A Survey on Software Defined Networks: Technical Challenges, Recent Advances and Security Issues

J. Angel Antonette Keziah
Pg Scholar
Dept. of Electronics and Communication
PSG College of Technology
Coimbatore, India

Mr. N. Saritakumar
Assistant Professor
Dept. Of Electronics and Communication
PSG College of Technology
Coimbatore, India

Abstract: **Today a large amount of communication occurs within data centres where a huge number of virtual servers provide service providers with the infrastructure necessary for their applications and services. Software-Defined Networking (SDN) is a new principle in the networking paradigm, which makes a communication network programmable. With this utility, security properties can be redesigned. SDN provides new facts for security applications to implement security services. With SDN the decisions are made by a centralized SDN controller that decides upon the best path and instructs the devices along this path as to what action each should perform. SDN has encountered trust issues and general concerns related to whether software-based solutions are as reliable as hardware based solutions. SDN has also encountered these issues and discussion of these issues continues in this paper. Concerns about trust remain a problem for the growing number of cloud-based services where it may lead to loss of personal integrity and other security risks. As a relatively new technology, SDN is still immature and has a number of vulnerabilities. The security risks increases as with most software-based solutions. In this paper, we will look at the next step in the virtualization revolution, the virtualized network. This paper deals with the technical challenges, recent trends and security issues in SDN and a prediction of the directions of future research in SDN security.**

Keywords: SDN-software defined network, security issues, vulnerabilities and virtualization.

I. INTRODUCTION

Software-Defined Networking (SDN) concept was first time introduced in 2010 as the new networking paradigm which aims to ease the control and the management of a computer network environment.SDN can be explained as an architectural principle where the networks control and the management are centralized and decoupled from data plane, thus making the network programmable. Software-Defined Networking (SDN) is a refactoring of the relationship between network devices and the software that controls them. Traditionally, the data and the control planes in the Ethernet networking devices (and most of the communication principles) have been tied together. This means, the prevailing operating system and its features with the provided hardware are implemented in a single device. Therefore, network devices, such as switches, routers, firewalls, etc., are built with the intelligence of handling traffic relative to the adjacent devices. This makes the intelligence distributed and scattered in the network. In addition, most of the network devices are Command Line Interface (CLI) based and configuration is done separately per device, making configuration slow and prone to errors. The data plane has a sensible layering model which is known by the name Open Systems Interconnection model (OSI model) is a well know model in the networking industries and academies. The OSI model enables network applications and services to isolate the data operations to a single layer and provide interfaces between layers. This has enforced developers to develop and improve the operations without concerning the other layers. This type of layering models are used in many other fields like operating system and the overall view and the interactions are understood. As a result, we can witness increase in the development and research in these fields. Networks control and management plane, which was not available, is essentially needed. This creates the need to invent a new networking architecture; SDN.SDN architecture decouples control from data plane and provides it a new layering model. Following are the layering scheme for SDN architecture.

II. LAYERS OF SDN ARCHITECTURE

As seen from the Figure 1, SDN architecture is divided into three layers: application layer, control layer, and infrastructure layer. This architecture and arrangement of control and management, provides the possibility to centralize the state of the network and the intelligence into one part of the network. This, enhances the property of network programmability, the network industry can start to innovate and enable differentiation in the developing process Furthermore, programmability accelerates creativity and introduction of new network features and services.

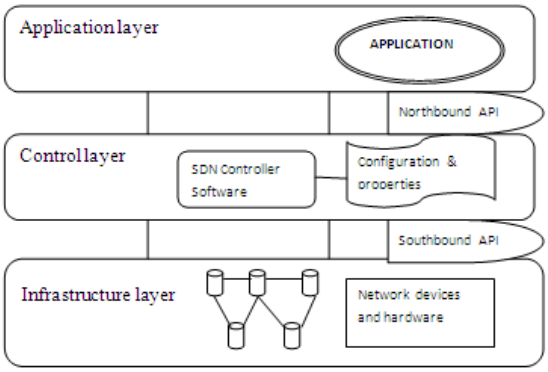

Fig. 1. The SDN Stack.

With centralization, SDN simplifies provisioning while optimizing performance and granularity of the policy management. Therefore, SDN can make networks become more scalable, flexible and proactive. SDN architecture stack abstracts and decouples hardware from software, control`1 plane from forwarding plane, and physical from logical configuration.

A. Infrastructure Layer

This is the layer where all the hardware exists are connected physically. On these hardware devices runs software which provides a control data plane interface (Southbound API) which is used to communicate with the upper level Control layer.

B. *Control Layer*

Control layer is the most important layer in the architecture. The controller does the work of communicating to all the network devices in the infrastructure and keeps track of the topology. While exchanging information of the network state with upper layer applications (through (Northbound API), the controller translates their commands to the network devices to have respective and desired network behavior.

C. *Application Layer*

Application layer is the layer where all the features, services and policies are defined. Applications request the information of network devices and the topology in order to act upon it. The applications can be create with the necessary features according to the changes in the network. When the network topology, feature, or policy requirements changes, applications have the control to change dynamically the network behavior from one single point between these layers, there are Application Programming Interfaces (APIs) which provide the essential communication tools between the layers. The Northbound API is provided by the controller and the applications have to manage their communication to the controller through it. The Southbound API provides the necessary communication between the controller and the network devices. In the following section, a protocol providing this feature shall be further discussed.

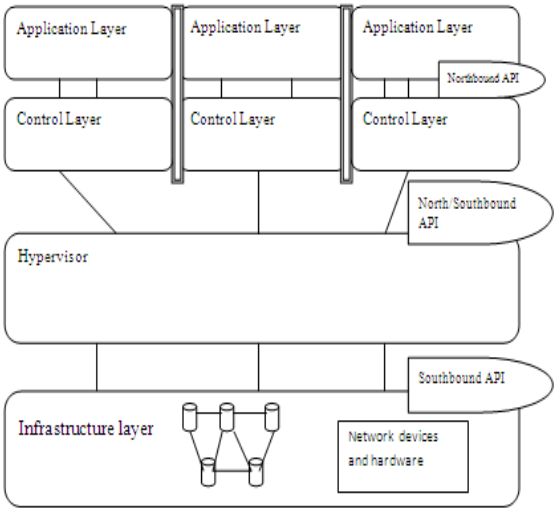

Fig. 2. The SDN Stack with Hypervisor (Slicing).

The SDN layer model can be adjusted to fit and satisfy a data center scheme where virtualization may need logical network segment slicing. Hypervisor, the controller of controllers, is an extra layer which is added in the model (see Figure 2) provides logical slicing.. In other words, the Hypervisor allows every individual controller to control only their own hosts (physical or virtual) on the network without affecting other parts of the network.SDN brings new challenges in networking technology and in this paper it focuses on the network security. A programmable network provides full control of a network thus, bringing more capabilities to handle security in the network. However, SDN has gained its reputation rapidly and is the biggest hype word in networking business.

III. WHY CENTRALIZE?

Maintaining a network-wide view of resources is possible by the logically centralized control plane which can then be exposed to the application layer. To provide such a centralized architecture, SDN uses one or more network elements that interface with the SDN controller. The benefit of centralized networks is simplified network management, and improved agility. Centralization equips networks for programmability, which in turn increases autonomy. One possibility enabled by programmability is the automatic detection and mitigation of DDOS attacks, which results in rapid resolution of any problems that may arise. Programmability also allows network resources to be shared automatically, which together with the capability to create virtual networks created on top of existing network infrastructure enables automatic sharing by multiple tenants.

IV. SDN BENEFITS

SDN, with its inherent decoupling of control plane from data plane, offers a greater control of a network through programming. This combined feature would bring potential benefits of enhanced configuration, improved performance, and encouraged innovation in network architecture and operations, as summarized in Table I. The control embraced by SDN may include packet forwarding at a

switching level, link tuning at a data link level, breaking the barrier of layering. Moreover, with an ability to acquire instantaneous network status, SDN permits a real-time centralized control of a network based on both instantaneous network status and user defined policies. This further leads to benefits in optimizing network configurations and improving network performance. The potential benefit of SDN is further evidenced by the fact that SDN offers a convenient platform for experimentations of new techniques and encourages new network designs, attributed to its network programmability and the ability to define isolated virtual networks via the control plane. In the subsection, we dwell on these afore mentioned benefits of SDN.

A. Enhancing Configuration:

In network management, configuration is one of the most important functions. Specifically, when new equipment is added into an existing network, proper configurations are required to achieve coherent network operation as a whole. However, owing to the heterogeneity among network device manufacturers and configuration interfaces, current network configuration typically involves a certain level of manual processing. This manual configuration procedure is tedious and error prone. At the same time, significant effort is also required to troubleshoot a network with configuration errors. It is generally accepted that, with the current network design, automatic and dynamic reconfiguration of a network remains a big challenge. SDN will help to remedy such a situation in network management. In SDN, unification of the control plane over all kinds of network devices, including switches, routers, Network Address Translators (NATs), firewalls, and load balancers, renders it possible to configure network devices from a single point, automatically via software controlling. As such, an entire network can be programmatically configured and dynamically optimized based on network status.

B. Improving Performance:

In network operations, one of the key objectives is to maximize utilization of the invested network infrastructure. However, owing to coexistence of various technologies and stakeholders in a single network, optimizing performance of the network as a whole has been considered difficult. Current approaches often focus on optimizing performance of a subset of networks or the quality of user experience for some network services. Obviously, these approaches, based on local information without cross-layer consideration, could lead to suboptimal performance, if not conflicting network operations. SDN offers an opportunity to improve network performance globally. Specifically, SDN allows for a centralized control with a global network view and a feedback control with information exchanged between different layers in the network architecture. In a properly designed centralized algorithm many challenging performance optimization problems would become manageable. It follows that new solutions for classical problems, such as data traffic scheduling, end-to-end congestion control, load balanced packet routing, energy efficient operation, and Quality of Service (QOS) support, can be developed and easily deployed to verify their effectiveness in improving network performance.

C. Encouraging Innovation:

In the presence of continuing evolution of network applications, future network should encourage innovation rather than attempt to precisely predict and perfectly meet requirements of future applications. Unfortunately, any new idea or design immediately faces challenges in implementation, experimentation, and deployment into existing networks. The main hurdle arises from widely used proprietary hardware in conventional network components, preventing modification for experimentation. In comparison from table 2.1, SDN encourages innovation by providing a programmable network platform to implement, experiment, and deploy new ideas, new applications, and new revenue earning services conveniently and flexibly. High configurability of SDN offers clear separation among virtual networks permitting experimentation on a real environment. Progressive deployment of new ideas can be performed through a seamless transition from an experimental phase to an operational phase.

Table 2.1 comparison between SDN and conventional networks

CHARACTERISTICS	SDN	CONVETIONAL NETWORKING
Features	Decoupled data and control plane and programmability	A new protocol per problem, complex network control
Configuration	Automated configuration with centralized validation	Error prone manual configuration
Performance	Dynamic global control with cross layer information	Limited information and relatively static configuration
Innovation	Easy software implementation for new ideas, sufficient test environment with isolation, and quick deployment using software upgrade.	Difficult hardware implementation for new ideas, limited testing environment, long standardization process.

D. SDN Challenges

Given the promises of enhanced configuration, improved performance, and encouraged innovation, SDN is still in its infancy. Many fundamental issues still remain not fully solved, among which security is the most urgent ones. An open-source Open Flow driver is still absent for SDN controller development, a standard north-bound API or a high level programming language is still missing for SDN application development. A healthy ecosystem combining network device vendors, SDN application developers, and network device consumers, has yet to appear. SDN offers a platform for innovative networking techniques. However the shift from traditional networking to SDN can be disruptive and painful. Common concerns include SDN interoperability with legacy network devices, performance and privacy concerns of centralized control, and lack of experts for technical support. Existing deployments of SDN are often limited to small test bed for research prototypes.

V. LITERATURE REVIEW ON SDN

A literature review essentially examines relevant literature for a specific field of study. It creates a stable basis by examining what is already known about a chosen topic. As a result, a literature review opens new approaches for further studies and progresses in the concerning field of research. The review has its main goal in identifying the used methods and concepts for effects and challenges in SDN development.

Network technologies have always been a crucial part of success for technologies like cloud computing. But due to the slow development of a scalable IT infrastructure, this can lead to issues in competitiveness. Software defined networking (SDN) can thereby counteract such issues by giving new functions to the whole network topology. With SDN, administrators have the possibility to abstract the underlying network infrastructure for applications and network services. This chapter reports on the main outcomes of a systematic literature review on challenges and effects of SDN. It shows that most papers address the implementation of software defined networking as a challenge, like the general risk of changing traditional network architectures. Attention is also given to security issues arising with software defined networks and the permanent high demand from the end-user combined with the fear of changing traditional networks. Issues dealing with specialized know-how were identified as another challenge category. Effects of SDN are discussed by defining unique features of SDN like decoupling hardware from the software and the global view of the whole network architecture. SDN furthermore affects the management of the network, including changes in deployment of policies, the programmability and maintenance of the network.

A. Overview Of The Literature Review:

The main results of the literature review are given in brief .Figure 3 shows an overview of the identified challenges and effects of SDN. Most papers address the implementation as a challenge. Factors, like the high risk of changing traditional network architectures, are included in this category, and discussed and researched most often. The highest in terms of attention given is the category of security issues which Included in this category are demand arising with software defined networking and the permanent high demand from the end-user combined with the fear of changing traditional networks. The other categories are implementation and know-how existing for software defined networking. Administrating and controlling software defined networks with the existing staff and the overload arising from this were subsumed in this category.

Figure 3 shows the overview of the literature survey. Elements, like decoupling hardware from the software and the global view of the whole network architecture, are reviewed. Easier deployment of policies, the programmability and maintenance of the network are included. These trends show that current research is more focused on technical and scientific consideration Software defined networking is seen as an evolutionary paradigm shift, but still faces several challenges. The covered areas of challenges and effects provide a general view of what may slow down further development and what is possible when the technology is integrated successfully

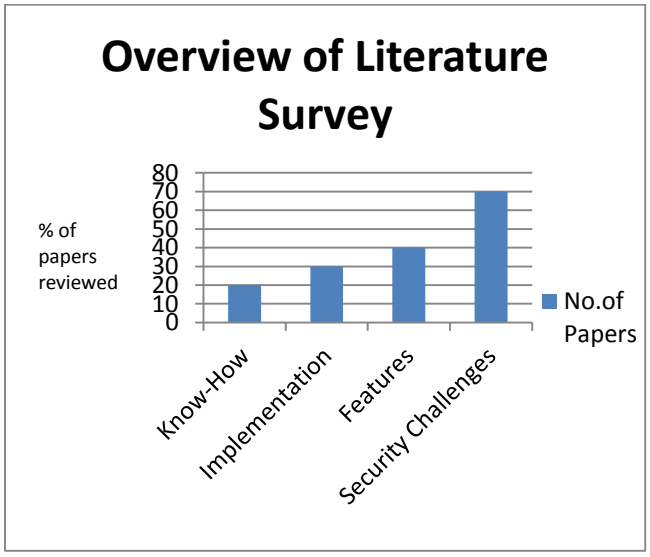

Fig. 3. Overview of Literature Survey

Certain lacks of know-how, combined with high complexity when it comes to integration into traditional networks, are main reasons for a delayed diffusion of the technology. Furthermore, the analyzed papers mostly describe software defined networking on a very detailed and technological basis, making it very hard for enterprises and organizations to assess if the technology can have a specific business impact (e.g. on increasing efficiency or reducing costs). Nevertheless, the steady increase of users and requirements leave providers faced with the need to rethink the usage of current network technologies in order to stay competitive and profitable. As the literature review has revealed, the separation of the control and data plane

offers great benefits, such as easier management, enhanced features, as well as economic factors. Besides outlining the technical aspects these benefits should play an important role in further research, especially in the security aspects.

VI. SECURITY ANALYSES OF SDN

The basic properties of a secure communications network are: confidentiality, integrity, availability of information, authentication and non-repudiation. In order to provide a network protected from malicious attack or unintentional damage, security professionals must secure the data, the network assets (e.g. devices) and the communication transactions across the network. The alterations to the network architecture introduced by SDN must be assessed to ensure that network security is sustained. Systems are becoming increasingly complex. This complexity in turn has led to increased risk and severity of bugs and errors in implementations. This complexity increases with virtualization within networks, as increasing numbers of traditional hardware functions are realized by software. This puts a large amount of pressure upon programmers to deliver flawless software solutions. Additionally, this pressure is increasing due to the business trend towards cloud computing.

A. Security Challenges with SDN

While security as an advantage of the SDN framework has been recognized, solutions to tackle the challenges of securing the SDN network are fewer in number. SDNs provide us with the ability to easily program the network and to allow for the creation of dynamic flow policies. It is, in fact, this advantage that may also lead to security vulnerabilities. Within this dynamic environment, it is vital that network security policy is enforced. The promises of agility, simplified control, and real-time programmability offered by software-defined networking (SDN) are attractive incentives for operators to keep network evolution apace with advances in virtualization technologies. But the capabilities that undermine the security is a question to be answered. The aim is to ensure that networks are protected from attack by malicious intruders.

B. Key Security Assets:

Networks that are built according to SDN architecture principles need to protect a number of key security assets:

- Availability – the network should remain operational even under attack.
- Performance – the network should be able to guarantee a baseline bandwidth and latency in the event of an attack
- Integrity and confidentiality – control plane and data plane integrity and isolation should be upheld between tenants.

VII. PROCESS NECESSARY FOR SDN SECURITY:

To assure protection of these assets, a number of processes need to be in place. They are authentication and authorization, resiliency, multi-domain isolation and repudiation.

A) Authentication and Authorization

Only authenticated and authorized actors should be able to access SDN components. The granularity of authentication and authorization must be detailed enough to limit the consequences of stolen credentials or identity hijacking.

B) Resiliency

Networks must be able to recover as autonomously as possible from an attack, or a software or hardware failure. Alternatively, networks must be able to dynamically work around any affected functionality.

C) Multi-Domain Isolation

Systems must be able to isolate tenants in multiple domains, such as the resource and traffic domains. The following forms of isolation apply:

- Resource Isolation – prevents tenants from stealing resources, like bandwidth, from each other.
- Traffic Isolation – required by multi-tenant deployments, so a tenant can see its own traffic only (this requirement applies to both data plane and control plane traffic).

D) Repudiation

All actions carried out by all system actors both internal and external must be logged, and the all logs need to be secured. Systems should provide visibility into operations and network status so they can determine the most appropriate action when issues arise. An active approach to security requires correct identification and classification of an issue so the most appropriate action to mitigate it may be chosen. Any action should be verified to ensure that it has been enforced effectively.

E) Resilient Control Plane

The three main elements of SDN are: SDN apps, the SDNC, and NES. Given that control of the network is centralized, all communication within the control plane needs to be treated as critical, as an outage resulting from a successful attack may lead to an undesired impact on business continuity. If, for example, the SDNC is prevented from taking critical action to mitigate a dos attack, the entire network and all of its tenants may be affected. To avoid this, the control plane needs a greater level of resiliency built into it. To communicate with tenant applications and NES, the SDNC exposes a set of interfaces. All these interfaces may experience heavy traffic loads, depending on the type and number of running applications. Traffic on the interfaces can be further impacted by NES, for example, forwarding packets for which they have no forwarding rules. So, in terms of dependence on the SDNC, traditional networks appear to be more robust. An effective way to improve the resilience of the centralized control plane and prevent the spread of DDOS control-plane attacks to the rest of the network is to limit NES in terms of bandwidth and resource consumption

such as CPU load, memory usage, and API. Resilience can be further enhanced through proper resource dedication where the SDNC authenticates each resource request, and subsequently checks requests against strong authorization control policies.

F) Strong Authentication and Authorization

Authentication and authorization are the processes used to identify an unknown source and then determine its access privileges. Implemented correctly, these processes can protect networks from certain types of attack, such as:

- Provision of false (statistical) feedback to the system – for example, fooling the system into believing it is under attack, resulting in unnecessary deployment of countermeasures, which consumes resources and inevitably leads to suboptimal usage.
- Modification of a valid on-path request – which results in a direct attack that alters network behavior.
- Forwarding Traffic that is not meant to be forwarded or not forwarding traffic that should be – subverting network isolation.
- Gaining control access to any component – rendering the entire network untrustworthy.

The critical nature of the SDNC dictates that additional security measures need to be taken to protect it. At the very least, traffic must be integrity protected to prevent tampering of on-path traffic, but even this level of protection does not secure control data. Encryption is one way of preventing control data from being leaked. But, even together with integrity protection, encryption is not sufficient to protect against man-in-the-middle-type attacks. And so, all communication within the control plane must be mutually authenticated. Security protocols like TLS and IPSEC provide a means for mutual authentication as well as for replay attack protection, confidentiality, and integrity protection. Mutual authentication does, however, present some difficulties, such as how to bootstrap security into the system. One way to solve this is by using security certificates. How then these certificates are issued, installed, stored, and revoked then becomes the significant security difficulty. Encryption and integrity protection without mutual authentication are less useful from a security point of view. The problem with mutual authentication is that it requires previous knowledge of the remote communicating endpoint – unless a commonly trusted third party exists.

G) Network Security Challenges

IT infrastructure is rapidly moving to the cloud, creating a dramatic technology shift in the data center. This shift has significantly influenced user behavior: end users now expect anytime, anywhere access to all their data. Additionally, network operations are being transformed from operator-intensive management towards greater automation. The data center of the future is emerging as a highly virtualized environment that must address a diverse set of user needs, including anytime, anywhere access to their data, the consumerization of IT and increased reliance on cloud services. Security concerns are consistently identified as a major barrier to this data center transformation. While protecting user data is of paramount importance, mobility and virtualization pose new threats that must be understood and secured. And the human factor continues to lead to unnecessary downtime, expense, and unauthorized intrusion. Throughout the enterprise, end devices and data center resources including hypervisors, storage devices, servers, switches, and routers must be secured. Despite the diverse threats, existing security strategies can be successful at minimizing many of the security risks in the data center. Currently available security solutions are, however, difficult to deploy, manage, program, scale, and secure. Policies are tightly coupled to physical resources as opposed to services and applications. Security solutions struggle to provide quick and automated threat mitigation across equipment from multiple vendors. Consistent security policies are difficult to administer across compute, storage, and network domains and multiple data centers. No solutions today allow for complete security orchestration across data center networks.

H) DenialoOf-Service-Attacks

Computer networking has come a long way, but even with today's advanced network architecture, there are vulnerabilities. DOS attacks are one of the most common security-related problems of servers today. A DOS attack can be accomplished by several methods, but most of these attacks can be categorized into one of three different methods: vulnerability attacks, connection flooding, and bandwidth flooding.

I) Vulnerability Attacks

Vulnerability attacks take advantage of bugs or exploits in the service at the server. In this way, the service stops functioning and in the worst case, the server hosting the service could crash.

J) Connection Flooding

Connection flooding also called TCP SYN flood attacks occurs when a large number of TCP connection attempts arrive at the targeted server. The attacker causes these TCP SYN packets to be sent, either by one source or by many sources*. When a TCP connection is being created, the client and server exchange messages to establish a TCP connection before they send any data. The first packet sent by the client has the SYN (synchronization) flag set and an initial sequence number. The server allocates a TCP control block and sends a SYN-ACK (synchronization acknowledgement) back to the client along with the server's SYN flag sent to indicate that it is sending its own initial sequence number. The client would normally send an ACK (acknowledgement) back to server thus establishing the TCP connection. If the last step of the procedure does not occur, there is a half-open TCP connection. At some point the server will not be able to establish anymore connections until the half open TCP connections are closed (thus releasing the storage associated with their TCP control blocks), therefore all new

legitimate connection establishment attempts will be denied.

K) Bandwidth Flooding

Occurs when a large number of packets are sent (nearly) simultaneously by the attacker (or by hosts controlled by the attacker) to the targeted host. The target's incoming link will be choked (i.e., all of the available bandwidth will be used up) and legitimate usage of the server becomes constrained. In some cases, one attack machine is insufficient to cause sufficient damage. For example, such a bandwidth flooding DOS attack would fail when the targeted server has an access bandwidth much greater than the amount of traffic coming from the attacker. In this case, a DDOS attack would be used by the attacker. In a DDOS attack the attacker creates a network, often referred to as a botnet, by infecting multiple computers with viruses or Trojans. These infected computers are often called zombie computers. The attacker can now have a much larger impact on the targeted server because it can coordinate multiple zombies to generate traffic at a much higher aggregate rate. Due to the above mentioned security issues there is an urging need to focus on the security issues of SDN as SDN is the current emerging technology.

VIII. SOLUTIONS FOR INCREASING SECURITY WITH SOFTWARE-DEFINED NETWORKING

In the legacy Ethernet network, the control and data planes go tight together, making it easier for the attacker to find ways to attack these networks. In SDN, the control plane is decoupled from the data plane, isolating the data flow from the control. This makes it harder to target with several legacy network attacks. Nevertheless, as SDN can tackle most of the security issues, it brings new issues to discover and to deal with. In each section there will be a brief description of the threat and the issue in network, which is followed by a possible solution to overcome it Using SDN. To perform any attack to any network, the attacker has to gain access to it either by physically or by controlling existing resources. After the access, the attacker can perform several different types of attacks. An attacker can have five different motives to the attack: (1) learning about the private network topology and the network traffic for use in a later attack, (2) gaining control over switches, routers, or servers in the LAN, (3) eavesdropping, (4) manipulating information, or (5) disrupting the availability of the network.

A) Unauthorized Joins and Expansion of the Network.

The basic physical access to a network is a threat if it has been done by an unauthorized person. Connecting to an Ethernet segment in a plain legacy network is easy since the purpose of Ethernet was to provide as little as possible administration overhead. This means that if an unauthorized person access a switch, an existing host or a wall socket, which are connected to the target segment, an unauthorized join is possible. From the design of Ethernet network, expanding the network is possible by connecting a switch to another switch. Therefore, if an attacker gains access to the network, the attacker is permitted to expand the network for providing multiple access points (wired or wireless) to the target network. Initially, the issue can be solved by limiting the access points by keeping switches and other network devices in secure and secret locations, providing the existing hosts and switches authorized connections and by maintaining the unused wall sockets to be disabled for connections. Nevertheless, the maintenance of such a setting requires multiple configurations in several different devices. This demands costly professional personnel and a considerable work effort. In SDN, the solution would be similar, but it would require less effort. Since the controller is centralized, switches' ports can be controlled off and on from a single point, and the connectivity status to all hosts can be followed. Furthermore, the authorization of hosts can be provided by an application running over the controller.

B) Remote Access to the LAN

By compromising a host inside the target segment, the attacker can use the compromised host to access administrative services and commence the attack further. In this case, the vulnerability is created with social engineering (i.e. Phishing). To prevent them, the utilization of security services has to be conducted, such as firewalls,

Spam filters, anti-virus, and anti-spy-ware software. Although, these services are not airtight and can be bypassed as new vulnerabilities are discovered. Another way is to educate users to be more cautious when using the Internet. It is worth to mention that, this issue is out of Ethernet's scope. As initially, the host has to have an access to communicate with the network. In principle, SDN cannot provide more protection against social engineering, but then, when a host is compromised, administrative access is not possible from a host (unless access is separately granted, which is not the norm). The main target of SDN is to isolate the control from the data flows, giving its host a transparent network.

C) Topology and Vulnerability Discovery.

An attacker can send to a network messages to investigate the responses and to discover the topology and services provided by the hosts. From simply listening to broadcast requests, an attacker can find out used host IP addresses, servers, and gateways where the hosts are connected to. The IP range used can be also investigated from the replies. With these bits of information, an attacker may scan throughout the network and reveal vulnerable hosts, enabling the attacker to access and do further attacks. Obviously, this threat presents two issues: broadcast messages and network scanning. In normal networks, solutions are not directly provided to solve these issues individually, and the solutions are higher layer oriented. Against network scanning and intrusion there are solutions to prevent them such as firewalls, Intrusion Detection Systems (IDS), and Intrusion Prevention Systems (IPS). These mechanisms are designated to detect and stop intrusions and unauthorized traffic. A solution to remove broadcast messages from a network can be done with SDN .As it turns out, with SDN, it possible to remove broadcast

messages from a network without affecting the performance or functionality. For network scanning, SDN presents again a way to overcome it. When an attacker has the information of vulnerability in a host, an attack can be executed using Ethernet as a medium. Usually, this appears in the target as a normal network session, thus the attack remains undetected. Generally, with SDN, host break-ins are also hard to detect. If an attacker has found a host and its vulnerabilities, then it is likely that the flow created for the attack was successful (i.e. the attack comes from a trusted source).

D) Switch Control

When switches are newly installed, as default they are provided with a default password (or none) for accessing the administrative tools and services. If the network administrator was careless and did not change the default password, the attacker gained the physical access to the switch and reset the password with physical reset, attacker gained the password by stealing, or found vulnerability with the switch, and an attacker can access and control the switch. The attacker could reroute the traffic by shutting down links, or by performing a Denial of Service attack to selected links. Anyway, the attacker can make a switch redirect or mirror its traffic to the attacker, which makes eavesdropping possible and also unauthorized access.

A switch is preferred to be placed in a secure and secret location. This way an attacker cannot perform a physical password reset, thus making switch control attack harder. The administrative tools and services should always be kept behind passwords, and preferably if it is possible, behind authentication services with certificates such as TLS, SSL, or anything similar. This way the intrusion into a switch is limited and restricted, making an attacker harder to gain the control of the network. In SDN, a switch is tightly connected with its controller. Without the controller, the switch is dumb and thus cannot act on any network. Nevertheless, a SDN supporting switch is protected with password in the same manner as the legacy switches.
The hardware of a SDN switch is more powerful than in a legacy switch, since handling flow generation and matching packets requires more computation power. This means that a SDN switch has more abilities to tamper traffic and get control of it. Assuming that an attacker gained the control of an SDN switch, he could look up the generated flow rules from the flow rule tables and save it for later usage and then change the switch's controller to the attackers' own controller, which shall keep the same flow rules available and mirror every traffic to the attacker for eavesdropping. Obviously, the complete control of the network (switches) belongs to the attacker. In SDN, it is recommended to isolate physically the control network from the managed data network. Else, in the case of logical control traffic isolation, the attacker could eavesdrop switches' control traffic by passively listening to inter switch links and in addition try to connect to the administration interface. As for unauthorized administration access, a possible solution could be a mechanism where the switch notifies the controller of an access attempt providing the controller to act upon it.

An additional thereat to the survey has to be discussed. As SDN differs not only of its programmability, but also in its physical and design structure. The centre node of a SDN network is the controller, which brings one more threat to the network and system access topic: Controller Control. The SDN controller is not currently standardized and existing implementations vary in many levels. More importantly, the controller is a network device, practically a computer, which possesses multiple Network Interface Controllers (NIC) with multiple ports for efficient switch connectivity. In case an attacker compromises the controller, the ultimate control of the whole network managed by that controller is accessible. As critical as it sounds, this calls for drastic measurement to protect this central orchestration unit. Since the controller is a computer, it is highly recommended that security levels of accessing this device are with the highest priority. This is currently the weakest point of SDN.

E) Traffic Confidentiality

Since confidential messages and information are transmitted in the network, one of the biggest interests of an attacker is to listen to the traffic (eavesdropping). In a network, a switch sends to a host its own traffic, broadcast messages, and frames which are flooded at the time of the switch's table timeout .In addition, a switch floods frames in case of a unknown destination. This means, an attacker can generate frames with random destinations to overwrite table and let the switch flood all frames, thus enabling eavesdropping.

In SDN, broadcast messages can be removed and random flooding does not happen except if the controller has applied such a mechanism in the switch. By default, in SDN, unknown traffic is directed to the controller for further processing. In the case of the controller not knowing the recipient, the controller can flood out a broadcast message to determine the location of the unknown host. On the contrary, if the controller is implemented in the manner to keep track of all its connected hosts, this situation never comes up. An efficient way to eavesdrop is to perform it passively. Passive eavesdropping is performed between two host, two switches, or between a host and a switch. The attacker attaches a listening device in the medium (i.e. cable) and collects everything that goes through. Since there is no architectural change in the network, passive eavesdropping is fairly hard to discover. A slight advantage in SDN is that it can prevent a switches flooding any traffic and hence bypass passive eavesdropping.

F) Traffic Integrity

The integrity of the traffic is what the users rely on for genuine networking experience.

An attacker, after gaining the control of traffic, can modify and imitate reliable services and acquire confidential information from the host or the user. This is a threat both in SDN networks and in traditional networks.

G) Man in the Middle (MITM).

An attacker can direct the traffic intended between the host through the attacker, or imitate a server to create a scenario where the attacker is a proxy (hidden from the host) between the host and the real network. MITM attacks are mainly to perform for eavesdropping or to for tampering and modifying traffic. Man in Middle attack, where the attacker captures and divides the network into two splits where the traffic flow through. In SDN, the controller has the topology knowledge and the controller can decide the best route for each individual flow according to desired requirements. This leaves SDN out of scope for MITM attacks. Nevertheless, MITM attacks with passive eavesdropping are still possible in SDN networks.

H) Session Hijacking and Replay.

Many higher layer protocols create sessions to handle their communication services. If an attacker has a possibility to do a MITM attack and eavesdrop the traffic, the attacker can collect information about the sessions and thus recreate it or hijack it. The attacker can then pretend to be one of the endpoints and use the service disguised as the other endpoint. To quiet down the pretended endpoint, the attacker can simply execute a Denial of Service attack or divert the traffic. A session hijacking attack can be used to break in to a service which requires initial authentication but no further verification. An authenticated session can be recreated if an attacker gained packets which are used for initial authentication. This is called the Replay attack. Even though the packets might be encrypted, the attack does not require more than the knowledge of the content, and the attacker can reuse it without altering it. To prevent session hijacking and replay attacks, higher layer protection services have to be used. In SDN, these attacks cannot be directly addressed. As the session is created, it is assumed trustful from the lower layer connection, thus leaving it to higher layers to take care of the authenticity. Although a SDN application could be created to follow up sessions, triggered and terminated by the host. This solution would require host to controller connectivity. Generally, as in this threat it is assumed that MITM attack has succeeded, SDN has evenly a little role to stop the attack .

I) Denial of Service

For disrupting traffic or networking performance, an attacker can perform a Denial of Service (DoS) attack. It can be performed either to suspend or interfere with a service. In SDN a resource exhaustion attack can be controlled with decreasing traffic on a certain node or blocking misbehaving port completely. Since the management and control are decoupled in SDN the computation is outsourced and switches have only behaviour instructions regarding the traffic. Nevertheless, a switch can be overloaded if the switch is set to process certain frames before forwarding them. Limiting the traffic (by dropping packets or closing the port), or by redirecting the traffic through other path, resource exhaustion attack can be controlled. As explained in MITM attack an attacker can add a fake switch to the network and disrupt the whole network by flooding useless traffic.SDN falls out of scope for this threat. Furthermore, the concept of SDN leaves the network functionalities completely to the controller and its applications. Thus, SDN networks can be built completely from the scratch, discarding all existing protocols. The network functionalities can be designed specifically to satisfy a specific type of network (LAN). In this sense, DoS attacks which are performed by misusing a specific protocol's vulnerability, can be completely disregard in SDN networks. It does not mean that SDN prevents DoS attacks, but it makes them certainly different.

J) System security

Building complex networks requires careful design of configurations. For every network device, precautions have to be taken against security threats while preserving network functionalities. To enable certain functionalities, the right configurations and hardware installations are in a network device essential. Configuration and Installation Issues may arise, as humans occasionally tend to make mistakes. An attacker can misuse or exploit network vulnerabilities in case of gaining the information about a configuration or installation mistakes. In traditional network, every switch needs a separate installation and configuration procedures thus, making it essential to carefully prefabricate configurations. On the contrary, SDN requires only careful hardware installation and a simple configuration (e.g. locating controller). The rest configurations are done by the controller, hence, configuration mistakes are simpler to repair.

IX. CONCLUSION

This paper reviews the current research into SDN security along with a prediction for the paths that future research will take. This research marks the attempts at survey of SDN security, which until recently has been somewhat lacking. However, the benefits provided by SDN also introduce new security challenges.

SDN can bring many advantages to improve security. Due to the entity of a central element in the network, the controller, SDN gives an extended control over the network. Hosts connected to network can be directly recognized and managed. Network traffic can be categorized by any parameter and routed precisely through any route, and new features can be implemented with proper authentication mechanisms. Ultimately, the network can be shaped according to the demand. SDN has the ability to control the network, removes the need of many protocols along with their vulnerabilities.

The existing SDN security implementation proves that the researching community has recognized the demand for secure SDN solutions and hence provided with a framework or a system checking solutions. There are also other implementations which were not introduced in this survey, yet they have a similar purpose. Research discussed here represents valid solutions to some of these security issues, but further work must be done in order to satisfactorily secure these vulnerabilities.

X. REFERENCES

[1] Wenfeng Xia, Yonggang Wen, Heng Foh, Dusit Niyato, and Haiyong Xie, "A Survey on Software-Defined Networking", *IEEE COMMUNICATION SURVEYS ,2015.*

[2] Michael Coughlin University of Colorado Boulder, "A Survey of SDN Security Research",2012

[3] Bhattacharya B, Das D,"SDN based Architecture for QOS Enabled Services across Networks with Dynamic Service Level Agreement", *IEEE ANTS, 2013.*

[4] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Communication, 2013*.

[5] S. Agarwal, M. Kodialam, and T. V. Lakshman, "Traffic engineering in software defined networks," *IEEE, 2013.*

[6] C. S. Li and W. Liao, "Software defined networks," *IEEE Communication, 2013.*

[7] S. Ortiz, "Software-defined networking: On the verge of a breakthrough?" Computer, July 2013.

[8] D. Kreutz, F. M. Ramos, and PVerissimo,"Towards secure and dependable software-defined networks", Proceedings of the second ACM Hot topics in software defined networking - *HotSDN , 2013.*

[9] K. Bakshi, "Considerations for software defined networking (SDN):Approaches and use cases," ,*IEEE Conference, March 2013.*

[10] S. A. Mehdi, J. Khalid, and S. A. Khayam. "Revisiting Traffic Anomaly Detection using Software Defined Networking", Recent Advances in Intrusion Detection 2013.

[11] Raghavan B, Koponen T, Ghodsi A, "Software defined networking: A new paradigm for virtual, dynamic, flexible networking," *IEEE 2014.*

[12] Casado M, Ratnasamy S, Shenker S,"Software-defined internet architecture: decoupling architecture from infrastructure",*ACM Hotnets '12*,2012

[13] Kobayashi M, Seetharaman S, Parulkar G, Appenzeller G, Little J, van Reijendam J, Weissmann P, McKeown N,"Maturing of Open Flow and Software-defined Networking through deployments",*IEEE Conference on Computer Networks*,2014

[14] Galis A, Clayman S, Mamatas L, Rubio Loyola J, Manzalini A, Kuklinski S, Serrat J, Zahariadis T,"Softwarization of Future Networks and Services - Programmable Enabled Networks as Next Generation Software Defined Networks", *IEEE SDN for Future Networks and Services(SDN4FNS)*,2013.

[15] Caraguay ALV, Lopez LIB, Villalba LJG,"Evolution and Challenges of Software Defined Networking", *IEEE Communications Magazine*,2012.

[16] Cahn A, Hoyos J, Hulse M, Keller E,"Software-Defined Energy Communication Networks: From Substation Automation to Future Smart Grids",*IEEE Smart Grid Communication 2013 Symposium - Smart Grid Services and Management Models*,2013

[17] Wang J, Wang Y, Hu H, Sun Q, Shi H, Zeng L,"Towards a Security-Enhanced Firewall Application for OpenFlow Networks",*Springer LNCS CSS*,2013.

[18] Dixit A, Hao F, Mukherjee S, Lakshman TV, Kompella R,"Towards an elastic distributed SDN controller".*IEEE international conference on HotSDN* ,2013.

[19] Gupta M, Sommers J, Barford P,"Fast, accurate simulation for SDN prototyping",*Springer*,2013

[20] Jarraya Y, Madi T, Debbabi M,"A Survey and a Layered Taxonomy of Software-Defined Networking",*IEEE Communication Tutorials* ,2014.