Computer Communications 183 (2022) 64–82

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Towards security automation in Software Defined Networks

Noe M. Yungaicela-Naula [a,*], Cesar Vargas-Rosales [a], Jesús Arturo Pérez-Díaz [a], Mahdi Zareei [a]

[a] Tecnologico de Monterrey, School of Engineering and Sciences, Monterrey, NL 64849 MX, Mexico

ARTICLE INFO

Keywords:
Software defined networks
Security automation
Network function virtualization
Machine learning
Deep learning
Reinforcement learning

ABSTRACT

Software-Defined Networking (SDN) is a modern paradigm that provides a platform for implementing reliable, centrally managed, and automated security solutions for conventional and new generation networks, such as IoT, cloud computing, 5G/6G mobile communication networks, and vehicular communications. In these complex systems, manual security operations can delay or obstruct the identification, mitigation, and prevention of ever-increasing sophisticated threats. Thus, the idea of security automation for networks using the SDN paradigm has become fundamental, given that SDN was created to facilitate the operation and management of complex networks with minimal human intervention, which is considered error-prone. This survey studies the state-of-the-art research efforts concerned with security automation in SDN environments. We identified and ranked various classes of security solutions with different levels of automation and complexity. The level of automation is measured using four well-defined qualitative parameters: self-healing, self-adaptation, self-configuration, and self-optimization. The complexity is characterized by the amount of processing and storage resources and implementation requirements. This work represents the first endeavor to analyze the level of automation and complexity of security solutions in SDN environments. Our findings reveal important advances in the area of security automation in SDN. However, there are still several open problems and challenges, which we detail in this work.

Contents

* Corresponding author.
E-mail address: a00821711@itesm.mx (N.M. Yungaicela-Naula).

https://doi.org/10.1016/j.comcom.2021.11.014
Received 19 March 2021; Received in revised form 23 September 2021; Accepted 14 November 2021
Available online 27 November 2021

1. Introduction

Security demands in new complex and dynamic networks, such as IoT [1–4], cloud computing [5], 5G/6G communication networks [6,7], and vehicular communications, [8,9], in addition to the increasing number of sophisticated and intelligent attacks, are key motivating factors for deploying automated defense solutions for networks using the Software-Defined Networking (SDN) technology. On the one hand, key networking trends place an increasing burden on systems and network administrators. These trends include an increase in network traffic volume, the virtualization of servers, storage, and networking devices, the growth in size and complexity of data centers, and the growth of cloud computing and IoT applications. In this context, the Information Technology (IT) workforce becomes a significant security bottleneck. This workforce cannot manage the vastly increasing number of incidents and alerts and the need to deploy optimized security controls in response. Manual procedures result in slower detection and remediation of issues, mistakes in resource configuration, and erratic security policy implementation, leaving systems vulnerable to compliance issues and threats.

On top of that, advanced security skills are in higher demand than ever before; however, there is a significant lack of qualified and skilled security professionals. An estimated 3.5 million cybersecurity jobs will be available but unfilled by 2021 [10]. In addition, existing cybersecurity staff is overworked and not set up for 24x7x365 incoming threats [11]. According to the CISCO Cybersecurity Report Series 2020 [12], from a survey of 2800 organizations around 13 countries, 40% of these were receiving 5000 daily alerts. This overwhelming number of alerts is having an impact on security administrators' fatigue. As a result, according to a report presented by the Enterprise Strategy Group [13], IT teams ignore about 74% of security threat alerts, even when they have security solutions in place, due to the sheer volume.

On the other hand, agile and sophisticated threats are rising, such as the Advanced Persistent Threats (APTs) [14]. Different from common cyber-attacks, APTs focus on large organizations and industry sectors, causing severe damage, such as failure of essential services and destruction of critical infrastructure. These attacks usually become undetected, and the damage caused can be critical [15]. Among the most frequent, sophisticated, and dangerous APTs are DDoS attacks. According to a Kaspersky report [16], the number of DDoS attacks in 2020 almost doubled the number of DDoS attacks of 2019. APTs can cause huge capital and reputational losses to the victims. Indeed, the World Economic Forum detected that cyberattacks are perceived as the #2 global risk of concern to business leaders in advanced economies [17].

Including automation in the heart of network security management will help to deal with the increasing number of alerts and skilled IT administrators' shortages. Security automation is the machine-based execution of security actions to programmatically detect, investigate, mitigate, and prevent threats timely and efficiently with minimal human intervention. With this automation, repetitive and time-consuming activities are taken out of the hands of the IT workforce so they can focus on more relevant value-adding work.

Security automation can reduce the average cost of a breach by 95% [18]. Therefore, most organizations plan to increase automation to simplify and speed up response times in their security ecosystems. According to Gartner, security automation is one of the top security and risk management trends [19]. The execution of automated security tasks is much faster than human interactions and generates fewer errors.

SDN is enabling security managers to bridge the response resource gap through intelligent incident detection and automated response. SDN decouples the control plane from the network devices and logically centralizes the administration of network resources. This networking technology presents new security challenges for network designers and operators, but it also provides a platform for implementing reliable, centrally managed, and automated security solutions for networks.

Although many surveys have already studied intelligent methods applied in SDN-based security solutions, none have comprehensively analyzed and compared the complexity and level of automation achieved with these techniques. The idea of security automation in SDN environments is fundamental since this technology was created to facilitate network operation and management with minimal human intervention, which is considered error-prone.

Network automation has been studied from different perspectives and defined using terms such as cognitive, self-managing, or self-aware networks [20–22]. The common characteristics of these definitions are the ability to think, learn, and remember. Therefore, an automated network can adapt in response to conditions or events based on the reasoning and prior knowledge it has obtained. Moreover, an automated network can be characterized by its ability to complete its tasks autonomously by using its self-properties, namely, self-healing, self-adaptation, self-configuration, and self-optimization, to adapt dynamically to changing requirements while considering the end-to-end goals [22].

In this paper, we survey existing research efforts related to the automation of security in SDN-based networks, using peer-reviewed studies published in the last seven years (2015–2021).

The security solutions reviewed are divided into two main classes, namely reactive and proactive solutions. On one hand, reactive security solutions (also known as detection/mitigation solutions) are based on different Artificial Intelligence (AI) enabled techniques such as Machine Learning (ML), Deep Learning (DL), and Reinforcement Learning (RL) with online or offline model learning. The methods of AI have been used to discover threats in networks [23–25] since they are considered as robust potential solutions to make networks self-aware, self-adaptive, self-secured, self-healing, and self-managed [26,27]. The study of SDN-based reactive security solutions requires an analysis of the characteristics that makes detection and mitigation mechanisms adaptive to changes in the network. On the other hand, proactive security solutions use emerging concepts such as Network Function Virtualization, Moving Target Defense, Cyber Deception, Network Slicing,

and Blockchain to secure the SDN architecture effectively. Proactive defense methods focus on evade/prevent attacks or on reducing the attack surfaces in the network. We evaluate the level of automation, of reactive and proactive solutions, in terms of their complexity level with respect to their implementation, and their self-* properties (self-healing, self-adaptation, self-configuration, and self-optimization).

The contributions of this work are:

1. We introduce a new taxonomy of security solutions for SDN-based networks. Different classes of security solutions are identified and studied, with emphasis on their current state of development.
2. We provide a comprehensive analysis, identification, and comparison of the complexity (processing, storage, and implementation) and the level of automation (self* capabilities) of each class. Using these parameters, we present a ranking of the classes pursuing security automation. This analysis provides a comprehensive and updated landscape of how the SDN paradigm is empowering the creation of innovative and automated defense solutions for networks and also helps researchers identify possible areas of contribution in this imperative field of study.
3. We discuss the scalability and assessment elements of automated security solutions, which are critical when these solutions operate in production environments.
4. We identify open problems, challenges, and future works in the area of security automation in SDN.

This work represents the first attempt to study SDN-based security solutions from an automation perspective.

1.1. Existing surveys

The problem of securing SDN networks has attracted a lot of attention. Many comprehensive reviews have been produced recently. These reviews divided security and SDN into two categories: security for SDN, and SDN as a security solution for networks. In the first category, security issues specific to the SDN architecture and their solutions have been surveyed [28–31]. The most common security issues targeting the SDN elements are the DoS/DDoS attacks. Therefore, many surveys focus on these attacks and their solutions [32–35].

In the second category, the SDN-based security solutions to protect traditional networks have been reviewed. The standard strategy to review these solutions has been to divide them into statistics-based, ML-based, DL-based, and metaheuristics-based solutions [23–25,36–38]. An alternative approach has been threat-oriented reviews [39–41]. These reviews define different types of existing threats to networks and study SDN-based solutions to these threats. Besides, several surveys integrate the study of both categories, security for SDN and SDN-based security solutions [26,42,43].

Recent work in [20] reviews self-aware computer networks that include those that propose the use of SDNs as a means to implement these concepts. It analyzes the benefits of ML and RL mechanisms to implant self-awareness into networks. Additionally, this study integrates the QoS, energy, and security in a system. This system uses an RL-based decision engine to achieve a self-aware network. However, this study does not focus on SDN-based defense and does not analyze the level of automation and complexity of the self-aware mechanisms.

In contrast to previous surveys, we study SDN-based security solutions for networks from an automation point of view. With this purpose, we propose a new taxonomy of security solutions according to different levels of automation and complexity. The level of automation of these techniques is evaluated and compared in terms of their self-* properties. No previous survey has studied the automation capabilities of SDN-based security solutions. We also introduce a discussion about the scalability and assessment factors of the SDN-based security systems, which are fundamental in the design of complete, trustworthy, and ready to deploy automated solutions. Finally, we present the problems and challenges of current defense solutions and future research directions in the area of security automation in SDN environments.

Table 1
List of acronyms.

AI	Artificial Intelligence	**IDS**	Intrusion Detection System
AP	Application Plane	**ML**	Machine Learning
APT	Advanced Persistent Threats	**MTD**	Moving Target Defense
CP	Control Plane	**NBI**	North Bound Interface
DL	Deep Learning	**NFV**	Network Function Virtualization
DP	Data Plane	**OF**	OpenFlow
DPI	Deep Packet Inspection	**QoE**	Quality of experience
EWBI	East West Bound Interface	**RL**	Reinforcement Learning
FP	False Positive	**SBI**	South-Bound Interface
GT	Game Theory	**SDN**	Software Defined Networking
HMM	Hidden Markov Model	**VNF**	Virtual Network Function

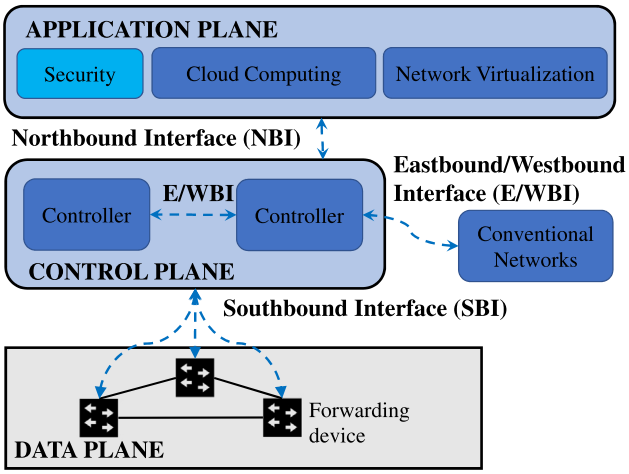

Fig. 1. General architecture of SDN.

1.2. Survey organization

This survey is organized as follows. Section 2 presents the background knowledge of the SDN architecture and the current state of the SDN technology. Section 3 introduces the taxonomy of the methods searching for network defense automation using the SDN technology. Section 4 reviews the reactive security solutions for networks using SDN and intelligent techniques. Section 5 reviews the state-of-the-art proactive security solutions. Section 6 summarizes the findings of the level of automation and complexity of reactive and proactive solutions. Discussion and challenges are presented in Sections 7 and 8, respectively. Finally, the conclusions are drawn in Section 9. Table 1 shows the list of the acronyms used in this survey.

2. An overview of SDN

SDN is a new networking paradigm where forwarding hardware is decoupled from control decisions. SDN provides new networking capabilities: dynamic network programming, centralized network monitoring, network virtualization, among others.

Fig. 1 shows the general architecture of SDN. It has three planes: the data plane (DP), the control plane (CP), and the application plane (AP). The DP contains forwarding devices that forward, drop, and modify packets based on policies received from the CP. The CP programs the network resources, updates forwarding rules dynamically, and makes network administration flexible and agile. The AP contains applications which according to the business requirements, can implement the control logic to change the network behavior. Examples of applications are Network Virtualization, Security in Networks, Mobility Management, and Network Monitoring.

Three interfaces are present in the SDN architecture. The southbound interface (SBI), defined between the DP and the CP, permits monitoring and notifies network events from the DP to the CP. Also,

Table 2
SDN-like architectures.

Alternative	Example	Characteristics
SDN via APIS	SNMP, CLI, NETCONF	Low changes, distributed, not open
SDN via hypervisor	NVP (Nicira), Juniper's Contrail, Cisco's ACI	Network virtualization, automation, centralized control
SDN via opening up the device	Devices capable of running open-source software	Distributed, open devices

SBI enables to forward control policies from the CP to the DP to provide functions, such as programmatic control of devices. OpenFlow (OF) is the first and most popular protocol for SBI [44]. NetConf and SNMP are other commonly used SBI APIs. The Northbound Interface (NBI), defined between the CP and the AP, is used by the applications to exploit the abstracted network provided by the CP to express the network behavior. Rest, Java, and Python are examples of NBI APIs. Finally, the Eastbound/Westbound Interface (EWBI) allows communication in multi-controller environments. The EWBI also enables the connection of the SDN controller to other network architectures.

2.1. State of development of SDN technology

The original and ideal characteristics of the SDN paradigm are: control plane separation, simplified forwarding device, centralized control, network automation and virtualization, and openness [45]. The networking paradigm with all these characteristics, hereafter, is referred to as *Open SDN* [45]. *Open SDN* was expected to bring significant benefits to networking, such as logically programmable infrastructure, centralized administration, dynamic and fast optimization of network operation and services, networking automation, and low-cost deployments. OF-based SDN is an example of *Open SDN*.

The OF-based SDN has several limitations [38,43,45–47]. The main limitations of this technology are that OF excessively decouples and centralizes the control of the network and allows complete visibility. These factors imply that all network traffic needs to be relayed to a central controller. To implement novel solutions, such as traffic management in high-performance networks, one has a difficult task since it results unfeasible due to the bottleneck created in the controller and the switches. In addition, the controller is vulnerable to hardware and software failures. Without the controller, the whole network lacks functionality. Solutions to these problems have been extensively studied as established in different surveys [28–35].

Alternatively, the SDN movement has led to the origin of SDN-like architectures to solve real and urgent networking problems and render new and innovative network services in production environments. These network overlay technologies were designed to create new competitive environments shifting the focus from physical infrastructure to software features. They can provide great agility and automation, simplified operational requirements, and extensive configuration function. These solutions can be seen as alternatives to *Open SDN*, which address real-life problems. Table 2 shows these alternatives to SDN-like solutions [45].

It is important to recognize that these alternative SDN-like solutions do not face all the issues and implementation challenges of the *Open SDN*. However, they present partial advantages of the original idea of the SDN paradigm. For instance, openness is not a characteristic of the examples of SDN via hypervisor presented in Table 2. In this survey, we include both *Open SDN* and SDN-like based security solutions.

Current trends in SDN include leveraging new technologies such as P4 to increase network programmability and openness. P4, which stands for Programming Protocol-independent Packet Processors, provides high visibility and control of events in the data plane. This technology releases network devices from closed protocols and runtime APIs (such as OF). Moreover, P4 increases the action field of the devices in the data plane, which reduces the SDN-controller processing overhead. Thus, this technology boosts the overall SDN-based network performance, enables *deep network programmability*, and amplifies the spectrum of possible network applications [48].

3. Defense automation in SDN-based networks

Network security automation is a prominent area of study and is expected to be solved by modern network architectures, such as SDN. We noticed different methods presented in the literature that pursue the automation of defending networks using SDN and intelligent techniques. To review such techniques, we divided the security solutions into two main classes: reactive and proactive. Fig. 2 shows the proposed classification analyzed in the following sections. Reactive or detection-based security solutions use different techniques, including Statistics, Machine Learning, Deep Learning, and Reinforcement Learning to detect and mitigate attacks. Proactive solutions prevent attacks by leveraging techniques such as Moving Target Defense, Network Function Virtualization, Cyber Deception, Network Slicing, and Blockchain technology.

Each subclass of defense strategy requires a different level of complexity (computation, storage, and implementation), also in order to achieve different levels of automation (Fig. 2). The level of automation of the classes can be assessed as a function of their self-* capabilities [22]:

- Self-healing: To automatically detect, diagnose and recover from faults that result from internal or external attacks.
- Self-adaptation: To constantly provide resilience towards internal and external changes in the networks; the system adapts to detect zero-day attacks and AI-enabled adversarial attacks.
- Self-configuration: To automatically configure the system following high-level specifications, and to self-organize into desirable structures and/or patterns.
- Self-optimization: To constantly seek the improvement of the system performance and efficiency.

These parameters are analyzed and compared in the following sections.

4. SDN-based reactive defense

4.1. Threat detection based on statistics

The SDN paradigm has high computation capabilities in the controller, which allows the implementation of advanced and effective methods to detect attacks. Specific attacks, such as DDoS, have been intensely studied, and their typical behavior can be used to detect them quickly. For instance, in an exploitation DDoS attack [49], many widely distributed zombies can attack a host, a segment of the network, or a service, by flooding the target with packets to exhaust the victims' bandwidth or resources. Such a DDoS attack can be detected by registering the number of packets directed to an IP destination and analyzing this information.

Two methods widely used to characterize flows in networks and analyze suspicious behaviors are entropy and wavelet transform. The entropy is a measure of uncertainty of a variable that allows measuring the information gain. The acquisition of information corresponds to the reduction in entropy. Entropy has been extensively used for anomaly detection since it enables the detection of changes in traffic distribution of several attacks [50–52]. For instance, during a DDoS attack on a specific service, there is a significant decrease in the destination IP and port entropy [53]. During a Port Scan attack, the attacker sends probe packets to a wide range of ports in a specific host to find available services. In this case, a significant decrease in the source port entropy can

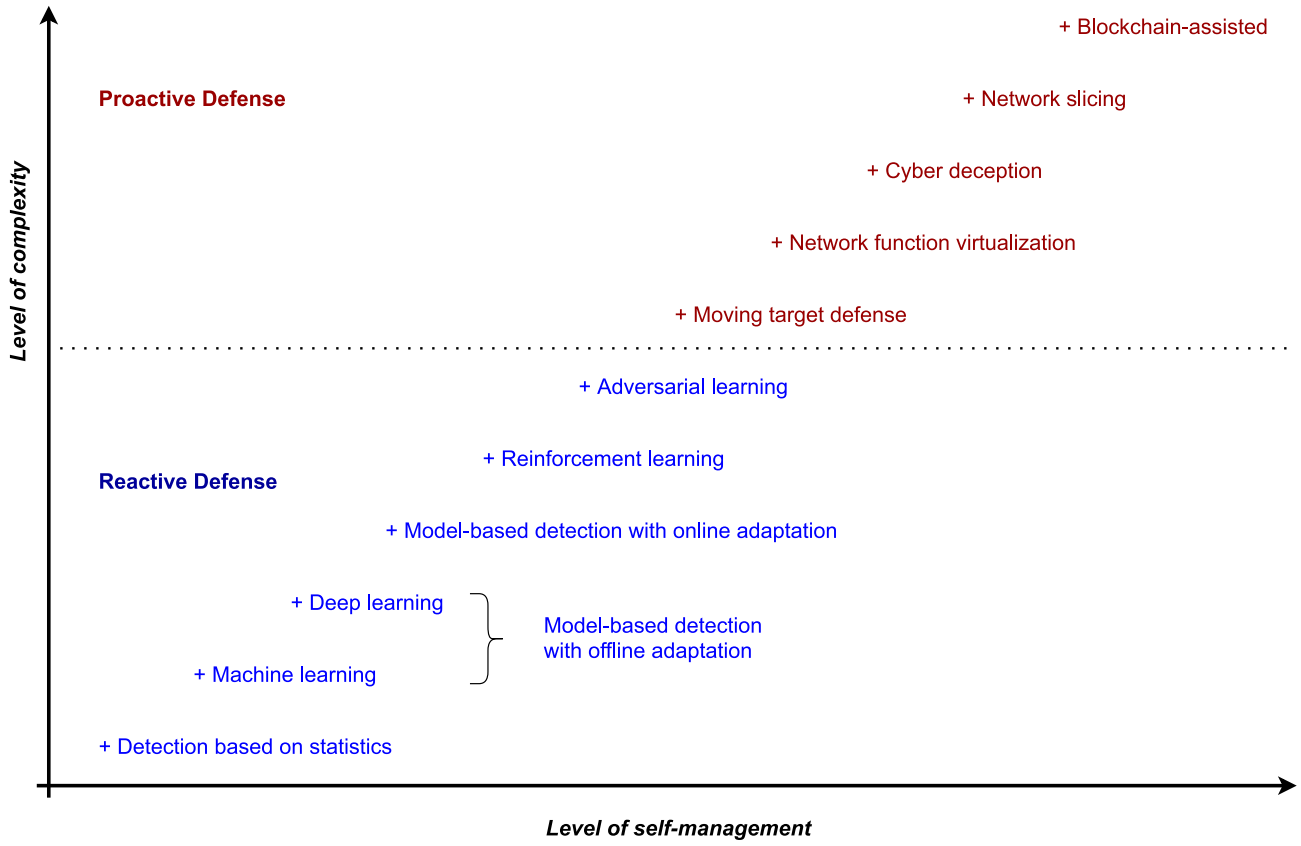

Fig. 2. Different methods searching for network defense automation with SDN technology.

reveal the attacker if it does not use random source ports [54]. These anomalies can be identified using predefined thresholds on changes in the entropy values.

The accuracy of the entropy-based approaches can decrease drastically at lower attack rates because the entropy difference in low-volume attacks is almost non-detectable. These threats are frequent in IoT-based port scanning and DDoS attacks. Given that the thresholds are defined as fixed parameters during normal traffic conditions, this lack of adaptability can reduce the efficiency of these methods. Finally, a fundamental parameter to fit is the window size used to calculate the entropy, which must trade off processing usage and detection accuracy. It is better to have adaptive window size and threshold.

Wavelet transform is another tool to detect attacks in networks, particularly in flooding attacks [55]. Wavelet transform is helpful in data feature extraction or characterization. It transforms data from time to scale domain. This method allows to obtain two kinds of coefficients, one is the low-frequency coefficient, and the other is the high-frequency coefficient. These coefficients can help to detect attacks, using defined thresholds to identify the presence of anomalies [56].

Complexity and level of automation

Statistic-based detection approaches are well explored and have a relatively low calculation overload and storage demand. These methods are real-time capable and can handle a large amount of traffic data.

Detection methods based on statistics cannot help to achieve the self-configuration and self-optimization properties. Although statistical methods achieve self-healing for known attacks, they do not achieve self-adaptation. These techniques operate under human supervision and often fail to detect sophisticated zero-day, polymorphic, and AI-enabled attacks. That fact has driven the security mechanisms to evolve to the more robust threat detection methods based on models using different intelligent techniques, presented in the following.

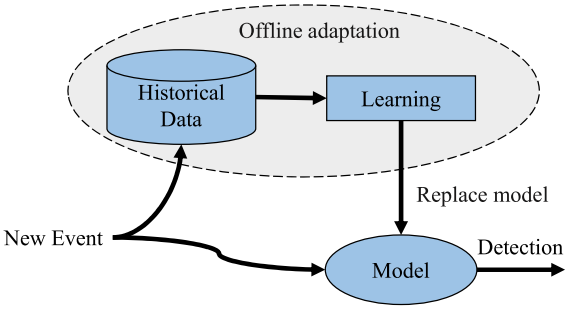

Fig. 3. Attack detection with offline adaptation. For model updating, a new model is trained offline and it replaces the existing one.

4.2. Model-based attack detection with offline adaptation

Model-based methods learn the behavior of a network using historical data, including attacks and non-attack events. These models are trained once and then used to detect attacks. That is, these methods learn the system's behavior offline. Fig. 3 shows the workflow of the offline learning attack detection approach.

Offline learning is necessary given the high dynamics of networks. On unexpected and extreme network changes, the models being in use to detect attacks will require updating through offline retraining to adapt to the new network's behavior.

Offline learning methods include ML-based and DL-based attack detection methods.

4.2.1. ML-based attack detection

Many proposals have demonstrated that SDN architecture is suitable to deploy ML algorithms given the higher computational resources in

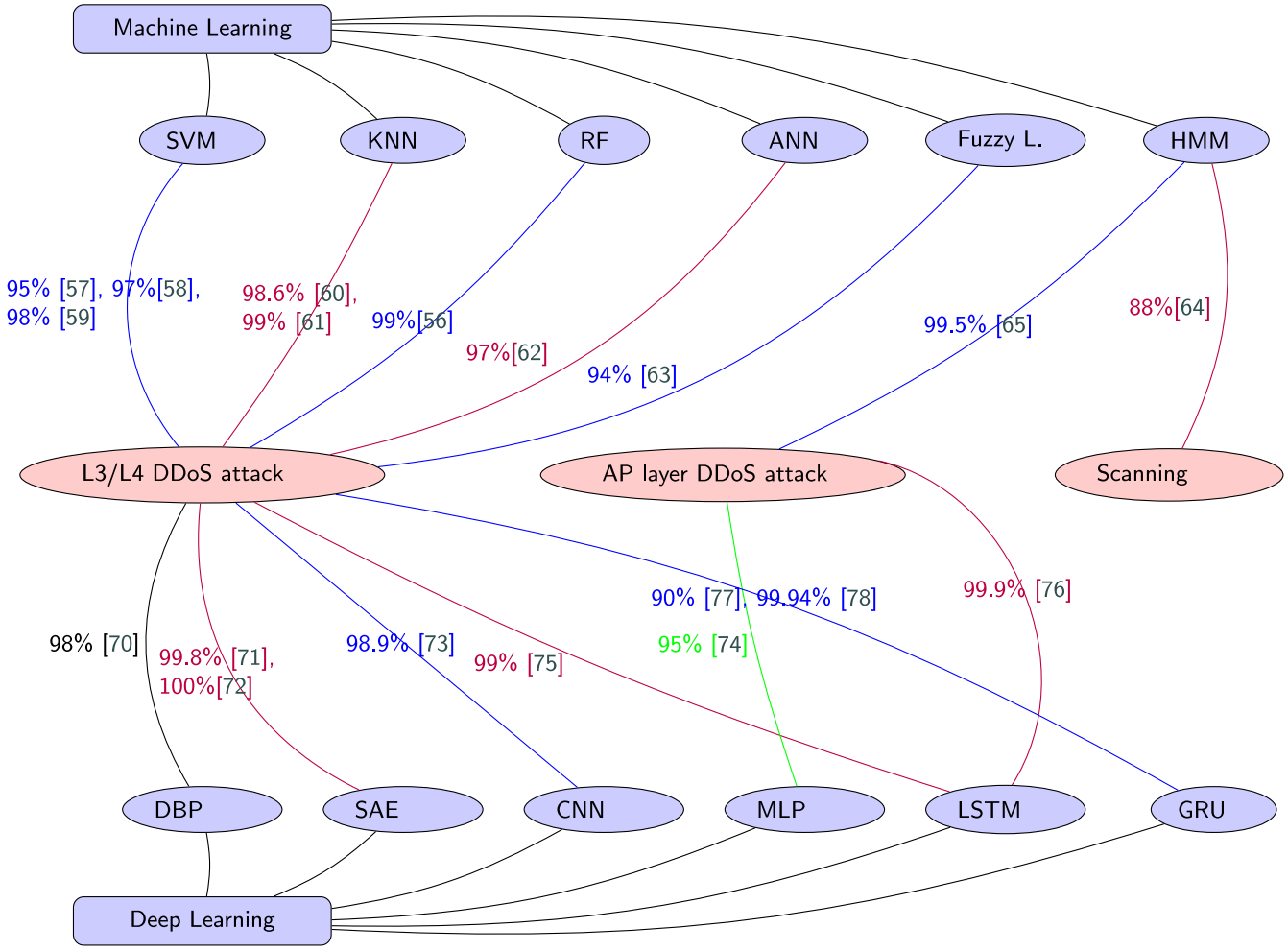

Fig. 4. ML and DL techniques applied to SDN-based security solutions. The accuracy achieved by each proposal is shown.

the CP and the global visibility of the network in real-time. Many surveys revealed that almost all ML techniques were explored jointly with SDN technology to detect complex network attacks [24–26,36–38,40,42]. Therefore, we decided not to do redundant work by reviewing all those works. Instead, we analyze how recent ML methods support detecting attacks in SDN networks.

Fig. 4 shows the most used ML methods to implement SDN-security solutions: Support Vector Machine (SVM) [57–59], *k*-Nearest Neighbors (KNN) [60,61], Random Forest (RF) [56], Artificial Neural Network (ANN) [62], Fuzzy Logic [63], and Hidden Markov Model (HMM) [64,65]. Most of ML-based security solutions focus on Network/Transport Layer (L3/L4) DDoS attacks, which are high-volume attacks and easier to detect. Application Layer DDoS attacks that are harder to detect are also solved using ML methods.

Other less common model-based mechanisms based on intelligent strategies use feature-pattern graph model [66], Holt Winter for Digital Signature System (HWDS) [67], catastrophe theory [68], and time-series data mining (e.g., ARIMA) [69]. The methods achieve accuracy above 90% on DDoS attack detection.

Even though most ML methods provide high accuracy in detecting threats (Fig. 4), they still lack a fundamental factor that could limit their implementation in production networks, namely the scalability. Most of the ML-based solutions have been tested using security datasets or small test networks. Therefore, the complexity of these techniques is not optimized. This shortage could bring critical problems when implementing these systems in production environments, such as unacceptable delays.

4.2.2. DL-based attack detection

DL algorithms are capable of automatically finding correlations in a high volume of data. For that reason, DL-based techniques have been used in the last years to detect complex attacks on networks. Several surveys have studied the application of DL for attack detection in SDNs [36–38]. Therefore, we decided not to do redundant work by reviewing all those works. Instead, we explore the most recent DL-based attack detection proposals.

Fig. 4 shows the application of DL methods to attack detection. The most common DL methods explored are Deep Belief Propagation (DBP) [70], Stacked Auto Encoder (SAE) [71,72], Convolutional Neural Network (CNN) [73], Multilayer Perceptron (MLP) neural network [74], Long Short-Term Memory (LSTM) neural network [75,76], and Gated Recurrent Unit (GRU) neural network [77,78]. These methods demonstrate high accuracy in detecting L3/L4 DDoS attacks. Most interestingly, DL helps to detect complex APTs, namely, application-layer DDoS attacks, with high accuracy.

It is noticeable that DL allows higher accuracy than ML in detecting attacks (Fig. 4). However, DL requires more computation and storage capabilities, which may limit the scalability of the proposed methods.

Nevertheless, given the latest CPU and GPU capabilities customized for DL, this could not represent a relevant problem.

Complexity and level of automation

ML techniques have been extensively studied to detect attacks. DL techniques are newer than ML and continue to be explored as promising solutions to achieve high accuracy in detecting sophisticated threats.

Offline learning methods focus on two main goals: (1) to minimize the complexity of the security solution and (2) to maximize its detection performance. The complexity can be reduced by optimizing the algorithm/model or by minimizing the number of input variables. High detection performance, measured in general by accuracy and false positive (FP) rates, can be achieved with more complex algorithms/models and using a high number of input variables. Therefore, these techniques need a trade off between accuracy and complexity. DL methods have higher demands of these resources than ML methods.

As the training is an offline process that can be performed on the idle time of the controller or on an alternative server, the time overhead of this process can be ignored. However, several researchers have proposed data preprocessing methodologies, whose overhead is an important factor. Also, the periods to update (offline) the models should be defined appropriately, according to the attack detection accuracy desired and the computational resources available.

Although these offline learning-based techniques are robust to detect complex and zero-day attacks, they may not adapt timely to drastic changes in the network. Furthermore, these methods can be easily defeated by intelligent attackers. These methods demand periodic human supervision to update/correct them. It is important to recognize that offline learning strategies with appropriate update periods could reach enough adaptability to follow timely network dynamics. Nonetheless, the emerging techniques of adaptive ML [79] will potentially be a key enabler of autonomous threat detection capable of operating in ever-changing contexts.

Finally, model-based attack detection with offline adaptation cannot help to achieve the self-configuration and self-optimization properties.

4.3. Model-based attack detection with online adaptation

As networks are constantly evolving, detection models with online adaptation can be appropriate strategies to timely follow the network dynamics, thus increasing the detection rate, minimizing the FP rate, and reducing the detection time.

To better understand the necessity of the detection model adaptation, we present the *concept drift* in the following. Consider X, the input features, and Y, the output variables of a system to be modeled. According to the Bayesian decision theory, a classification can be described with the equation,

$$P(y|X) = \frac{p(y)p(X|y)}{p(X)}, \tag{1}$$

where, $p(X) = \sum_{y=1}^{n} p(y)P(X|y)$.

In dynamically changing and non-stationary environments, the system behavior can change over time. The changes in the system may affect the probability distribution of the input data $p(X)$, the prior probabilities of the classes $p(y)$, the class conditional probabilities $p(X|y)$, and as a result, the posterior probabilities of classes $P(y|X)$, affecting the prediction [80]. These variations over time of the distribution probabilities are known as the phenomenon of *concept drift*.

Computer networks are dynamic and non-stationary environments where the *concept drift* is present. Therefore, threat detection models need to have mechanisms to adapt to evolving data over time. Otherwise, their performance will degrade. These models must be updated with the new data distribution or be replaced to meet the changed situation.

Online learning strategies can be adequate to deal with the *concept drift* since they process data sequentially. These strategies can produce a model with a partial training dataset, and the model is continuously updated during the operation as more information arrives. Fig. 5 shows the general structure of online adaptive learning that can effectively prevent the *concept drift*, according to [80]. The online learning procedure has the following steps:

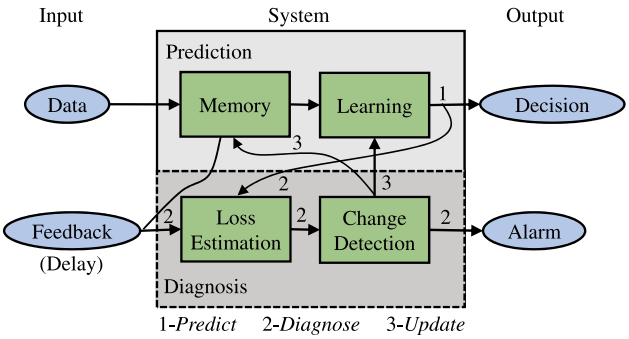

Fig. 5. General structure of online adaptive learning.

(i) Predict when a new example arrives, using the current model;
(ii) Diagnose, when receiving the correct label (delayed), the loss function is estimated; and
(iii) Update the model, if required, using the new example.

In this structure, a *Memory* module defines how and which data is presented to the *Learning* module. The *Loss Estimation* module tracks the performance of the learning algorithm. It also sends information to the *Change Detection* module to update the model if required. For details of each component in this architecture, please refer to [80].

From the implementation perspective, as model adaptation is online, it is necessary to consider the SDN controller processing capabilities. Moreover, the initial model can be trained offline, using historical data, and loaded into the SDN controller. Subsequently, the mechanisms of online adaptation will update the model using measures gathered from the network. In general, we assume that online learning mechanisms are unsupervised since there is no immediate and direct feedback about the model decisions. Therefore, anomaly detection mechanisms presented in the literature [81] are in the category of model-based attack detection with online adaptation, as long as they update the model according to network variations. In this way, unsupervised and adaptive ML/DL methods are in the category presented in this section.

For instance, work in [82] presents an online learning algorithm to detect Heavy Hitters (HH) in SDN networks. HH are entities whose number of network elements is at least a specific fraction of the total number of network elements (e.g., flows whose volumes are larger than certain thresholds). This study detects heavy traffic. A model receives flows collected from the switches. A market list M contains HHs historically identified. The online learning mechanism maps every incoming flow to a network state ($w = 0$ for HH and $w = 1$ for normal network activity) according to its bytes and weights changes. The weights and thresholds in M are pre-defined by the administrator based on the characteristics of the flow and a historical dataset. The network state feedbacks on the market list M. The OF protocol gathers statistics from the DP. To reduce the overhead of gathering flow information of the switches, the controller decides how to change the period of collecting statistics and sends coarse-grained or fine-grained rules to the switch as a zoom-out or zoom-in of existing rules. The evaluation of this proposal, based on detection accuracy, detection time, and communication overhead, demonstrates its effectiveness.

Although we can find several studies that propose online unsupervised learning attack detection (anomaly detection), the incursion on developing comprehensive online learning mechanisms to detect attacks on networks leveraging the SDN technology is limited. A complete online detection method should solve the *concept drift*, explained before, using a structure similar to that shown in Fig. 5. The limited

exploration in this due to online learning does not necessarily mean superiority to offline learning. In fact, due to the need for real-time model adaptation of online mechanisms, they can produce high processing overhead or even high latency. Hence, the importance of developing light-weighted but effective online learning mechanisms.

Streaming algorithms are another approach to achieve adaptability in attack detection and operate between online and offline learning methods. They can process high-speed continuous flows and use limited processing and memory. Therefore, examples are sequentially processed as they arrive in a few steps. Streaming algorithms are less restrictive than online algorithms since they can handle the input batch by batch to update the decision model and offer faster adaptation than offline learning. Streaming algorithms can have random access to previous examples or representative/selected examples. Thus, streaming algorithms can effectively detect threats in SDN, offering a high level of adaptability to network changes and the scalability required since resources on the controller are limited.

Complexity and level of automation

Model-based attack detection techniques with an online adaptation that solves the *concept drift* require further explorations. These techniques demand high computational resources. One significant advantage of online learning is the low requirement of storage capacity because these methods preserve limited batches of samples for model adaptation. The implementation of these techniques may require elevated expertise in implementing sophisticated algorithms.

In addition, these techniques learn the network behavior in real-time. Therefore, they provide the advanced capabilities needed to adapt against zero-day attacks. These conditions make these techniques to be less dependent on human supervision. However, AI-enabled attacks can still represent a big concern for these methods. Finally, online adaptation only on the attack detection cannot enable the self-configuration and self-optimization properties.

4.4. Reinforcement learning for threat mitigation

Reactive solutions require a mitigation part after an attack has been detected. Blocking or dropping malicious traffic is the most basic mitigation strategy [56,59,63,67,83–85], and it assumes an attack detection performance of 100%. However, even complex DL-based solutions cannot achieve such perfection in detecting attacks. Moreover, other mitigation strategies have been proposed, such as delaying suspicious traffic [86–88] or redirecting the suspicious traffic to Network Processor Units (NPUs) such as deep packet inspection (DPI) modules for deeper traffic analysis [62,89]. These approaches are intended to reduce the effect of legitimate traffic affected by false positives. However, in these solutions, attackers still use part of the network resources.

A need arose for a defense system that would not only detect an intrusion on time but also would make the most optimal real-time crisis-action decision on how the network security policy should be modified to mitigate the threat. Thus, mitigation solutions based on Reinforcement Learning (RL) have been proposed.

In the RL paradigm, software agents automatically define the ideal behavior within a specific context by continually making value judgments to select good actions over bad [90]. In a typical RL model, an agent monitors the environment via perception. It also takes actions within the environment, via an actuator, which could change the state of the environment. At each time step, the agent receives an observational input reflecting the state s of the environment. In turn, it can take an action a that transfers the state of the network into a new state s'. This is conveyed to the agent, along with a reinforcement signal r, indicating the value and desirability of the action held [91].

Attack mitigation proposals configure certain states, action-sets, and rewards to implement the RL to automate the attack mitigation [90,92–94]. Work in [91] proposed a unified general scheme for designing complex and adaptive RL agents for the network threat mitigation task. This proposal is able to automate the mitigation of complex attacks, such as APT.

RL-based mitigation strategies still face critical practical obstacles. Low sample efficiency and reward function specification are possibly the biggest obstacles to applying RL algorithms. Other critical shortcomings of RL are:

- It is a very immature technique;
- It is a highly computational extensive technique and data-hungry, that is, it requires to interact with the environment at each new training iteration;
- It requires stationary or fully simulated environments;
- It requires deep expertise in its implementation; and
- Most current data science and ML platforms do not have RL capabilities.

RL promises an active and automated solution to complex threats. However, further exploration is needed.

Complexity and level of automation

RL-based mitigation strategies are adaptive to the environment and changes in the attacks. However, these methods are emerging, and crucial practical challenges need to be solved, such as elevated computation and data storage demands. Moreover, it requires a high level of expertise to define the states' models and reward function shaping. Nevertheless, solving these issues will lead to a big step towards security automation in networks.

RL-mitigation methods combined with Network Function Virtualization technology could provide the capabilities of self-configuration and self-optimization to protect networks. However, further efforts are required to solve several related issues, such as the scalability property.

4.5. Adversarial learning

All defense proposals studied previously unrealistically assume that the attackers are not aware of the defense mechanisms. However, once deployed these implementations in production networks, intelligent and adaptive adversaries may violate the initial assumption of the AI techniques. The commoditization of ML tools can enable attackers to perform many experiments to identify how to use them for malicious purposes. Also, the use of public datasets on security model training is a problem because attackers could use the information to create a strategy for poisoning training data, known as adversarial samples.

Most ML methods are sensitive to adversarial samples, e.g., regression models, SVM, DT, RF, NN, and KNN. In addition, work in [95] demonstrated that adversarial attacks have a relatively high impact on different DL-based detections, such as (MLP), CNN, and LSTM, resulting in a reduction of the detection accuracy and increasing the training time. Even adaptive techniques, such as RL, can be affected by Adversarial Machine Learning (AML) methods. Authors in [96] demonstrated two ways of how AML can affect RL-based defense performance: (1) the AML applies a perturbation to the state observed by the target so that the agent chooses a different action as the optimal, and (2) the AML corrupts the reward function, so the direction towards the loss function minimization is perturbed, which results in delayed optimal policy implementation. The study showed that even these simple attacks might cause RL agents to take suboptimal actions. The authors suggested that simple solution ideas such as adding a 5% error to the reward signal in free attack periods can help the agent learn optimal actions, although some delay is maintained.

The defense methods must be in constant work against adversarial activities. The work in [97] proposed to reduce the impact of adversarial impact by generating synthetic attacks that can target networks with SDN architecture and using them to train a detection model to capture different attack variations. They proposed a Generative Adversarial Deep Neural Network (GAN) to automate the generation of realistic data in a semi-supervised manner. GAN applies a set of non-linear transformations on an original malicious sample to generate an adversarial example that evades classification models. GAN helped the adversarial

examples generation that can deceive two ML detection models for SDN, KNN, and RF classification models. The authors evaluated the decrease in attack detection rate as the effect of adversarial attacks.

Protecting AI-powered security systems and considering malicious use of AI by attackers are part of the security strategy technology trends for 2020 [98]. Adversarial learning jointly with SDN technology has not been widely explored to solve the security problem in networks. As the attacks are becoming complex, the use of methods capable of defeating intelligent adversary attacks is becoming critical towards the automation of network defense.

Complexity and level of automation

Methods that countermeasure adversarial AI-enabled attacks require high computation capacity, medium storage extent, and elevated AI implementation abilities. These methods can timely adapt to zero-day attacks and thwart AI-enabled adversaries. That makes these methods highly independent of human supervision.

These methods can be enhanced using the Network Function Virtualization technology to provide the capabilities of self-configuration and self-optimization. Nevertheless, these methods have not been widely explored. Sophisticated and smart attacks are expected to target networks in the following years. Therefore, it is recommended to keep exploring adversarial learning techniques that enable the detection of dynamic threats in adversarial environments.

5. SDN-based proactive defense

Reactive approaches imply obtaining as much telemetry as possible to detect potential attacks. Even if SDN simplifies and centralizes the network monitoring and control, the detection/mitigation approach is costly for the defender since logs management is storage greedy, and threat detection is expensive in computation. Furthermore, the attackers will briefly affect the network performance until they are detected, causing considerable costs. The defender's role is unfair in this approach since it attempts to prevent intrusions at every possible location. Meanwhile, the attacker only needs to discover and exploit a single vulnerability to breach the complete defense.

In contrast, proactive defense methods allow inverting this asymmetry between attackers and defenders. These methods implement mechanisms to evade/prevent attacks or reduce the attack surfaces in the network. SDN has been a fundamental element in designing and deploying proactive defense for networks, given its centralized monitoring and control of networks with dynamic programming capability.

Next, existing methods of proactive defenses are analyzed.

5.1. Moving target defense (MTD)

MTD consists of mechanisms that automatically implement diverse and dynamic network configurations to make the system attack surface unpredictable to adversaries. As a result, the system status with the MTD strategy is hard to exploit and more resilient to different forms of attacks [99,100].

MTD mechanisms are not new as they have been used in conventional networks. However, in these traditional architectures, the implementation of MTD requires the reconfiguration and synchronization of network devices, interrupting the current communication sessions, evidencing low efficiency and high cost. Using SDN architecture that offers a highly flexible and dynamic programmable network, we can deploy specialized network applications and services to create robust MTD systems.

The adaptive modification of IP addresses and ports of hosts can prevent recognizing-based attacks like man-in-the-middle and port scanning attacks [101]. We can randomly replace the source and destination IP address of packets with virtual IP addresses in the SDN switches, which enhances the unpredictability of the network behavior [102].

The selection of dynamic routes provides significant advantages to avoid adversaries eavesdrop on the network links or launch DDoS/DoS attacks on certain network links. However, they are still predictable and cannot provide resilience against intelligent adversaries. The use of random routing mutation (RRM) techniques can effectively defend the network while satisfying the performance and QoS requirements of the network [103]. Also, data of various users can be mixed, among different routes, making it difficult for attackers to recover complete data from one communication and avoiding attackers to separate data for a single user [104].

The lack of dynamism on the mutation cycles of RRM methods and their difficulty to search for an optimal mutation routing path represents a weakness against advanced adversarial. To deal with these problems, the generation of the routing paths can be specified and solved as an optimization problem, considering the constraints of routing paths' randomness and network load balance [105]. SDN facilitates this approach, given its global network monitoring and in-line network configuration.

The limitation of the MTD mechanisms is the scalability since they require monitoring the network state in real-time, which means a high burden on the controller and a high charge on the control channel to gather the statistics from the switches.

Complexity and level of automation

From the proactive security approaches, MTD solutions have been more studied. Nevertheless, further efforts are required to obtain scalable solutions. These mechanisms demand high capacities of computational processing and storage to maintain their dynamism. As a result, it provides adaptability against zero-day and intelligent attacks. The combination of MTD with Network Function Virtualization (NFV) technology enables the Self-configuration and self-optimization capabilities.

5.2. Network function virtualization (NFV)

NFV is a network architecture paradigm that virtualizes different network node functions, making these network functions (NFs) and services much more adaptive and scalable. NFV can significantly reduce the hardware cost, greatly improve operation efficiency, and dramatically shorten the development of the lifecycle of a network service [106].

Although SDN and NFV are two distinct and independent technologies, combining them in a network can bring multiple advantages to emerging technologies. SDN provides centralized knowledge about the network status and programmability, whereas NFV boosts the development of virtualized network appliances executed on top of commodity servers [107].

Next, we review new strategies to defend networks using SDN and NFV. Work in [108] presents a proactive defense against complex DDoS attacks. This approach uses virtual networks (VN) to dynamically reallocate the network resources without disrupting the network service or violating the VN properties. The components of this solution are (1) the correct-by-construction VN migration planning and (2) the VN migration mechanism. The former implements and solves a constraint satisfaction problem using Satisfiability Modulo Theory (SMT). The second component controls the execution of the VN migration strategy on a virtualized physical network. This component handles all movement logistics, including the precise sequencing/scheduling of the steps' realization to complete the move and the timing of such process. The solution was tested against a crossfire attack. It showed high effectiveness in restoring the downgraded bandwidth because of the DDoS attack through the timely migration to a safety threat placement. Work in [109] proposes a defense system for IoT networks based on SDN and NFV. A reinforcement learning agent is used to evaluate the risks of potential attacks and take optimal actions to mitigate them.

SHIELD [110] was a project funded by the European Union to develop a security framework that offers Security-as-a-Service (SaS). SHIELD applied NFV/SDN for virtualization and dynamic placement of virtualized security functions in the network. In combination with Big

Data analytics, it allows for real-time incident detection and mitigation. The controller can dynamically reconfigure the Virtualized Network Security Functions based on the recommendations issued by the Big Data analytic.

MTD using SDN/NFV has also been explored. Work in [111] proposes an SDN/NFV-based MTD mechanism using multiple fuzzy systems and a proxy virtual network function (VNF) to achieve DDoS detection and mitigation. The fuzzy logic is used as an SDN application to detect whether traffic from an external network is DDoS traffic. NFV is used to implement a virtual relay node to prevent DDoS traffic directly affecting the target server. When an attack is not occurring, a module called reverse proxy VNF serves as a relay transmission between the user and the target server, thus transferring the DDoS attack surface. When an attack occurs, a mechanism based on SDN technology segregates suspicious traffic from legitimate traffic, redirecting suspicious traffic to a restricted node or dropping them.

Work in [112] proposes various NFV-enabled virtual shadow networks used in the MTD mechanisms implementation via route mutation. The aim is to dynamically change the routes for specific reconnaissance packets to harden the attackers' job of identifying the current network topology for potential DDoS attacks while enabling the defender to store potential attackers' information through forensics. This framework offers three MTD strategies to obfuscate the network topology information for thwarting indirect DDoS attacks at the reconnaissance phase. (1) to mute the routes of the real network to the real host destination, (2) to use virtual shadow network (VSN) (e.g., virtual honeypot) as the mirror network and use of one of the VSN as the destination of the route mutation, and (3) to use the VSN as an overlay network while the destination of the route mutation is the real host. In each cycle, a roulette wheel selection mechanism selects the MTD strategy using different predefined mutation probabilities. Also, the VSN placement uses a heuristic topology-aware algorithm.

Work in [113] proposes a topology mutation technique combining NFV and SDN to make the attack surface dynamic. The aim is to create a virtualized network architecture that can transform itself in real-time by dynamically introducing virtual components and rearranging the network's logical structure. Under this dynamic system, an attacker finds it complicated and costly to eavesdrop on network packets or monitor the network traffic to acquire the network topology. Also, the authors proposed honeypots to obtain a better security response.

Additionally, works in [106,114] surveyed other recent SDN/NFV based security approaches. NFV-based security solutions provide on-demand network programmability, energy efficiency, and mobility support, characteristics that are relevant to IoT networks.

One big concern of the NFV paradigm is that new generation networks are large-scale and have strict speed, latency, and resource-efficiency requirements. Software-based data plane functions could introduce unacceptable delays. Therefore, recent studies have proposed different solutions for high-speed packet processing in user-space, using common off-the-shield (COTS) hardware [115,116]. In addition, it is important to mention the vector packet processing (VPP) [115,117] framework that is a kernel-bypass method, which provides flexibility (multiprogramming) with hardware-comparable packet processing performance by using techniques of batch processing. VPP can leverage a general-purpose CPU to process packets in a vectorized fashion, which is highly efficient. Consequently, VPP provides full network functionalities of layers 2, 3, and above. The nature of VPP-enabled VNFs will provide the open-access, software-based, scalable, and high-performance capabilities to deploy automated security solutions in SDN-based networks.

Complexity and level of automation

NFV promises several advantages to defend the networks proactively. However, it can demand high computation and storage resources. In addition, NFV has been presented before, jointly with other mechanisms, to provide self-configuration and self-optimization capabilities. That is, in combination with other strategies studied in this paper, NFV can potentially provide complete security automation. However, research in this area is still emerging. Therefore, many challenges remain unsolved, such as optimal selection and deployment of SDN/NFV-based security mechanisms and optimal orchestration over different network domains [3].

5.3. Cyber deception

Another type of proactive defense mechanism is cyber deception. Deception tools distribute a collection of traps and decoys across a system architecture to imitate genuine assets. Unlike MTD, deception methods introduce uncertainty for the attacker without changing the actual surface but instead creating a virtual attack surface [118]. They provide adversaries with an incorrect view of the target system so that attacks based on such wrong information are likely to fail and present an opportunity for detection [118,119]. Examples of deception tools are simple honeypot systems. Modern deception tools require analytics and automation methods to make them simple to deploy but effective for thwarting attacks. The capabilities of SDN provide an opportunity to design an optimized and autonomous solution for network defense, which can permit advanced use cases of the deception platform such as integrated, proactive threat hunting, and active attacker engagement [120].

Works in [121,122] proposed reconnaissance deception systems (RCSs) to defend from malicious discovery and reconnaissance attacks in computer networks. These systems use SDN to simulate complete virtual network topology, thus achieving an effective defense strategy against adversarial reconnaissance. Work in [123] proposes an SDN-based framework for unobtrusively integrating decoy services into production workloads. The SDN technology divides a production network into segments of production and decoy. Attackers are redirected to decoys, which have the same configuration as the protected production system. Therefore, the attackers never reach the protected assets.

MimePot [124] is a cyber–physical honeypot conceived for industrial control networks. This security framework can simulate physical processes operations of Industrial Control Systems to capture intricate cyber–physical attacks using SDN technology. Work in [118] proposes a cyber deception system to combat cyber adversaries in a wireless virtualization framework. In this wireless framework, the SDN controller continuously monitors the network traffic, observes the connections, and creates mobile virtual network operators (MVNO). Additionally, it directs cyber adversaries towards deception MVNO. Work in [125] proposes a security framework with adaptive defense. This solution protects key nodes based on game theory (GT) through the formulation of a multistage game by using the architectural advantages of SDN. Moreover, it considers deceptive behaviors of the cyber defender in different stages.

Work in [126] presents a cyber deception solution that implements defenses against penetration attacks using a decoy chain development (DCD) method based on the combination of NFV and SDN. In this defense architecture, the SDN controller monitors the security status of the whole network and can detect possible attack sources using Intrusion Detection System (IDS) technology. Moreover, using SDN technology, the DCD can deploy decoy chains in the generic servers on the DP to change the attack surface, based on the global network status and under resource constraints and multiple attack sources. NFV helps to deploy dynamic service delivery in the network.

The goal of the security framework KAGE [127] goes beyond the implementation of sophisticated honeypots or MTD mechanisms to dynamically defend the system. KAGE expanded into a new area of active defensive deception, where the adversary's decision loop is deliberately manipulated by proactive and reactive adaptation of the environment as KAGE learns about the capabilities and intents of the client that KAGE is engaging. That is, a detected intrusion does not result in blocking or shutting down connections. Instead, KAGE seamlessly moves the

adversary's interactions into an alternate reality, where the adversary abides unproductively engaged for a long time, distracted from the real target.

Complexity and level of automation

Deception mechanisms have demonstrated the potential to improve the security automation in networks but, at the same time, optimizing the resources. However, this area still lacks further development. The costs and benefits of deception in defense need rigorous evaluations before practical use [127,128]. For instance, a cost would result in an equilibrium point between the security of having more deceptive virtual subnets and the increased delay of sending packets through virtual devices. The proper allocation of honeypots in the network to achieve better defense and resource utilization is also an important area to be further explored.

Combining deception techniques with NFV can provide all the self-* capabilities. However, this area requires further effort.

5.4. Network slicing (NS)

Network slicing strategies promise effective solutions such as dynamic QoS/QoE management [27] for IoT and 5G/6G [7,129] technologies. Network slicing allows to isolate different traffics to maintain security and reliability in the network [3]. Each slice is isolated and has diverse network functions and virtual resources. Thus, individual network applications and services can run on each slice, with different levels of security required by individual slices. The combination of SDN and NFV on network slicing can support the design of advanced security solutions with fined-grained security features, such as per-application traffic flow [129,130]. Still, further research efforts are required to achieve the required granularity protection, with dynamic and flexible management, and at the same time maintaining a low cost of implementation.

Complexity and level of automation

Network slicing, combined with SDN and NFV, promises to enable all self-* capabilities needed to obtain a fully automated solution. Furthermore, all techniques previously studied, including reactive and proactive solutions, can be integrated into each slice, thus providing a comprehensive security solution for new generation networks. However, this area is emerging, and strong efforts are required to obtain such solutions.

5.5. Blockchain-assisted SDN

In recent years, Blockchain technology has been explored to design more scalable, efficient, decentralized, secure, and trustworthy SDN architectures [131–135]. Blockchain enables verification information transactions via distributed network authorization and, subsequently, it adds that data to an unalterable ledger. That is, unknown entities can communicate with each other without a trusted third party (controller). The decentralization eliminates control by a single controller, removing the risk of single-point failures. In addition, since each information transaction is public, this technology offers a reliable and easily verifiable infrastructure [136–138].

Given the decentralization property and continuous nodes/users/information-transaction verification, blockchain can help to prevent the most common and disruptive attacks in SDN-based networks, namely, DDoS attacks. The primary use of this technology is to solve the consensus and synchronization problem of multiple distributed SDN controllers [132,139] and enable trustworthy controller/switch associations [140,141]. Nevertheless, we consider blockchain as a proactive mechanism since it ensures the executed transactions in the network must be approved and verified by all participating nodes, which creates irrefutable records to avoid data tampering and double-spending problems and ensure ledger consistency [139,142]. For instance, the blockchain technology integration into SDN provides capabilities to prevent IoT devices from being used as botnets that launch DDoS attacks [143].

Complexity and level of automation

Blockchain-enabled SDN is an emergent area of study. Still, many challenges remain unsolved in this technology [136]. A relevant issue is that pragmatically a blockchain may require a long time, even hours, to complete the update of all ledgers. This issue will cause delays and raise uncertainty for participating nodes. Moreover, storing a blockchain in OF switches requires memory resources in large numbers, which can become a bottleneck as the size of the blockchain increases [144]. In addition, the cost of computational power and energy of some consensus mechanisms can be prohibitive for organizations or groups interested in adopting blockchain technology.

Finally, blockchain is adaptive to new internal or external changes making it more robust to novel and complex attacks. Also, this strategy can enable security self-configuration. Self-optimized security solutions using blockchain and SDN require further development.

Table 3 shows a comparison of SDN-based proactive security solutions described before. It is noticeable that some proposals combined two or more proactive technologies/techniques to defend the networks from advanced attacks. In addition, it is fundamental to recognize the use of different AI techniques in the majority of these proactive solutions.

6. Analysis and comparison

In this section, we compare the complexity and level of automation of the techniques described before and present their common application areas.

6.1. Complexity and level of automation

Table 4 ranks proactive and reactive SDN-based security solutions, which summarizes the findings of the complexity and level of automation of the classes analyzed previously.

As described before, the complexity analysis includes the computational and storage resources and the expertise required to implement a technique. We analyze the automation level in function of the self-* capabilities that each strategy enables. In the case of self-adaptation, the analysis considered two factors: the adaptability to mitigate or prevent zero-day attacks and the ability to defeat AI-enabled adversaries.

In the case of complexity and adaptability capabilities, we provide a Null, Low, Medium, High, and Advanced ranking to assess how the SDN-based security solutions stack up against one another. Regarding self-healing, self-configuration, and self-optimization properties, we assign Yes (Y) or No (N) checks, which indicate whether a technique enables the corresponding feature. Furthermore, we include the progress of the strategies, which shows their development status.

In general, we identify that solutions with a high level of automation demand more complex designs, which implies more computation and storage capability requirements. Also, we note that the reactive solutions have been more explored than the proactive solutions, which still lacks the study of an optimal trade off between performance and accuracy on defending SDN-based networks. The adaptability of proactive solutions against zero-day attacks, adversarial environments, and AI-enabled attacks is advanced since these methods constantly evolve to minimize/eliminate attack surfaces. However, the elevated demand for computational resources is a common problem of proactive solutions. This condition may further limit the scalability of these solutions. Nevertheless, most of these techniques are emerging, and their explorations in the coming years will potentially solve these issues.

Finally, we recognized that NFV technology and AI are crucial to enable the self-configuration and self-optimization properties of almost all reactive and proactive security solutions. Techniques of RL, adversarial learning, MTD, cyber deception, and network slicing

Table 3
Proactive approaches for network defense with SDN.

Proposal	MTD	NFV	CDT	NS	BC	Defense	Attack Protection	Intelligence
[101]	✓					IP mutation	-	-
CHAOS [102]	✓		✓			IP and Port mutation, decoy servers	-	-
[103]	✓					RRM	DoS	GT
[104]	✓					DHC	Sniffer	-
[105]	✓					RRM	Reconnaissance, eavesdropping and DoS	Ant Colony
[108]		✓				Migration mechanisms	Complex DDoS	-
[109]		✓				Detection and mitigation	SSH password brute-force, Slow HTTP DoS	Reinforcement Learning
SHIELD [110]		✓				Dynamic placement of VF	-	Big data
[111]	✓	✓				Detection and mitigation	DDoS	Fuzzy logic
[112]	✓	✓				RRM	DDoS	Heuristics
[113]	✓	✓	✓			Topology mutation	Eavesdrop	-
[118]			✓			Virtual network	Cyberattacks	-
[119]			✓			Honeypots	Measurement of network deception's effectiveness	Bayesian Inference
RDC [121]			✓			Virtual topology	Reconnaissance	-
[123]			✓			Honeypots	Side channel fingerprinting	-
MimePot [124]			✓			Honeypots	Zero dynamic attacks	-
[125]			✓			Honeypots	Port scan, crack password, upload virus	GT
ACyDS [122]			✓			Virtual network, honeypots	Scanning attacks	SMT
[126]	✓	✓	✓			Decoy chain development	Penetrations	-
[128]			✓			Deceptive subnets	Scanning stage of APT	Genetic Algorithm
KAGE [127]	✓		✓			Active deception	-	-
[129]		✓		✓		Network slicing	Security in 5G environment	-
[130]		✓		✓		Network slicing	Security in 5G and IoT	-
[139]					✓	Smart contracts	DDoS attacks across multiple domains	-
DistBlock Net [142]					✓	Flow tables updating using blockchains	Securing large-scale IoT networks	-
BlockSDN [144]					✓	Permissioned Blockchain	Malware at DP and DDoS attacks to CP	-
[145]		✓		✓	✓	Blockchain	Protect Network Slices	-
[143]					✓	Distributed Blockchain	Prevent Botnets	

MTD = Moving Target Defense, NFV = Software Function Virtualization, CDT = Cyber Deception, NS = Network Slicing, BC = Blockchain-assisted, SMT = Satisfiability Modulo Theory.

Table 4
Comparison of the complexity, automation capability, and progress of development of the different methods for network defense using SDN technology.

Security solution	Complexity			Automation Capability					Progress
	Proc.	Stg.	Impl.	Adaptability		Self-*			
				Zero-day	Adversaries	SH	SC	SO	
Statistics	Low	Low	Low	Low	Null	Y	N	N	Widely explored
ML	Medium	Medium	Medium	Medium	Null	Y	N	N	Widely explored
DL	High	High	Medium	Medium	Null	Y	N	N	Medium explored
Online adaptation	High	Medium	Medium	Advanced	High	Y	N	N	Barely explored
RL	High	High	High	Advanced	High	Y	Y[a]	Y[a]	Emerging
Adversarial learning	High	Medium	High	Advanced	Advanced	Y	Y[a]	Y[a]	Little explored
MTD	High	High	High	High	High	Y	Y[a]	Y[a]	Medium explored
NFV[b]	Advanced	High	Advanced	Advanced	Advanced	Y	Y	Y	Little explored
Cyber Deception	Advanced	High	Advanced	Advanced	Advanced	Y	Y[a]	Y[a]	Little explored
Network Slicing	Advanced	High	Advanced	Advanced	Advanced	Y	Y[a]	Y[a]	Emerging
Blockchain-assisted	Advanced	Advanced	Advanced	Advanced	Advanced	Y	Y	N	Emerging

Proc.= Processing , Stg.= Storage, Impl.= Implementation, SH: Self-healing, SC=Self-configuration
SO = Self-optimization.
[a]In combination with NFV technology.
[b]NFV in combination with the rest of techniques promises fully automated security solutions.

cannot provide a self-optimized or self-configured security solution without NFV, which provides the flexibility to deploy (virtual) network functions on-demand and in real-time. Moreover, AI helps to remember, learn, and improve any strategy outcome that implements it.

Note that even the sophisticated design could not achieve complete independence of humans to supervise them. Still, more efforts are needed in this area to obtain more automated solutions. For instance, combining detection-based solutions with proactive solutions could result in an effective defense that entails minimal or no human intervention. However, it is essential to be aware that computational resources are limited.

6.2. Application areas

Table 5 shows the typical application of each security solution. Although we can implement all the studied defense solutions through all types of SDN-enabled networks, their complexity of implementation and unique advantages have made them focus on specific application areas and network threats.

Reactive solutions become proper for next-generation SDN-based firewalls [146] commonly used to protect data centers, campus networks, internet service providers (ISPs) infrastructure, and industrial control systems (ICSs) [147]. Most reactive defense proposals center on DDoS attacks, which are the most frequent and dangerous network threats. Nevertheless, each technique can solve different levels of sophistication of these attacks. Statistical methods can solve simple DoS/DDoS and port scanning attacks. More complex DDoS threats require the use of ML and DL methods. ML techniques render high performance for network layer (high-volume) DDoS threats. DL approaches demonstrate better results for application-layer attacks such as low-rate, slow-rate, and one-hit DDoS attacks. Online adaptation, RL, and adversarial learning techniques can defeat even more sophisticated APTs, including AI-enabled threats.

Table 5
Application of different methods for network defense using SDN technology.

	Security solution	Typical threats that solves	Areas of application
Reactive	Statistics	Port scan and simple DoS/DDoS attacks.	Next-generation Firewalls in Datacenters, Campus Networks, ISPs infrastructure, and ICSs.
	ML	Network Layer DDoS attacks (SYN flood, UDP flood, Port Scan)	
	DL	Application Layer DDoS attacks (low-rate, slow-rate, one-hit DDoS attacks).	
	Online adaptation	APT	
	RL	Mitigation of APT	
	Adversarial learning	AI enabled attacks	
Proactive	MTD NFV Cyber Deception Network Slicing Blockchain-assisted	DDoS attacks, eavesdrop, sniffer, botnets prevention.	IoT, UAV networks, 5G/6G, (VANETs)

Proactive solutions occur more proper to protect new generation networks, such as IoT, 5G, 6G, vehicular ad-hoc networks (VANETS), unmanned aerial vehicle (UAV) networks [148], etc. These networks have in common that they are heterogeneous, dynamic, and large-scale, making them hard to meet the strict requirements, such as low latency, high reliability, high mobility, top security, and enormous connectivity. DDoS attacks are the most common and disruptive threats in these transitional networks. Other threats treated by proactive solutions are eavesdrop, sniffer, botnets prevention. The heterogeneity of the devices could hinder the use of reactive solutions in these networks since they can expose to numerous FPs. Thus, preventing threats rather than mitigating them could be a better strategy in these complex systems. We also observed that almost all proactive security solutions work in combinations. Examples of these combinations are MTD-NFV, MTD-Cyber Deception, NF-Network Slicing, MTD-NFV-Cyber Deception, and Blockchain assisted-NFV-Network Slicing-NFV, as shown in Table 3.

Finally, we highlight that proactive and reactive solutions can be combined for their application in different scenarios. For instance, ML and DL can enable more effective and optimized proactive solutions. And conversely, NFV could bring the automation capability to the reactive solutions.

7. Discussion

We studied reactive and proactive strategies that pursue security automation in SDN-based networks. Applying these solutions to diminish the effects of attacks is powerful, but those solutions can have some drawbacks that sophisticated attacks can exploit to beat them. Once the attacker gains access to the system, proactive solutions cannot thwart them. Therefore, the detection-based mechanisms remain relevant. Worthwhile approaches can combine reactive and proactive strategies to automate the security solution in SDN-based networks. An automatic security solution can follow the rule: *Prevent everything you can, everything you cannot prevent, investigate, detect, and mitigate fast*.

The SDN-based security solution for networks is expected to be automated in the coming years. Such solutions must consider all advantages and limitations of the SDN architecture. A solution proposal that does not consider the effects on the network performance, such as the bottleneck to collect packets for analysis, will not be scalable. Also, there is a need for a proper evaluation of the network defense mechanisms. The proposals reviewed in this survey used partial and different metrics to assess their efficiency in protecting networks. A good practice is to use a standard evaluation system of security solutions for networks based on the SDN paradigm. Next, we analyze the scalability and assessment factors, which we consider essential elements to different security proposals before their implementation in production networks. Finally, we discuss open problems to manage automation.

7.1. Scalable SDN-based security solutions

Most of the SDN-based security solutions revised throughout this survey propose to perform main processes in the AP. The controller is in charge of capturing data from DP through an SBI protocol and deploying the mitigation strategies on the physical devices. In this way, the intelligent approach leverages the computational and storage resources of the SDN controller, as well as its global visibility and centralized control.

However, this approach can limit the scalability of the different security solutions and even introduce new vulnerabilities to DoS/DDoS attacks. That is, as the traffic demand of modern networks increases, analyzing all of these packets with a security system at wire speeds will be challenging using sophisticated algorithms, ensuring at the same time high accuracy.

Effective SDN-based security solutions require considering the optimization of different resources to provide scalability. Fig. 6 shows specific strategies to add scalability to SDN-based security solutions, which we discuss in the following.

7.1.1. Traffic sampling

Sampling techniques reduce the flow of data gathering from the controller and minimize the communication required between switches and the controller, thus easing the CP overload in the vast network traffic condition. The sampling period is critical in sampling strategies. The samples' requests must be periodic in short time intervals since long intervals are impractical for accurate and rapid detection of anomalies. The attack events appearing over a time scale inferior to the one selected for traffic sampling will make the security control rule useless.

Improvements to the sampling strategy include adaptive flow collection [149]. The improper implementation of sampling methods compromises the accuracy of threats detection, given the relevant flows missing probability.

7.1.2. Stateful SDN

We can solve the scalability of the *Open SDN* architecture by keeping data flows in the DP as much as possible to maintain the controller's safety and reduce the burden of the controller and SBI bandwidth.

In a stateful SDN architecture, the switches make independent forwarding decisions based on their local flow state information without contacting the controller [150]. The facts that supported the development of the stateful SDN architecture include:

(i) The computational resources available on current switches are sufficient to extend their functionalities, adding a certain level of intelligence;
(ii) Some threats, such as scanning, sniffing, and DoS attacks, can be locally identified and neutralized; and
(iii) The centralized control of stateful SDN architecture remains as in the *Open SDN* architecture.

Stateful architecture can help to effectively reduce the impact on the network performance during DoS/DDoS attacks [151–156]. Further, the P4 technology [157] and VPP-based VNFs will enable intelligence in switches, to obtain an open-access, automated and scalable security solution.

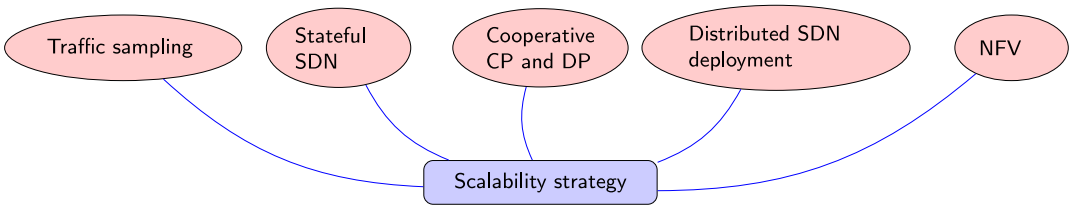

Fig. 6. Strategies to scale SDN-based security solutions.

7.1.3. Cooperative control and data plane

It is important to note that sophisticated attacks, such as slow DDoS attacks, might be troublesome to detect using local switches' information, meaning that stateful SDN architecture could not be efficient. A cooperative intelligence between DP and CP could potentially minimize the effects of advanced DoS/DDoS attacks on network performance [89,158,159]. In this approach, the stateful switches commonly perform coarse-grained detection to determine suspicious flows and send them to the controller. The controller leverages its computation capabilities to perform fine-grained classification of attacks and propose defense strategies. Finally, the switches react to mitigate the attacks, according to rules installed by the controller.

It is essential to design the cooperation mechanisms between the DP and CP carefully and effectively. Switches with some additional/special modules or a new component like DP cache, will increase the network complexity and the overall cost [34]. Moreover, it will be unrealistic to run complex detection algorithms, such as ML, DL, MTD, etc., within the switches. Usually, the SDN controller runs these algorithms. Nevertheless, switches could only perform coarse-grained attack detection and take independent actions if required.

7.1.4. Distributed SDN deployment

Distributing monitoring capability in a multi-controller environment [160–162] could also enable the scalability of an SDN-based security solution. The architecture of multi-controllers can operate on either a two-level hierarchical controller or a distributed controller system.

Hierarchical or distributed architectures face the bottleneck problem when the network size becomes large, and the applications running require advanced processing and analysis to operate their functionalities in the network. Hierarchical controller architecture can still produce bottlenecks for large networks, and the distributed controller scheme can present synchronization overheads [153]. Besides, in network environments that require diversity in SDN controllers, such as in IoT, having a backup controller for each controller may become costly, thus resulting in non-scalable and complex solutions [163]. The optimized deployment of multi-controller environments is still an unsolved challenge.

7.1.5. NFV

The emerging NFV technology can enable the scalability and flexibility factors of SDN-based security solutions since NFV deploys different virtual functions (VFs) on-demand and on the fly. In addition, NFV enables openness through vendor lock-in alleviation, which is currently a factor affecting the deployment of holistic, comprehensive, and flexible automated solutions.

All the strategies discussed can be combined to obtain a scalable and automated security solution.

7.2. Evaluation of SDN-based solutions

It is tough to find many studies with an evaluation that is both comprehensive and realistic. Various works give many results on a wide range of performance metrics, using small test networks, lacking the realism of implementation. Other works try to build a large topology using simulation tools but run into some issues due to the limitations of the environment. Without a strong evaluation, one cannot fully assess the benefit and overhead associated with a new system, so the practical applicability of a given solution usually remains in question [39]. The evaluation of critical parameters such as the traffic added by the method, the processing and memory demands, the changes requirements in the current architecture of SDN, or the level of automation of the proposed method, can help to define if the technique provides crucial benefits when deployed in a production network.

The most common metrics used to validate SDN-based security solutions for networks, particularly those based on the threat detection/mitigation approach, are accuracy, recall, F-score, FPs rate, ROC curve, and detection rate. Since these metrics are related to the classifier performance used on the threat detection, they only allow a partial assessment of the overall conditions existing in the network. Besides using the classification metrics, we can evaluate the SDN-based security solutions using other performance metrics, as described in the following.

Table 6 defines a set of parameters and metrics to evaluate the network security design using SDN effectively.

Firstly, reactive solutions are mainly evaluated based on the performance of threats detection and mitigation. In addition to the standard classification metrics, we can quantify the preprocessing effectiveness, the models' training time, the robustness on attack detection, the ratio of malicious traffic blocked, and the legitimate packet loss rate.

The flexibility parameter is also significant since integral security solutions will provide a more complete and holistic system to defend the network.

Moreover, all security proposal designs must consider the scalability component. A sophisticated security system could not be scalable, whereas a simple one could be ineffective for advanced attacks. The mechanism of security must have a trade off between effectiveness and scalability. Direct metrics to assess the scalability property of a security solution include the end-to-end communication delay, the controller CPU and memory usage, the switches TCAM usage, the throughput, and the additional traffic on the network. The analysis of the algorithms' complexity (preprocessing, detection, and mitigation algorithms) using big O notation can be helpful in the proper definition of the scalability factor of a specific solution.

The effectiveness and scalability of a security solution must not affect the quality of experience (QoE) provided to legitimate users. The QoE can be measured using the QoS of legitimate users, using surveys with automatic tools.

A crucial evaluation parameter is the degree of automation of the security solution, which measures the system's human independence. We can evaluate the ability of a solution to enable specific self-* properties. Maintaining the right balance of machine-powered security automation and human intervention will empower security professionals to effectively and efficiently carry out incident responses.

A fundamental evaluation parameter is the implementation and deployment complexity of a security solution. Too complex solutions will result in time-consuming solution deployment and locked to highly skilled developers.

Finally, the cost is a fundamental parameter that could limit the implementation of revolutionary ideas. All solution approaches need

Table 6
Evaluation parameters of an SDN-based security implementation.

Parameter	Evaluate	Metric
Detector performance	Effectiveness in threat mitigation	Detection rate, FPs rate, ROC, Recall and F-score, accuracy, confusion matrix, rate of discarded flows, training time, robustness, ratio of malicious traffic blocked, and legitimate packets loss rate.
Scope	Flexibility	The number of threats solved, and modularity.
System complexity	Scalability of the approach	End-to-end communication delay, CPU and memory usage, switches' TCAM usage, throughput, additional traffic on the network, and algorithms complexity (big O notation).
QoE	Bandwidth available to legitimate users	QoS of legitimate users.
Degree of automation	How human-independent is the system design	The percentage of the events response actions human review, self-* capabilities enabled.
Implementation complexity	Ease of solution deployment	Time required to setup tools/models/applications for a comprehensive implementation of the solution.
Costs	Additional costs to implement the security	New hardware or appliances required, energy consumption, and new skills required by network administrators.

to analyze the financial cost that the proposal could have. The requirement of additional hardware or software, the elevated energy consumption, and the network administrators' training campaigns could be costly.

It is essential to recognize the need to optimize all the parameters before indicated, which will allow an automated security solution to be scalable at a low cost. A trade off between the large-term advantages of a security solution design and its associated costs will define whether to implement this solution in a specific production network.

7.3. Open problems

SDN-like architectures can certainly automate different solutions in modern network environments. According to the Cisco Report of Global Networking Trends [164] for 2020, almost 60% of data centers use some form of SDN. Moreover, for 2021 the SDN and NFV technologies combination will be transporting near 44% of traffic within enterprise data centers. Finally, this report highlighted that the current best investment areas in networking trends are AI and security. We note that both academia and industry are working further to find automated security solutions with convergent ideas. However, these two stakeholders need to raise their level of cooperation to boost the creation of automated networks defense.

We also noted that almost all the security proposals included in this survey did not consider applying to multi-controller, multi-domain, and hybrid environments. A functional and automated security solution must recognize that new network environments manage massive traffic and devices and provide new services with different security requirements. Such network environments will require more than one controller (physical or virtual) and will operate with multiple domains. In addition, there is a need of more studies analyzing the practicality of SDN-based security solutions for partial SDN deployment, also known as hybrid networks [165,166], and design methods to be effective in such a situation. The automation of security in these contexts needs further development.

8. Challenges and future directions to network security automation

8.1. P4 enabling automation

We stated that one strategy to scale the automated security solution is to add intelligence to forwarding devices. Conventional switches and OF-based switches will require additional appliances to enable their intelligence, which will increase the cost and complexity of the solution.

P4 is a domain-specific language for programming the data plane of network devices [157]. The objective of P4 is to obtain a software-based programmable forwarding device. P4 brings significant advantages such as flexibility and expressiveness to implement sophisticated and hardware-independent packet processing algorithms. The most popular open-source controllers OpenDaylight and ONOS, are working to support P4 in their southbound interface [167,168].

P4 allows sophisticated applications implementations such as advanced traffic engineering, firewalls, or IDS/IPS solutions in the data plane. Furthermore, P4 can potentially support Blockchain data structure, achieving the security-by-design capability of this emerging technology [169]. These characteristics bring fundamental advantages to a security solution to reduce the processing burden on the controller and the reduction of DDoS attack surfaces. In summary, P4 will boost SDN, NFV, and Blockchain technologies to create more sophisticated solutions required to automate the network defense, maintaining the desired scalability, simplicity, low costs, and openness. The same purposes are attempted through the formulation of COTS hardware-based VNF, such as the VPP. We can shortly expect intelligent data planes that boost open, high-performance, and automated networks security solutions.

8.2. Orchestration for a comprehensive security solution

The SDN-based security solutions presented in this survey try to solve a single security threat without manual intervention. However, multiple threats could occur at the same time. Therefore, implementing isolated security solutions to automate network defense could result in redundant functions in many of their subsystems (such as collecting information), the execution of inconsistent tasks, or conflicts with other non-security-related network applications. The SDN controller can execute the automated security solutions in a purposeful order and verify the success of each task. Besides, integrating all network security solutions in a unique framework, one can achieve autonomy and resilience in the overall network operation. This structure is known as orchestration, which can solve rule conflicts generated by different security applications, also merge and simplify similar tasks. In short, an SDN-based security orchestrator can enable an automated, robust, and holistic defense solution with optimized resources utilization.

One example of implementing SDN-based orchestration is to use the modern architecture named Service Function Chaining (SFC) [170], which is the most promising way to process the packets in pre-described order of VNFs. SFC provides a specific network service to the user and

satisfies end-to-end QoS constraints [171]. The SFC and NFV technologies integration in SDN-based architecture can help us to provide a holistic and self-checking security framework [171,172], which has optimal resources utilization, even in multi-domain environments [173].

Intent-based networking (IBN) is another tool enabling SDN-based security orchestration, which can continuously enforce our defense policies while optimizing network resources. The purpose of IBN is to introduce a deeper level of intelligence and intended state insights to networking, which augments the automation capabilities of SDN [164, 174]. IBN allows network administrators to tell the SDN-controller what outcome they want (business intent) instead of coding and executing individual tasks, which can be more challenging, time-consuming, and error-prone. The SDN-controller will perform resource allocation and policy enforcement to achieve the objective. Thus, IBN simplifies the implementation of SDN-based security solutions and their remediation and assurance. Moreover, IBN can integrate SDN, NFV, and AI to provide fully automated solutions considering the network state, the dynamic state of all users and applications, the service performance and security requirements, and the business needs across all domains. Although several studies, such as the presented in [175], have focused their research on IBN-enabled attack protection for SDN networks, more efforts are needed.

9. Conclusion

There are multiple approaches to solve the security problem in SDN-based networks. In this document, we survey them with the vision of automation of defense mechanisms to protect networks using SDN technology. The nature of programmable and centralized control of SDN enables this technology to generate scalable, optimized, intelligent, and automated security solutions to defend networks. We categorized the SDN-based security reviewed into two classes: reactive defense based on attack detection and proactive protection based on in-line network reconfiguration. The reactive solutions, which use ML, DL, and RL methods, mitigates sophisticated attacks. The state-of-the-art proactive security solutions implement mechanisms like MTD defense, Cyber Deception, Network Slicing, and Blockchains combined with NFV capabilities and intelligent algorithms. Strategies of Reinforcement Learning, Network Slicing, and Blockchain-assisted are still facing fundamental challenges, such as scalability for their adoption in production networks.

We found that reactive and proactive mechanisms can enable different levels of automation (self-* capabilities) and even can be combined to achieve automated security solutions. Nevertheless, it is important to identify the complexity needed to implement them, translated to deployment costs. We provided a ranking of the security solutions that reveal the automation level enabled by each strategy and the complexity related to their implementation.

Also, we discussed the scalability factor and the parameters needed for the comprehensive assessment of the security solution proposals. With this evaluation, we can identify the advantages, limitations, and associated costs, of a security solution and decide whether to implement it in specific production networks.

We hope that our discussions and exploration give readers a profound grasp of how the SDN paradigm is pushing the creation of innovative and automated security solutions for next-generation networks, which we believe is the ultimate goal of SDN technology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially supported by FRIDA (Fondo Regional para la Innovación Digital en America Latina *y* el Caribe); the project "Red temática Ciencia *y* Tecnología para el desarrollo (CYTED) 519RT0580" by the Ibero-American Science and Technology Program for development CYTED; a 2020 Seed Fund award from Tecnologico de Monterrey & CITRIS and the Banatao Institute at the University of California, USA; the School of Engineering and Sciences at Tecnologico de Monterrey; and the Telecommunications Research Group.

References

[1] R. Vilalta, R. Ciungu, A. Mayoral, R. Casellas, R. Martinez, D. Pubill, J. Serra, R. Munoz, C. Verikoukis, Improving security in internet of things with software defined networking, in: 2016 IEEE Global Communications Conference, GLOBECOM, IEEE, 2016, pp. 1–6.
[2] B. Alzahrani, N. Fotiou, Enhancing internet of things security using software-defined networking, J. Syst. Archit. 110 (2020) 101779.
[3] I. Farris, T. Taleb, Y. Khettab, J. Song, A survey on emerging SDN and NFV security mechanisms for IoT systems, IEEE Commun. Surv. Tutor. 21 (1) (2019) 812–837.
[4] P.B. Pajila, E.G. Julie, Detection of DDoS attack using SDN in IoT: A survey, in: Intelligent Communication Technologies and Virtual Mobile Networks, Springer, 2019, pp. 438–452.
[5] A.A. Abbasi, A. Abbasi, S. Shamshirband, A.T. Chronopoulos, V. Persico, A. Pescape, Software-defined cloud computing: A systematic review on latest trends and developments, IEEE Access 7 (2019) 93294–93314.
[6] Q. Long, Y. Chen, H. Zhang, X. Lei, Software defined 5G and 6G networks: A survey, Mob. Netw. Appl. (2019) 1–21.
[7] A.A. Barakabitze, A. Ahmad, R. Mijumbi, A. Hines, 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges, Comput. Netw. 167 (2020) 106984.
[8] O.S. Al-Heety, Z. Zakaria, M. Ismail, M.M. Shakir, S. Alani, H. Alsariera, A comprehensive survey: Benefits, services, recent works, challenges, security, and use cases for SDN-VANET, IEEE Access 8 (2020) 91028–91047.
[9] W.B. Jaballah, M. Conti, C. Lal, Security and design requirements for software-defined VANETs, Comput. Netw. 169 (2020) 107099.
[10] The mad dash to find a cybersecurity force, 2021, Available at https://www.nytimes.com/2018/11/07/business/the-mad-dash-to-find-a-cybersecurity-force.html (Accessed 18 May 2021).
[11] The cloud talent drought continues (and is even larger than you thought), 2021, Available at https://www.forbes.com/sites/emilsayegh/2020/03/02/the-2020-cloud-talent-drought-is-even-larger-than-you-thought/?sh=cf99e8658c0b (Accessed 18 May 2021).
[12] CISCO, Securing What's Now and What's Next, Tech. Rep., CISCO, 2020, Available at https://www.cisco.com/c/dam/en/us/products/collateral/security/2020-ciso-benchmark-cybersecurity-series-feb-2020.pdf (Accessed 18 May 2021).
[13] J. Oltsik, The Shift to Incident Response Automation and Orchestration, Technical Report, Enterprise Strategy Group, 2016.
[14] A. Alshamrani, S. Myneni, A. Chowdhary, D. Huang, A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities, IEEE Commun. Surv. Tutor. 21 (2) (2019) 1851–1877.
[15] S. Quintero-Bonilla, A. Martín del Rey, A new proposal on the advanced persistent threat: A survey, Appl. Sci. 10 (11) (2020) 3874.
[16] DDoS attacks in Q4 2020, 2021, Available at https://securelist.com/ddos-attacks-in-q4-2020/100650/ (Accessed 18 May 2021).
[17] R. Smith-Bingham, This is what CEOs around the world see as the biggest risks to business, Tech. Rep., World Economic Forum, 2019, Available at https://www.weforum.org/agenda/2019/10/risks-to-doing-business-2019-developing-developed/ (Accessed 18 May 2021).
[18] R. Hat, What is Security Automation? Red Hat, 2020, Available at https://www.redhat.com/en/topics/automation/what-is-security-automation (Accessed 18 May 2021).
[19] Top security and risk management trends, 2021, Available at https://www.gartner.com/document/3981492?ref=solrAll&refval=287935324 (Accessed 18 May 2021).
[20] E. Gelenbe, J. Domanska, P. Fröhlich, M.P. Nowak, S. Nowak, Self-aware networks that optimize security, qos, and energy, Proc. IEEE (2020).
[21] S. Ayoubi, N. Limam, M.A. Salahuddin, N. Shahriar, R. Boutaba, F. Estrada-Solano, O.M. Caicedo, Machine learning for cognitive network management, IEEE Commun. Mag. 56 (1) (2018) 158–165.
[22] Q.H. Mahmoud, Cognitive networks, John Wiley& Sons Ltd, 2007, pp. 57–71.
[23] J. Xie, F. Richard Yu, T. Huang, R. Xie, J. Liu, C. Wang, Y. Liu, A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges, IEEE Commun. Surv. Tutor. 21 (1) (2019) 393–430.

[24] N. Sultana, N. Chilamkurti, W. Peng, R. Alhadad, Survey on SDN based network intrusion detection system using machine learning approaches, Peer-To-Peer Netw. Appl. 12 (2) (2019) 493–501.
[25] M. Latah, L. Toker, Artificial intelligence enabled software-defined networking: A comprehensive overview, IET Netw. 8 (2) (2019) 79–99.
[26] Y. Zhao, Y. Li, X. Zhang, G. Geng, W. Zhang, Y. Sun, A survey of networking applications applying the software defined networking concept based on machine learning, IEEE Access 7 (2019) 95397–95417.
[27] A.A. Gebremariam, M. Usman, M. Qaraqe, Applications of artificial intelligence and machine learning in the area of SDN and NFV: A survey, in: 16th International Multi-Conference on Systems, Signals and Devices, SSD 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 545–549.
[28] Y. Liu, B. Zhao, P. Zhao, P. Fan, H. Liu, A survey: Typical security issues of software-defined networking, China Commun. 16 (7) (2019) 13–31.
[29] J.C.C. Chica, J.C. Imbachi, J.F.B. Vega, Security in SDN: A comprehensive survey, J. Netw. Comput. Appl. 159 (2020) 102595.
[30] I. Ahmad, S. Namal, M. Ylianttila, A. Gurtov, Security in software defined networks: A survey, IEEE Commun. Surv. Tutor. 17 (4) (2015) 2317–2346.
[31] T. Jafarian, M. Masdari, A. Ghaffari, K. Majidzadeh, A survey and classification of the security anomaly detection mechanisms in software defined networks, Cluster Comput. 24 (2) (2021) 1235–1253.
[32] N. Dayal, P. Maity, S. Srivastava, R. Khondoker, Research trends in security and ddos in SDN, Secur. Commun. Netw. 9 (18) (2016) 6386–6411.
[33] K. Kalkan, G. Gur, F. Alagoz, Defense mechanisms against ddos attacks in SDN environment, IEEE Commun. Mag. 55 (9) (2017) 175–179.
[34] M. Imran, M.H. Durad, F.A. Khan, A. Derhab, Toward an optimal solution against denial of service attacks in software defined networks, Future Gener. Comput. Syst. 92 (2019) 444–453.
[35] M.A. Aladaileh, M. Anbar, I.H. Hasbullah, Y.-W. Chong, Y.K. Sanjalawe, Detection techniques of distributed denial of service attacks on software-defined networking controller–a review, IEEE Access 8 (2020) 143985–143995.
[36] J.A. Herrera, J.E. Camargo, A survey on machine learning applications for software defined network security, in: International Conference on Applied Cryptography and Network Security, Springer, 2019, pp. 70–93.
[37] P. Wang, L.T. Yang, X. Nie, Z. Ren, J. Li, L. Kuang, Data-driven software defined network attack detection : State-of-the-art and perspectives, Inform. Sci. 513 (2020) 65–83.
[38] Y. Hande, A. Muddana, A survey on intrusion detection system for software defined networks (SDN), Int. J. Bus. Data Commun. Netw. 16 (1) (2020) 28–47.
[39] O. Yurekten, M. Demirci, SDN-Based cyber defense: A survey, Future Gener. Comput. Syst. 115 (2021) 126–149.
[40] R. Swami, M. Dave, V. Ranga, Software-defined networking-based DDoS defense mechanisms, ACM Comput. Surv. 52 (2) (2019).
[41] M.P. Singh, A. Bhandari, New-flow based ddos attacks in SDN: Taxonomy, rationales, and research challenges, Comput. Commun. 154 (2020) 509–527.
[42] K.S. Sahoo, S.K. Panda, S. Sahoo, B. Sahoo, R. Dash, Toward secure software-defined networks against distributed denial of service attack, J. Supercomput. 75 (8) (2019) 4829–4874.
[43] S. Scott-Hayward, S. Natarajan, S. Sezer, A survey of security in software defined networks, IEEE Commun. Surv. Tutor. 18 (1) (2016) 623–654.
[44] N. Mckeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, OpenFlow: Enabling Innovation in Campus Networks, in: ACM SIGCOMM Comput. Commun. 2008, pp. 69–74.
[45] P. Goransson, C. Black, T. Culver, Software Defined Networks: A Comprehensive Approach, Morgan Kaufmann, 2016.
[46] W. Stallings, Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Addison-Wesley Professional, 2015.
[47] K. Bhushan, B.B. Gupta, Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment, J. Ambient Intell. Humaniz. Comput. 10 (5) (2019) 1985–1997.
[48] E.F. Kfoury, J. Crichigno, E. Bou-Harb, An exhaustive survey on P4 programmable data plane switches: Taxonomy, applications, challenges, and future trends, IEEE Access 9 (2021) 87094–87155.
[49] I. Sharafaldin, A.H. Lashkari, S. Hakak, A.A. Ghorbani, Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy, in: 2019 International Carnahan Conference on Security Technology, ICCST, IEEE, 2019, pp. 1–8.
[50] S.M. Mousavi, M. St-Hilaire, Early detection of ddos attacks against SDN controllers, in: 2015 International Conference on Computing, Networking and Communications, ICNC 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 77–81.
[51] K.S. Sahoo, D. Puthal, M. Tiwary, J. Rodrigues, B. Sahoo, R. Dash, An early detection of low rate ddos attack to SDN based data center networks using information distance metrics, Future Gener. Comput. Syst. 89 (2018) 685–697.
[52] M. Conti, C. Lal, R. Mohammadi, U. Rawat, Lightweight solutions to counter ddos attacks in software defined networking, Wirel. Netw. 25 (5) (2019) 2751–2768.
[53] R. Wang, Z. Jia, L. Ju, An entropy-based distributed ddos detection mechanism in software-defined networking, in: Proceedings - 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2015, 1, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 310–317.
[54] K. Giotis, C. Argyropoulos, G. Androulidakis, D. Kalogeras, V. Maglaris, Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments, Comput. Netw. 62 (2014) 122–136.
[55] M.V. De Assis, M.P. Novaes, C.B. Zerbini, L.F. Carvalho, T. Abrao, M.L. Proenca, Fast defense system against attacks in software defined networks, IEEE Access 6 (2018) 69620–69639.
[56] C.B. Zerbini, L.F. Carvalho, T. Abrão, M.L. Proença, Wavelet against random forest for anomaly mitigation in software-defined networking, Appl. Soft Comput. 80 (2019) 138–153.
[57] J. Ye, X. Cheng, J. Zhu, L. Feng, L. Song, A ddos attack detection method based on SVM in software defined network, Secur. Commun. Netw. 2018 (2018).
[58] M. Myint Oo, S. Kamolphiwong, T. Kamolphiwong, S. Vasupongayya, Advanced support vector machine-(ASVM-) based detection for distributed denial of service (DDoS) attack on software defined networking (SDN), J. Comput. Netw. Commun. 2019 (2019).
[59] J. Cui, M. Wang, Y. Luo, H. Zhong, DDoS detection and defense mechanism based on cognitive-inspired computing in SDN, Future Gener. Comput. Syst. 97 (2019) 275–283.
[60] H. Peng, Z. Sun, X. Zhao, S. Tan, Z. Sun, A detection method for anomaly flow in software defined network, IEEE Access 6 (2018) 27809–27817.
[61] L. Zhu, X. Tang, M. Shen, X. Du, M. Guizani, Privacy-preserving ddos attack detection using cross-domain traffic in software defined networks, IEEE J. Sel. Areas Commun. 36 (3) (2018) 628–643.
[62] Z. Liu, Y. He, W. Wang, B. Zhang, DDoS attack detection scheme based on entropy and PSO-BP neural network in SDN, China Commun. 16 (7) (2019) 144–155.
[63] H. Pillutla, A. Arjunan, Fuzzy self organizing maps-based ddos mitigation mechanism for software defined networking in cloud computing, J. Ambient Intell. Humaniz. Comput. 10 (4) (2019) 1547–1559.
[64] Z. Fan, Y. Xiao, A. Nayak, C. Tan, An improved network security situation assessment approach in software defined networks, Peer-To-Peer Netw. Appl. 12 (2) (2019) 295–309.
[65] W. Wang, X. Ke, L. Wang, A HMM-R approach to detect L-DDoS attack adaptively on sdn controller, Future Internet 10 (9) (2018) 83.
[66] Y. Xiao, Z.-J. Fan, A. Nayak, C.-X. Tan, Discovery method for distributed denial-of-service attack behavior in SDNs using a feature-pattern graph model, Front. Inf. Technol. Electron. Eng. 20 (9) (2019) 1195–1208.
[67] M.V. De Assis, A.H. Hamamoto, T. Abrao, M.L. Proenca, A game theoretical based system using holt-winters and genetic algorithm with fuzzy logic for DoS/DDoS mitigation on SDN networks, IEEE Access 5 (2017) 9485–9496.
[68] Y. Guo, F. Miao, L. Zhang, Y. Wang, CATH: An effective method for detecting denial-of-service attacks in software defined networks, Sci. China Inf. Sci. 62 (3) (2019) 1–15.
[69] X. Xu, S. Wang, Y. Li, Identification and predication of network attack patterns in software-defined networking, Peer-To-Peer Netw. Appl. 12 (2) (2019) 337–347.
[70] K. Alrawashdeh, C. Purdy, Toward an online anomaly intrusion detection system based on deep learning, in: Proc. 15th IEEE Int. Conf. Mach. Learn. Appl., Institute of Electrical and Electronics Engineers (IEEE), 2017, pp. 195–200.
[71] Q. Niyaz, W. Sun, A.Y. Javaid, A deep learning based ddos detection system in software-defined networking (SDN), EAI Endorsed Trans. Secur. Saf. 4 (12) (2017).
[72] N. Shone, T.N. Ngoc, V.D. Phai, Q. Shi, A deep learning approach to network intrusion detection, IEEE Trans. Emerg. Top. Comput. Intell. 2 (1) (2018) 41–50.
[73] Y. Jia, F. Zhong, A. Alrawais, B. Gong, X. Cheng, FlowGuard: An intelligent edge defense mechanism against IoT DDoS attacks, IEEE Internet Things J. 7 (10) (2020) 9552–9562.
[74] V. Punitha, C. Mala, N. Rajagopalan, A novel deep learning model for detection of denial of service attacks in HTTP traffic over internet, Int. J. Ad Hoc Ubiquitous Comput. 33 (4) (2020) 240–256.
[75] C. Li, Y. Wu, X. Yuan, Z. Sun, W. Wang, X. Li, L. Gong, Detection and defense of ddos attack–based on deep learning in OpenFlow-based SDN, Int. J. Commun. Syst. 31 (5) (2018) e3497.
[76] X. Liang, T. Znati, A long short-term memory enabled framework for ddos detection, in: 2019 IEEE Global Communications Conference, GLOBECOM, IEEE, 2019, pp. 1–6.
[77] T.A. Tang, L. Mhamdi, D. McLernon, S.A.R. Zaidi, M. Ghogho, F. El Moussa, DeepIDS: Deep learning approach for intrusion detection in software defined networking, Electronics 9 (9) (2020) 1533.
[78] M.V. Assis, L.F. Carvalho, J. Lloret, M.L. Proença Jr., A GRU deep learning system against attacks in software defined networks, J. Netw. Comput. Appl. 177 (2021) 102942.
[79] Gartner, Hype cycle for data science and machine learning, 2019, https://www.gartner.com/document/3955984?ref=lib.
[80] J.a. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, A. Bouchachia, A survey on concept drift adaptation, ACM Comput. Surv. 46 (4) (2014) 1–37.
[81] T. Jafarian, M. Masdari, A. Ghaffari, K. Majidzadeh, A survey and classification of the security anomaly detection mechanisms in software defined networks, Cluster Comput. 24 (2) (2021) 1235–1253.

[82] Z. Wang, C. Zhou, Y. Yu, X. Shi, X. Yin, J. Yao, Fast detection of heavy hitters in software defined networking using an adaptive and learning method, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11065 LNCS, Springer Verlag, 2018, pp. 44–55.
[83] M. Akbanov, V.G. Vassilakis, M.D. Logothetis, Ransomware detection and mitigation using software-defined networking: The case of WannaCry, Comput. Electr. Eng. 76 (2019) 111–121.
[84] S. Garg, K. Kaur, J. Rodrigues, J. Rodrigues, Hybrid deep-learning-based anomaly detection scheme for suspicious flow detection in SDN: A social multimedia perspective, IEEE Trans. Multimed. 21 (3) (2019) 566–578.
[85] Y. Cui, L. Yan, S. Li, H. Xing, W. Pan, J. Zhu, X. Zheng, SD-Anti-DDoS: Fast and efficient DDoS defense in software-defined networks, J. Netw. Comput. Appl. 68 (2016) 65–79.
[86] L. Dridi, M.F. Zhani, A holistic approach to mitigating DoS attacks in SDN networks, Int. J. Netw. Manag. 28 (1) (2018) e1996.
[87] H. Wang, L. Xu, G. Gu, FloodGuard: A DoS Attack prevention extension in software-defined networks, in: Proceedings of the International Conference on Dependable Systems and Networks, Vol. 2015-Septe, IEEE Computer Society, 2015, pp. 239–250.
[88] M. Kuerban, Y. Tian, Q. Yang, Y. Jia, B. Huebert, D. Poss, Flowsec: DOS attack mitigation strategy on SDN controller, in: 2016 IEEE International Conference on Networking, Architecture and Storage, NAS, IEEE, 2016, pp. 1–2.
[89] P. Krishnan, S. Duttagupta, K. Achuthan, VARMAN: Multi-plane security framework for software defined networks, Comput. Commun. 148 (2019) 215–239.
[90] M. Zolotukhin, S. Kumar, T. Hämäläinen, Reinforcement learning for attack mitigation in SDN-enabled networks, in: 2020 6th IEEE Conference on Network Softwarization, NetSoft, IEEE, 2020, pp. 282–286.
[91] I. Akbari, E. Tahoun, M.A. Salahuddin, N. Limam, R. Boutaba, ATMoS: Autonomous threat mitigation in SDN using reinforcement learning, in: NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, IEEE, 2020, pp. 1–9.
[92] Y. Liu, M. Dong, K. Ota, J. Li, J. Wu, Deep reinforcement learning based smart mitigation of ddos flooding in software-defined networks, in: 2018 IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks, CAMAD, IEEE, 2018, pp. 1–6.
[93] A. VishnuPriya, Reinforcement learning-based DoS mitigation in software defined networks, in: Lecture Notes in Electrical Engineering, vol. 500, Springer Verlag, 2019, pp. 393–401.
[94] L.S. Sampaio, P.H. Faustini, A.S. Silva, L.Z. Granville, A. Schaeffer-Filho, Using NFV and reinforcement learning for anomalies detection and mitigation in SDN, in: 2018 IEEE Symposium on Computers and Communications, ISCC, IEEE, 2018, pp. 00432–00437.
[95] C.H. Huang, T.H. Lee, L. huang Chang, J.R. Lin, G. Horng, Adversarial attacks on SDN-based deep learning IDS system, in: Lecture Notes in Electrical Engineering, vol. 513, Springer Verlag, 2019, pp. 181–191.
[96] Y. Han, B.I. Rubinstein, T. Abraham, T. Alpcan, O. De Vel, S. Erfani, D. Hubczenko, C. Leckie, P. Montague, Reinforcement learning for autonomous defence in software-defined networking, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11199 LNCS, Springer Verlag, 2018, pp. 145–165.
[97] A. AlEroud, G. Karabatis, SDN-Gan: Generative adversarial deep NNs for synthesizing cyber attacks on software defined networks, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11878 LNCS, Springer, 2020, pp. 211–220.
[98] Gartner, Top 10 strategic technology trends for 2020: AI security, 2020, Available at https://www.gartner.com/document/3981944?ref=lib (Accessed 28 April 2020).
[99] J. Zheng, A. Namin, A survey on the moving target defense strategies: An architectural perspective, J. Comput. Sci. Tech. 34 (1) (2019) 207–233.
[100] S. Sengupta, A. Chowdhary, A. Sabur, A. Alshamrani, D. Huang, S. Kambhampati, A survey of moving target defenses for network security, IEEE Commun. Surv. Tutor. 22 (3) (2020) 1909–1941.
[101] J.H. Jafarian, E. Al-Shaer, Q. Duan, Adversary-aware IP address randomization for proactive agility against sophisticated attackers, in: Proceedings - IEEE INFOCOM, Vol. 26, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 738–746.
[102] Y. Shi, H. Zhang, J. Wang, F. Xiao, J. Huang, D. Zha, H. Hu, F. Yan, B. Zhao, CHAOS: AN SDN-based moving target defense system, Secur. Commun. Netw. 2017 (2017).
[103] J.H. Jafarian, E. Al-Shaer, Q. Duan, Formal approach for route agility against persistent attackers, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 8134 LNCS, Springer, Berlin, Heidelberg, 2013, pp. 237–254.
[104] Z. Zhao, D. Gong, B. Lu, F. Liu, C. Zhang, SDN-Based double hopping communication against sniffer attack, Math. Probl. Eng. 2016 (2016).
[105] J. Liu, H. Zhang, Z. Guo, A defense mechanism of random routing mutation in SDN, IEICE Trans. Inf. Syst. E100D (5) (2017) 1046–1054.
[106] M. Pattaranantakul, R. He, Q. Song, Z. Zhang, A. Meddahi, NFV Security survey: From use case driven threat analysis to state-of-the-art countermeasures, IEEE Commun. Surv. Tutor. 20 (4) (2018) 3330–3368.
[107] A. Farshin, S. Sharifian, A modified knowledge-based ant colony algorithm for virtual machine placement and simultaneous routing of NFV in distributed cloud architecture, J. Supercomput. 75 (8) (2019) 5520–5550.
[108] F. Gillani, E. Al-Shaer, S. Lo, Q. Duan, M. Ammar, E. Zegura, Agile virtualized infrastructure to proactively defend against cyber attacks, in: Proceedings - IEEE INFOCOM, 26, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 729–737.
[109] M. Zolotukhin, T. Hämäläinen, On artificial intelligent malware tolerant networking for IoT, in: 2018 IEEE Conference on Network Function Virtualization and Software Defined Networks, NFV-SDN, IEEE, 2018, pp. 1–6.
[110] G. Gardikis, K. Tzoulas, K. Tripolitis, A. Bartzas, S. Costicoglou, A. Lioy, B. Gaston, C. Fernandez, C. Davila, A. Litke, et al., SHIELD: A Novel NFV-based cybersecurity framework, in: 2017 IEEE Conference on Network Softwarization, NetSoft, IEEE, 2017, pp. 1–6.
[111] C.C. Liu, B.S. Huang, C.W. Tseng, Y.T. Yang, L.D. Chou, SDN/NFV-based Moving target ddos defense mechanism, in: Advances in Intelligent Systems and Computing, 843, Springer Verlag, 2019, pp. 548–556.
[112] A. Aydeger, N. Saputro, K. Akkaya, A moving target defense and network forensics framework for ISP networks using SDN and NFV, Future Gener. Comput. Syst. 94 (2019) 496–509.
[113] M. Rawski, Network topology mutation as moving target defense for corporate networks, Int. J. Electron. Telecommun. 65 (4) (2019) 571–577.
[114] P.P. Ray, N. Kumar, SDN/NFV Architectures for edge-cloud oriented IoT: A systematic review, Comput. Commun. 169 (2021) 129–153.
[115] L. Linguaglossa, D. Rossi, S. Pontarelli, D. Barach, D. Marjon, P. Pfister, High-speed data plane and network functions virtualization by vectorizing packet processing, Comput. Netw. 149 (2019) 187–199.
[116] S. Han, K. Jang, A. Panda, S. Palkar, D. Han, S. Ratnasamy, SoftNIC: A software NIC to augment hardware, Tech. Rep. UCB/EECS-2015-155, EECS Department, University of California, Berkeley, 2015.
[117] J. Yan, L. Tang, T. Li, G. Lv, W. Quan, H. Yang, PPB: A path-based packet batcher to accelerate vector packet processor, in: 2020 15th International Conference on Computer Science & Education, ICCSE, IEEE, 2020, pp. 681–686.
[118] D.B. Rawat, N. Sapavath, M. Song, Performance evaluation of deception system for deceiving cyber adversaries in adaptive virtualized wireless networks, in: Proceedings of the 4th ACM/IEEE Symposium on Edge Computing, Association for Computing Machinery, 2019, pp. 401–406.
[119] S. Sugrim, S. Venkatesan, J.A. Youzwak, C.Y.J. Chiang, R. Chadha, M. Albanese, H. Cam, Measuring the effectiveness of network deception, in: 2018 IEEE International Conference on Intelligence and Security Informatics, ISI 2018, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 142–147.
[120] G. Sadowski, R. Kau, Improve your threat detection function with deception technologies, 2020, Available at https://www.gartner.com/document/3905698?ref=solrAll&refval=247943014 (Accessed 28 April 2020).
[121] S. Achleitner, T. La Porta, P. McDaniel, S. Sugrim, S.V. Krishnamurthy, R. Chadha, Cyber deception: Virtual networks to defend insider reconnaissance, in: Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats, Association for Computing Machinery, 2016, pp. 57–68.
[122] C.-Y.J. Chiang, Y.M. Gottlieb, S.J. Sugrim, R. Chadha, C. Serban, A. Poylisher, L.M. Marvel, J. Santos, ACyDS: An adaptive cyber deception system, in: MILCOM 2016-2016 IEEE Military Communications Conference, IEEE, 2016, pp. 800–805.
[123] M.P. Stoecklin, J. Zhang, F. Araujo, T. Taylor, Dressed up: Baiting attackers through endpoint service projection, in: Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, Association for Computing Machinery, 2018, pp. 23–28.
[124] G. Bernieri, M. Conti, F. Pascucci, MimePot: A model-based honeypot for industrial control networks, in: Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, Vol. 2019-Octob, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 433–438.
[125] D. Mao, S. Zhang, L. Zhang, Y. Feng, Game theory based dynamic defense mechanism for SDN, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11806 LNCS, Springer Verlag, 2019, pp. 290–303.
[126] Q. Zhao, C. Zhang, Z. Zhao, A decoy chain deployment method based on SDN and NFV against penetration attack, in: H. Wang (Ed.), PLoS One 12 (12) (2017) e0189095.
[127] N. Soule, P. Pal, S. Clark, B. Krisler, A. Macera, Enabling defensive deception in distributed system environments, in: Proceedings - 2016 Resilience Week, RWS 2016, Institute of Electrical and Electronics Engineers Inc., 2016, pp. 73–76.
[128] J. Kelly, M. DeLaus, E. Hemberg, U.-M. O'Reilly, Adversarially adapting deceptive views and reconnaissance scans on a software defined network, in: 2019 IFIP/IEEE Symposium on Integrated Network and Service Management, IM, IEEE, 2019, pp. 49–54.
[129] F. Kurtz, C. Bektas, N. Dorsch, C. Wietfeld, Network slicing for critical communications in shared 5G infrastructures - an empirical evaluation, in: 2018 4th IEEE Conference on Network Softwarization and Workshops, NetSoft 2018, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 262–266.

[130] Y. Khettab, M. Bagaa, D.L.C. Dutra, T. Taleb, N. Toumi, Virtual security as a service for 5G verticals, in: IEEE Wireless Communications and Networking Conference, WCNC, Vol. 2018-April, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 1–6.
[131] J. Luo, Q. Chen, F.R. Yu, L. Tang, Blockchain-enabled software-defined industrial internet of things with deep reinforcement learning, IEEE Internet Things J. 7 (6) (2020) 5466–5480.
[132] P. Fernando, J. Wei, Blockchain-powered software defined network-enabled networking infrastructure for cloud management, in: 2020 IEEE 17th Annual Consumer Communications & Networking Conference, CCNC, IEEE, 2020, pp. 1–6.
[133] L. Xie, Y. Ding, H. Yang, X. Wang, Blockchain-based secure and trustworthy internet of things in SDN-enabled 5G-VANETs, IEEE Access 7 (2019) 56656–56666.
[134] J. Gao, K.O.-B.O. Agyekum, E.B. Sifah, K.N. Acheampong, Q. Xia, X. Du, M. Guizani, H. Xia, A blockchain-SDN-enabled internet of vehicles environment for fog computing and 5G networks, IEEE Internet Things J. 7 (5) (2019) 4278–4291.
[135] D.B. Rawat, Fusion of software defined networking, edge computing, and blockchain technology for wireless network virtualization, IEEE Commun. Mag. 57 (10) (2019) 50–55.
[136] W. Li, W. Meng, Z. Liu, M.-H. Au, Towards blockchain-based software-defined networking: Security challenges and solutions, IEICE Trans. Inf. Syst. 103 (2) (2020) 196–203.
[137] T. Alharbi, Deployment of blockchain technology in software defined networks: A survey, IEEE Access 8 (2020) 9146–9156.
[138] A. Muthanna, A. A Ateya, A. Khakimov, I. Gudkova, A. Abuarqoub, K. Samouylov, A. Koucheryavy, Secure and reliable IoT networks using fog computing with software-defined networking and blockchain, J. Sens. Act. Netw. 8 (1) (2019) 15.
[139] N. Giri, R. Jaisinghani, R. Kriplani, T. Ramrakhyani, V. Bhatia, Distributed denial of service (ddos) mitigation in software defined network using blockchain, in: 2019 Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), I-SMAC, IEEE, 2019, pp. 673–678.
[140] M. Azab, R.R. Ergawy, E.M. Ghourab, A. Mokhtar, M. Rizk, Towards blockchain-based multi-controller managed switching for trustworthy SDN operation, in: 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON, IEEE, 2019, pp. 0991–0998.
[141] A. Bose, G.S. Aujla, M. Singh, N. Kumar, H. Cao, Blockchain as a service for software defined networks: A denial of service attack perspective, in: 2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress, DASC/PiCom/CBDCom/CyberSciTech, IEEE, 2019, pp. 901–906.
[142] P.K. Sharma, S. Singh, Y.-S. Jeong, J.H. Park, Distblocknet: A distributed blockchains-based secure SDN architecture for IoT networks, IEEE Commun. Mag. 55 (9) (2017) 78–85.
[143] Q. Shafi, A. Basit, Ddos botnet prevention using blockchain in software defined internet of things, in: 2019 16th International Bhurban Conference on Applied Sciences and Technology, IBCAST, IEEE, 2019, pp. 624–628.
[144] G.S. Aujla, M. Singh, A. Bose, N. Kumar, G. Han, R. Buyya, BlockSDN: Blockchain-as-a-service for software defined networking in smart city applications, IEEE Netw. 34 (2) (2020) 83–91.
[145] G.A.F. Rebello, G.F. Camilo, L.G. Silva, L.C. Guimarães, L.A.C. de Souza, I.D. Alvarenga, O.C.M. Duarte, Providing a sliced, secure, and isolated software infrastructure of virtual functions through blockchain technology, in: 2019 IEEE 20th International Conference on High Performance Switching and Routing, HPSR, IEEE, 2019, pp. 1–6.
[146] V.H. Dixit, S. Kyung, Z. Zhao, A. Doupé, Y. Shoshitaishvili, G.-J. Ahn, Challenges and preparedness of SDN-based firewalls, in: Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, Association for Computing Machinery, 2018, pp. 33–38.
[147] A. Tsuchiya, F. Fraile, I. Koshijima, A. Ortiz, R. Poler, Software defined networking firewall for industry 4.0 manufacturing systems, J. Ind. Eng. Manag. (JIEM) 11 (2) (2018) 318–333.
[148] A. Kumari, R. Gupta, S. Tanwar, N. Kumar, A taxonomy of blockchain-enabled softwarization for secure UAV network, Comput. Commun. 161 (2020) 304–323.
[149] J. Zheng, Q. Li, G. Gu, J. Cao, D.K. Yau, J. Wu, Realtime DDoS defense using COTS SDN switches via adaptive correlation analysis, IEEE Trans. Inf. Forensics Secur. 13 (7) (2018) 1838–1853.
[150] T. Dargahi, A. Caponi, M. Ambrosin, G. Bianchi, M. Conti, A survey on the security of stateful SDN data planes, IEEE Commun. Surv. Tutor. 19 (3) (2017) 1701–1725.
[151] M. Ambrosin, M. Conti, F. De Gaspari, R. Poovendran, LineSwitch: Tackling control plane saturation attacks in software-defined networking, IEEE/ACM Trans. Netw. 25 (2) (2017) 1206–1219.
[152] J. Boite, P.-A. Nardin, F. Rebecchi, M. Bouet, V. Conan, Statesec: Stateful monitoring for ddos protection in software defined networks, in: 2017 IEEE Conference on Network Softwarization, NetSoft, IEEE, 2017, pp. 1–9.
[153] T.V. Phan, N.K. Bao, M. Park, Distributed-SOM: A novel performance bottleneck handler for large-sized software-defined networks under flooding attacks, J. Netw. Comput. Appl. 91 (2017) 14–25.
[154] M. Caprolu, S. Raponi, R. Di Pietro, FORTRESS: AN efficient and distributed firewall for stateful data plane SDN, Secur. Commun. Netw. 2019 (2019).
[155] J. Xing, W. Wu, A. Chen, Architecting programmable data plane defenses into the network with fastflex, in: Proceedings of the 18th ACM Workshop on Hot Topics in Networks, Association for Computing Machinery, 2019 pp. 161–169.
[156] M. Zhang, G. Li, S. Wang, C. Liu, A. Chen, H. Hu, G. Gu, Q. Li, M. Xu, J. Wu, Poseidon: Mitigating volumetric ddos attacks with programmable switches, in: Proceedings of NDSS, 2020.
[157] P. Consortium, P4 language specification, 2020, Available at https://p4.org/p4-spec/docs/P4-16-v1.2.1.html, (Accessed 15 June 2020).
[158] K. Kalkan, G. Gur, F. Alagoz, Sdnscore: A statistical defense mechanism against ddos attacks in sdn environment, in: Proceedings - IEEE Symposium on Computers and Communications, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 669–675.
[159] B. Han, X. Yang, Z. Sun, J. Huang, J. Su, OverWatch: A Cross-plane ddos attack defense framework with collaborative intelligence in SDN, Secur. Commun. Netw. 2018 (2018).
[160] S. Lee, J. Kim, S. Shin, P. Porras, V. Yegneswaran, Athena: A framework for scalable anomaly detection in software-defined networks, in: Proceedings - 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2017, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 249–260.
[161] T. Wang, H. Chen, G. Cheng, Y. Lu, Sdnmanager: A safeguard architecture for SDN dos attacks based on bandwidth prediction, Secur. Commun. Netw. 2018 (2018).
[162] L. Fawcett, S. Scott-Hayward, M. Broadbent, A. Wright, N. Race, Tennison: A distributed SDN framework for scalable network security, IEEE J. Sel. Areas Commun. 36 (12) (2018) 2805–2818.
[163] I. Alam, K. Sharif, F. Li, Z. Latif, M. Karim, S. Biswas, B. Nour, Y. Wang, A survey of network virtualization techniques for internet of things using SDN and NFV, ACM Comput. Surv. 53 (2) (2020) 1–40.
[164] Cisco, 2020 Global Networking Trends Report, Tech. Rep., Cisco, 2020, Available at https://www.cisco.com/c/m/en{_}us/solutions/enterprise-networks/networking-report.html{#} (Accessed 18 May 2021).
[165] X. Huang, S. Cheng, K. Cao, P. Cong, T. Wei, S. Hu, A survey of deployment solutions and optimization strategies for hybrid SDN networks, IEEE Commun. Surv. Tutor. 21 (2) (2019) 1483–1507.
[166] R. Amin, M. Reisslein, N. Shah, Hybrid SDN networks: A survey of existing approaches, IEEE Commun. Surv. Tutor. 20 (4) (2018) 3259–3306.
[167] OpenDaylight, Project: P4 plugin, 2020, Available at https://wiki-archive.opendaylight.org/view/P4{_}Plugin:Main (Accessed 15 June 2020).
[168] ONOS, Project: P4 brigade, 2020, Available at https://wiki.onosproject.org/display/ONOS/P4+brigade{#}P4brigade-Learnmore (Accessed 15 June 2020).
[169] A. Yazdinejad, R.M. Parizi, A. Dehghantanha, K.-K.R. Choo, P4-to-blockchain: A secure blockchain-enabled packet parser for software defined networking, Comput. Secur. 88 (2020) 101629.
[170] J. Halpern, C. Pignataro, et al., Service function chaining (SFC) architecture, in: RFC 7665, 2015, Available at https://datatracker.ietf.org/doc/html/rfc7665 (Accessed 18 May 2021).
[171] M. Pattaranantakul, Q. Song, Y. Tian, L. Wang, Z. Zhang, A. Meddahi, Footprints: Ensuring trusted service function chaining in the world of SDN and NFV, in: International Conference on Security and Privacy in Communication Systems, Springer, 2019, pp. 287–301.
[172] A.M. Zarca, M. Bagaa, J.B. Bernabe, T. Taleb, A.F. Skarmeta, Semantic-aware security orchestration in SDN/NFV-enabled IoT systems, Sensors 20 (13) (2020).
[173] K.D. Joshi, K. Kataoka, pSMART: A lightweight, privacy-aware service function chain orchestration in multi-domain NFV/sdn, Comput. Netw. 178 (2020) 107295.
[174] Gartner, Hype cycle for enterprise networking, 2019, 2020, Available at https://www.gartner.com/document/3947237?ref=solrAll{&}refval=253070378 (Accessed 15 June 2020).
[175] M.F. Hyder, T. Fatima, Towards crossfire distributed denial of service attack protection using intent-based moving target defense over software-defined networking, IEEE Access 9 (2021) 112792–112804.