaining to SDN architecture, underscoring unresolved matters and sketching prospective directions for the future. Software Defined Networking (SDN) is an emerging technology that has garnered substantial attention in both corporate and academic spheres. By employing centralized control, SDN simplifies the overall network administration and customization. This study meticulously explores all the security issues while adhering to the SDN framework and addressing security concerns. This study also provided evidence of the security issues with the taxonomy design and outlined further research in this publication. .Different attack solutions are available, but there are still hardware-based problems with SDN. For instance, we discovered that if a controller and switch lose their connection, there would be serious problems. When an open v switch enters Standalone or Fail secure mode, it

will attempt to reconnect with the controller. If a forge controller attack occurs, the attacker may then seize control of the whole network, leading to a network hijacking. Therefore, this should be resolved as a hardware-based issue. In future SDN security researchers should conduct work on Forge-Controller Attack that is based on hardware base challenge that is not exist still. Additionally, we conducted a comparative poll with other studies on SDN security and attacks and found that our survey is much more reliable and in well manner as compared to other existing surveys although many surveys were found very excellent and appreciating work.

REFERENCES

Braun, W. and M. Menth, *Software-defined networking using OpenFlow: Protocols, applications and architectural design choices.* Future Internet, 2014. 6(2): p. 302-336.

Isyaku, B., et al., *Software defined networking flow table management of openflow switches performance and security challenges: A survey.* Future Internet, 2020. 12(9): p. 147.

Alsaeedi, M., M.M. Mohamad, and A.A. Al-Roubaiey, *Toward adaptive and scalable OpenFlow-SDN flow control: A survey*. IEEE Access, 2019. 7: p. 107346-107379.

Scott-Hayward, S., G. O'Callaghan, and S. Sezer. *SDN security: A survey*. in *2013 IEEE SDN For Future Networks and Services (SDN4FNS)*. 2013. IEEE.

Berde, P., et al. *ONOS: towards an open, distributed SDN OS*. in *Proceedings of the third workshop on Hot topics in software defined networking*. 2014.

Shalimov, A., et al. *Advanced study of SDN/OpenFlow controllers*. in *Proceedings of the 9th central & eastern european software engineering conference in russia*. 2013.

Hussain, M., et al., *Software-defined networking: Categories, analysis, and future directions.* Sensors, 2022. 22(15): p. 5551.

Jimenez, M.B., et al., *A survey of the main security issues and solutions for the SDN architecture.* IEEE Access, 2021. 9: p. 122016-122038.

Rahouti, M., et al., *SDN security review: Threat taxonomy, implications, and open challenges.* IEEE Access, 2022. 10: p. 45820-45854.

Blial, O., M. Ben Mamoun, and R. Benaini, *An overview on SDN architectures with multiple controllers.* Journal of Computer Networks and Communications, 2016. 2016.

Pradhan, A. and R. Mathew, *Solutions to vulnerabilities and threats in software defined networking (SDN).* Procedia Computer Science, 2020. 171: p. 2581-2589.

Paliwal, M., D. Shrimankar, and O. Tembhurne, *Controllers in SDN: A review report.* IEEE access, 2018. 6: p. 36256-36270.

Bannour, F., S. Souihi, and A. Mellouk, *Distributed SDN control: Survey, taxonomy, and challenges.* IEEE Communications Surveys & Tutorials, 2017. 20(1): p. 333-354.

Karakus, M. and A. Durresi, *A survey: Control plane scalability issues and approaches in software-defined networking (SDN).* Computer Networks, 2017. 112: p. 279-293.

https://openflow.stanford.edu/display/Beacon/Home.

Gude, N., et al., *NOX: towards an operating system for networks.* ACM SIGCOMM computer communication review, 2008. 38(3): p. 105-110.

Pox. http://www.noxrepo.org/pox/about-pox/.

Mul. http://sourceforge.net/p/mul/wiki/Home/.

Nascimento, M.R., et al. *Virtual routers as a service: the routeflow approach leveraging software-defined networks*. in *Proceedings of the 6th International Conference on Future Internet Technologies*. 2011.

Sherwood, R., et al., *Carving research slices out of your production networks with OpenFlow.* ACM SIGCOMM Computer Communication Review, 2010. 40(1): p. 129-130.

Open vswitch and ovs-controller. http://openvswitch.org/.

The nodeflow openflow controller. http://garyberger.net/?p=537.

Ryu. http://osrg.github.com/ryu/.

Simple Network Access Control (SNAC). http://www.openflow.org/wp/snac/.

Floodlight, an open sdn controller. http://floodlight.openflowhub.org/.

Helios by nec. http://www.nec.com/.

Trema openflow controller framework. https://github.com/trema/trema.

Ng, E.M., Z. Cai, and A. Cox, *A system for scalable openflow control.* Rice University: Houston, TX, USA, 2010.

Phemius, K., M. Bouet, and J. Leguay. *Disco: Distributed multi-domain sdn controllers*. in *2014 IEEE network operations and management symposium (NOMS)*. 2014. IEEE.

Matsumoto, S., S. Hitz, and A. Perrig. *Fleet: Defending SDNs from malicious administrators*. in *Proceedings of the third workshop on Hot topics in software defined networking*. 2014.

Shin, S., et al. *Rosemary: A robust, secure, and high-performance network operating system*. in *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. 2014.

ONOS .https://opennetworking.org/onos/.

Islam, M.S., et al., *A Survey on SDN and SDCN Traffic Measurement: Existing Approaches and*

Research Challenges. Eng, 2023. 4(2): p. 1071-1115.

Cabaj, K., et al. *SDN Architecture Impact on Network Security*. in *FedCSIS (Position Papers)*. 2014.

Sung, Y., et al., *FS-OpenSecurity: a taxonomic modeling of security threats in SDN for future sustainable computing.* Sustainability, 2016. 8(9): p. 919.

Onyema, E.M., et al., *A security policy protocol for detection and prevention of internet control message protocol attacks in software defined networks.* Sustainability, 2022. 14(19): p. 11950.

Algarni, S., et al., *BCNBI: A Blockchain-Based Security Framework for Northbound Interface in Software-Defined Networking.* Electronics, 2022. 11(7): p. 996.

Gadze, J.D., et al., *An investigation into the application of deep learning in the detection and mitigation of DDOS attack on SDN controllers.* Technologies, 2021. 9(1): p. 14.

Isong, B., T. Kgogo, and F. Lugayizi, *Trust establishment in SDN: controller and applications.* International Journal of Computer Network and Information Security, 2017. 9(7): p. 20.

Garg, S., et al., *Hybrid deep-learning-based anomaly detection scheme for suspicious flow detection in SDN: A social multimedia perspective.* IEEE Transactions on Multimedia, 2019. 21(3): p. 566-578.

Elmrabit, N., et al. *Evaluation of machine learning algorithms for anomaly detection*. in *2020 international conference on cyber security and protection of digital services (cyber security)*. 2020. IEEE.

Xie, J., et al., *A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges.* IEEE Communications Surveys & Tutorials, 2018. 21(1): p. 393-430.

Sultana, N., et al., *Survey on SDN based network intrusion detection system using machine learning approaches.* Peer-to-Peer Networking and Applications, 2019. 12: p. 493-501.

Ashraf, J. and S. Latif. *Handling intrusion and DDoS attacks in Software Defined Networks using machine learning techniques*. in *2014 National software engineering conference*. 2014. IEEE.

Ali, S.T., et al., *A survey of securing networks using software defined networking.* IEEE transactions on reliability, 2015. 64(3): p. 1086-1097.

Nunes, B.A.A., et al., *A survey of software-defined networking: Past, present, and future of programmable networks.* IEEE Communications surveys & tutorials, 2014. 16(3): p. 1617-1634.

Hu, F., Q. Hao, and K. Bao, *A survey on software-defined network and openflow: From concept to implementation.* IEEE Communications Surveys & Tutorials, 2014. 16(4): p. 2181-2206.

Abdou, A., D. Barrera, and P.C. van Oorschot. *What lies beneath? Analyzing automated SSH bruteforce attacks*. in *International conference on PASSWORDS*. 2015. Springer.

LongTail, "LongTail Log Analysis." http://longtail.it.marist.edu/honey/.

[Online; accessed 21-Mar-2016].

V. Sommer, "Anamoly Detection in SDN Control Plane," Master's

thesis, Technical University of Munich, Munich, Germany, 2014.

Qazi, Z.A., et al. *Application-awareness in SDN*. in *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. 2013.

OpenFlow: https://opennetworking.org/sdn-resources/customer-case-studies/openflow/.

McKeown, N., et al., *OpenFlow: enabling innovation in campus networks.* ACM SIGCOMM computer communication review, 2008. 38(2): p. 69-74.

Lara, A., A. Kolasani, and B. Ramamurthy, *Network innovation using openflow: A survey.* IEEE communications surveys & tutorials, 2013. 16(1): p. 493-512.

Benton, K., L.J. Camp, and C. Small. *OpenFlow vulnerability assessment*. in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*. 2013.

Dely, P., A. Kassler, and N. Bayer. *Openflow for wireless mesh networks*. in *2011 proceedings of 20th international conference on computer communications and networks (ICCCN)*. 2011. IEEE.

Li, T., J. Chen, and H. Fu. *Application scenarios based on SDN: an overview*. in *Journal of Physics: Conference Series*. 2019. IOP Publishing.

Thirupathi, V., et al., *A comprehensive review on sdn architecture, applications and major benifits of SDN.* International Journal of Advanced Science and Technology, 2019. 28(20): p. 607-614.

Ahmad, I., et al., *Security in software defined networks: A survey*. IEEE Communications Surveys & Tutorials, 2015. 17(4): p. 2317-2346.

Kreutz, D., et al., *Software-defined networking: A comprehensive survey.* Proceedings of the IEEE, 2014. 103(1): p. 14-76.

Dargahi, T., et al., *A survey on the security of stateful SDN data planes.* IEEE Communications Surveys & Tutorials, 2017. 19(3): p. 1701-1725.

Khan, S., et al., *Topology discovery in software defined networks: Threats, taxonomy, and state-of-the-art.* IEEE Communications Surveys & Tutorials, 2016. 19(1): p. 303-324.

Han, T., et al., *A comprehensive survey of security threats and their mitigation techniques for next-*

generation SDN controllers. Concurrency and Computation: Practice and Experience, 2020. 32(16): p. e5300.

Chica, J.C.C., J.C. Imbachi, and J.F.B. Vega, *Security in SDN: A comprehensive survey.* Journal of Network and Computer Applications, 2020. 159: p. 102595.

Khan, R., et al., *A survey on security and privacy of 5G technologies: Potential solutions, recent advancements, and future directions.* IEEE Communications Surveys & Tutorials, 2019. 22(1): p. 196-248.

Son, S., et al. *Model checking invariant security properties in OpenFlow*. in *2013 IEEE international conference on communications (ICC)*. 2013. IEEE.

Dhamecha, K. and B. Trivedi, *Sdn issues-a survey.* International Journal of Computer Applications, 2013. 73(18).

Coughlin, M., *A survey of SDN security research.* University of Colorado Boulder, 2014.

Benzekki, K., A. El Fergougui, and A. Elbelrhiti Elalaoui, *Software-defined networking (SDN): a survey*. Security and communication networks, 2016. 9(18): p. 5803-5833.

Hassani, I. and A. Moayeri, *SDN Security: A Survey.* International Journal of Information, Security and Systems Management, 2018. 7(1): p. 785-792.

Mubarakali, A. and A.S. Alqahtani. *A survey: Security threats and countermeasures in software defined networking*. in *2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT)*. 2019. IEEE.

Shaghaghi, A., et al., *Software-defined network (SDN) data plane security: issues, solutions, and future directions.* Handbook of Computer Networks and Cyber Security: Principles and Paradigms, 2020: p. 341-387.

Yusuf, A.H., *Major Security Threats of Software Defined Network.*

"SDN Threat Analysis ":https://www.ietf.org/proceedings/93/.

Arbettu, R.K., et al. *Security analysis of OpenDaylight, ONOS, Rosemary and Ryu SDN controllers*. in *2016 17th International telecommunications network strategy and planning symposium (Networks)*. 2016. IEEE.

Shi, Y., et al., *Chaos: An sdn-based moving target defense system.* Security and Communication Networks, 2017. 2017.

Jafarian, J.H., E. Al-Shaer, and Q. Duan. *Adversary-aware IP address randomization for proactive agility against sophisticated attackers*. in *2015 IEEE conference on computer communications (INFOCOM)*. 2015. IEEE.

Open Networking Foundation, "OpenFlow1.1.0 specification," 2011, http://www.openflow.org/documents/openflow-spec-v1.1.0 .pdf.

Alsmadi, I. and D. Xu, *Security of software defined networks: A survey.* Computers & security, 2015. 53: p. 79-108.

Jantila, S. and K. Chaipah, *A security analysis of a hybrid mechanism to defend DDoS attacks in SDN.* Procedia Computer Science, 2016. 86: p. 437-440.

Alhaj, A.N. and N. Dutta, *Analysis of security attacks in SDN network: A comprehensive survey*. Contemporary Issues in Communication, Cloud and Big Data Analytics: Proceedings of CCB 2020, 2022: p. 27-37.

Scott-Hayward, S., S. Natarajan, and S. Sezer, *A survey of security in software defined networks.* IEEE Communications Surveys & Tutorials, 2015. 18(1): p. 623-654.

Li, W., W. Meng, and L.F. Kwok, *A survey on OpenFlow-based Software Defined Networks: Security challenges and countermeasures.* Journal of Network and Computer Applications, 2016. 68: p. 126-139.

Othman, O.M. and K. Okamura, *Securing distributed control of software defined networks.* International Journal of Computer Science and Network Security (IJCSNS), 2013. 13(9): p. 5.

Yao, G., J. Bi, and P. Xiao. *Source address validation solution with OpenFlow/NOX architecture*. in *2011 19Th IEEE international conference on network protocols*. 2011. IEEE.

ISO, I., *7498-2. information processing systems open systems interconnection basic reference model-part 2: Security architecture.* ISO Geneva, Switzerland, 1989.

Zaalouk, A., et al. *OrchSec: An orchestrator-based architecture for enhancing network-security using network monitoring and SDN control functions*. in *2014 IEEE Network Operations and Management Symposium (NOMS)*. 2014. IEEE.

Ramachandran, A., et al. *Securing enterprise networks using traffic tainting*. in *Proc. SIGCOMM*. 2009.

Röpke, C., *SDN malware: problems of current protection systems and potential countermeasures.* Sicherheit 2016-Sicherheit, Schutz und Zuverlässigkeit, 2016.

Röpke, C. and T. Holz. *Sdn rootkits: Subverting network operating systems of software-defined networks*. in *Research in Attacks, Intrusions, and Defenses: 18th International Symposium, RAID 2015, Kyoto, Japan, November 2-4, 2015. Proceedings 18*. 2015. Springer.

Yoon, C., et al. *A security-mode for carrier-grade sdn controllers*. in *Proceedings of the 33rd Annual Computer Security Applications Conference*. 2017.

, https://www.oasis-open.org/standards/., https: //www.us-cert.gov/bsi/articles/knowledge/attack-patterns/introductionto-attack-patterns.

Alharthi, D.N., *Social Engineering Defense Mechanisms and InfoSec Policies: A Survey and Qualitative Analysis*. 2021: University of California, Irvine.

Syafitri, W., et al., *Social engineering attacks prevention: A systematic literature review.* IEEE Access, 2022. 10: p. 39325-39343.

Ravi, R., *A performance analysis of Software Defined Network based prevention on phishing attack in cyberspace using a deep machine learning with CANTINA approach (DMLCA).* Computer Communications, 2020. 153: p. 375-381.

Biju, J.M. and A.J. Prakash, *Survey on Phishing attack Detection and Mitigation.* 2018.

Bawany, N.Z., J.A. Shamsi, and K. Salah, *DDoS attack detection and mitigation using SDN: methods, practices, and solutions.* Arabian Journal for Science and Engineering, 2017. 42: p. 425-441.

Ali, T.E., Y.-W. Chong, and S. Manickam, *Machine Learning Techniques to Detect a DDoS Attack in SDN: A Systematic Review.* Applied Sciences, 2023. 13(5): p. 3183.

Aladaileh, M.A., et al., *Detection techniques of distributed denial of service attacks on software-defined networking controller–a review.* IEEE Access, 2020. 8: p. 143985-143995.

Adefemi Alimi, K.O., et al., *A survey on the security of low power wide area networks: Threats, challenges, and potential solutions.* Sensors, 2020. 20(20): p. 5800.

Hashem, I., et al., *A proposed technique for simultaneously detecting DDoS and SQL injection attacks.* Int. J. Comput. Appl, 2021. 183(11): p. 50-57.

Razaque, A., et al., *Survey: Cybersecurity vulnerabilities, attacks and solutions in the medical domain.* IEEE Access, 2019. 7: p. 168774-168797.

Sebbar, A., et al., *MitM detection and defense mechanism CBNA-RF based on machine learning for large-scale SDN context.* Journal of Ambient Intelligence and Humanized Computing, 2020. 11: p. 5875-5894.

Thankappan, M., H. Rifà-Pous, and C. Garrigues, *Multi-channel man-in-the-middle attacks against protected wi-fi networks: A state of the art review*. Expert Systems with Applications, 2022: p. 118401.

Buzura, S., et al., *An Extendable Software Architecture for Mitigating ARP Spoofing-Based Attacks in SDN Data Plane Layer.* Electronics, 2022. 11(13): p. 1965.

Chasaki, D. and C. Mansour. *Sdn security through system call learning*. in *2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. 2021. IEEE.

Joy Persial, G., M. Prabhu, and R. Shanmugalakshmi, *Side channel attack-survey.* Int. J. Adv. Sci. Res. Rev, 2011. 1(4): p. 54-57.

Aseeri, A., N. Netjinda, and R. Hewett. *Alleviating eavesdropping attacks in software-defined networking data plane*. in *Proceedings of the 12th Annual Conference on Cyber and Information Security Research*. 2017.

Feghali, A., R. Kilany, and M. Chamoun. *SDN security problems and solutions analysis*. in *2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS)*. 2015. IEEE.

Gao, D., et al., *Defending against Packet-In messages flooding attack under SDN context.* Soft Computing, 2018. 22: p. 6797-6809.

Khellah, F., *Control plane packet-in arrival rate analysis for denial-of-service saturation attacks detection and mitigation in software-defined networks.* Arabian Journal for Science and Engineering, 2019. 44(11): p. 9349-9362.

Subedi, K.P., et al. *RDS3: Ransomware defense strategy by using stealthily spare space*. in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*. 2017. IEEE.

Silva, J.A.H., et al., *A survey on situational awareness of ransomware attacks—detection and prevention parameters.* Remote Sensing, 2019. 11(10).

Gregorczyk, M., et al., *Sniffing detection based on network traffic probing and machine learning.* IEEE Access, 2020. 8: p. 149255-149269.

Thangavel, M. and V. Pavithra, *Network manipulation using network scanning in SDN*, in *Artificial Intelligence and Security Challenges in Emerging Networks*. 2019, IGI Global. p. 85-123.

Zhang, H., et al., *A survey on security-aware measurement in SDN.* Security and Communication Networks, 2018. 2018.