IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 19, 2020, accepted July 28, 2020, date of publication August 3, 2020, date of current version August 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3013998

Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking Controller–A Review

MOHAMMAD A. ALADAILEH, MOHAMMED ANBAR, (Member, IEEE), IZNAN H. HASBULLAH, YUNG-WEY CHONG, (Member, IEEE), AND YOUSEF K. SANJALAWE
National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia (USM), Penang 11800, Malaysia
Corresponding author: Mohammed Anbar (anbar@nav6.usm.my)

ABSTRACT The wide proliferation of telecommunication technologies in the last decade gives rise to the number of more sophisticated security threats. Software-Defined Networking (SDN) is a new networking architecture that isolates the network control plane from the data plane that incidentally provides better features and functionalities to detect and deal with those security threats. Its elastic programmable feature permits efficient network management and provides network operators with the flexibility to monitor and fine-tune their network. However, the new technology also created many new security concerns, and the threat of Distributed Denial of Service (DDoS) attack is one of the major concerns. This paper presents a comprehensive review of state-of-the-art techniques to detect DDoS attacks on SDN controller. It first describes the SDN technology and then elaborates on the mechanism of DDoS attacks on SDN. Additionally, this paper also describes all major DDoS detection techniques and classifies them at a very high level according to the techniques or methods used. The current survey is qualitatively compared with the existing surveys using various author-defined metrics. Finally, this paper provides a guideline for future research related to detection techniques of DDoS against the SDN controller.

INDEX TERMS Denial of service (DoS), distributed denial of service (DDoS), entropy, detection techniques, software-defined networking (SDN).

I. INTRODUCTION

The management of traditional network architecture is commonly characterized as complex and rigid due to the difficulty in controlling or transforming the network to satisfy changing business requirements [1]. The Internet has revolutionized the development of communication and computer technologies; and the integration of other technologies such as mobiles, radio, etc. brings additional capabilities and provides more services to users [2]. The integration of various technologies into the traditional network makes it more difficult to fulfil new demands such as scalability, security, flexibility, dependability, reliability, etc.

Software-defined networking (SDN) architecture [3], [4] is steadily gaining traction as a novel network architecture to fulfill the aforementioned requirements of network management as it is more agile and flexible to be implemented and managed compared to traditional network architecture. SDN is designed to provide a high-level abstraction on top of the hardware/software infrastructure by separating the network data plane from the control plane. Networking devices could also be programmed directly [5].

SDN comes with a logically centralized controller that can analyze traffic and configure new instructions to be forwarded to switches' tables. The controller is the brain of the network that manages the entire network traffic flows; makes decisions based on the analysis of the traffic flows; and collects statistics of incoming packets.

Those features provide the SDN with the ability to detect and react to changes or abnormalities in the network. On the other hand, the decoupling of the network control plane from the data plane also introduces a new attack vector for attackers to target or exploit that could potentially create a new security vulnerability.

The associate editor coordinating the review of this manuscript and approving it for publication was Jose Saldana.

Over the last decade, the Distributed Denial of Service (DDoS) attack has become a real threat to SDN due to the severity of its impact on the network; from cutting off access of legitimate users from network services or resources to the extent of collapsing the entire network. The centralized controller is the main component of an SDN. Any threat to the controller may cause network breakdown thus it becomes an attractive target to attackers [6], [7]. In other words, the controller becomes a single point of failure risk, which could directly affect the performance, availability, and reliability of the network. However, securing the SDN controller from DDoS attack is a challenging and resource-intensive task that reduces the effectiveness of the controller in managing the network. This is even more so, given the fact that there are different types of DDoS attacks on SDN [8]. Therefore, any effort to secure SDN infrastructure against DDoS attack requires a comprehensive understanding of SDN characteristics; significant features of network traffic that characterize DDoS attack against SDN; and DDoS attack behaviors in SDN. A study on DDoS attacks on SDN [9] had revealed several distinct features and behaviors that could be used as indicators to detect DDoS attacks.

The contributions of this survey paper are: (i) comprehensive review of different types of DDoS attack on SDN controller, and the various techniques to detect them; (ii) a qualitative comparison of this survey with other related surveys in similar areas; and (iii) suggestions of future research directions in the field of DDoS detection technique in SDN environment to benefit the research community.

The rest of the paper is structured as follows. Section II provides a research background on SDN and SDN controller. Section III provides a detailed explanation of security issues in SDN, including the impact of DDoS attacks as well as the most common types of DDoS attacks on the SDN controller. Section IV presents the result and discussion on the qualitative comparison with existing reviews on detection techniques of DDoS attacks against the SDN controller. Section V discusses techniques to detect DDOS attacks against the SDN controller. Finally, Section VI and VII provide future research directions and conclusion, respectively.

II. BACKGROUND

A. SOFTWARE-DEFINED NETWORKING

SDN is a new and better network architecture than traditional network architecture in controlling network traffic flows as well as having elasticity and flexibility to be programmed for efficient network management. Researchers have been trying to find methods to secure networks from attacks for years, but these efforts were confronted by many challenges in terms of performance, scalability, reliability, and security [10]–[12]. The emergence of SDN technology excites the research and security communities as it provides novel and alternative means to address the challenges [13].

The design of the SDN environment that decouples the control plane from the data plane enables innovative security solutions to protect networks from attacks. It allows the network to be managed dynamically via a logical and centralized control function that provides instructions to the data plane to forward network traffic [14]. However, the centralized control feature could potentially be a liability as it becomes a single point of failure risk due to the high dependency of the network on it. Thus, the centralized SDN controller appears to be an attractive target for DDoS attacks since any successful attack may lead to network degradation or even a complete breakdown. Attackers also exploit the limitations of the switches in the data plane such as its memory capacity. The main objective of DDoS attack that targets the SDN controller is to overwhelm and exhaust its resources, typically by flooding the network with spoofed IP packets, thereby resulting in congestion that degrades or collapses the network.

At the same time, a centralized SDN controller could play a role as a virtualized network that makes the network highly elastic and easier to manage by gathering network statistics of incoming packets; and identifying devices in the network that deal with the controller. The SDN controller could also contribute to improving network performance by leveraging its programmability and flexibility [15]. Particularly, since the control plane is separate from the data plane, all network packets without matching rules in the flow table will be passed to the controller [19]. In other words, the controller facilitates the monitoring of network traffic flows by dealing with two types of objects. The first object is for network control which includes the policies on packet forwarding for the switch table, and the second object is related to network monitoring in the form of network status which allows the study of network traffic behavior. Table 1 summarizes the features of SDN that facilitate the detection of DDoS attacks [9], [16], [17].

TABLE 1. Features of SDN to detect DDoS attacks.

Feature	**Feature Description**
Decoupling of control plane and data plane	Facilitates network traffic engineering, maintaining a network policy and security via programmable platform to implement experiments and virtualization environment of Network Functions.
Centralized Logical Controller	Ability to control, monitor, and analyze traffic behaviors from potential security threat. Help create proportionate security rules.
Programmability	Assist control on network behavior via software to simplify operations, enhance dynamic configuration of networks.
Updating of entry flow rules	Allows updating of entry flow rules to match abnormal behavior (attacks) to detect attack traffic entering the network.

SDN is expected to deal with over half of the network traffic flows in a not so distant future. Furthermore, SDN will help data centers to control costs and manage network traffic more efficiently which will drive its adoption, whether partially or fully, by a high percentage of network operators to control traffic flows as projected in a study conducted by Cisco [18]. Furthermore, in the past few years, many solutions have been proposed to solve issues related to data center security and to make it easier to manage with SDN which will further increase the adoption of SDN in the future [19].

As mentioned earlier, SDN is an innovative networking technology that is better than its predecessor due to its many characteristics such as logically centralized control, open programmable interfaces, switch management protocol, third-party network services, virtualized logical networks and centralized monitoring units [4], [10], [11], [20]. Figure 1 depicts a comparison between SDN and traditional network architecture. The complexity and inflexibility of the traditional network make it difficult to address all operations in the network. The inflexibility of the traditional network is due to all functional components–control, data, and application –residing in the same layer with no differentiation of their functions in dealing with incoming packets. On the other hand, SDN fulfils its objective to mitigate network complexity by dividing the network into three separate layers [12], [13], [14] that isolates the functional components which enable a controller to centrally manage and control the entire network. This separation enables applications to have a network-wide view and establish a centralized visibility to manage the traffic flow. It also provides the capability to virtualize the entire network infrastructure that will further simplify the task of configuring and managing the network.

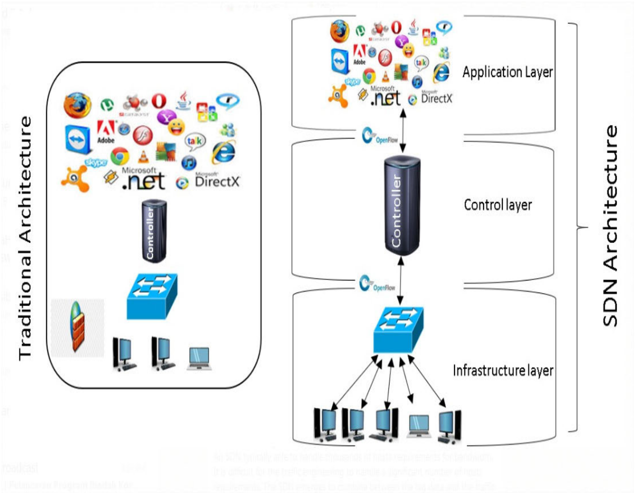

FIGURE 1. Traditional Network vs SDN Architecture [15].

SDN isolates the characteristic of control from data planes that allow network configurations to be made that could further enhance and improve the performance, as well as open the path for security innovations on the network architecture and operations [21]. Moreover, it provides instant network status which makes efficient control and flow handling procedures possible while keeping the control plane flexible and intelligent [16].

SDN properties of the network operations are important for the requirement to improve network security. However, optimizing the performance of the network is challenging because of the difficulties in managing and controlling a huge amount of data. The emergence of SDN offers an opportunity to broadly enhance the performance of the network by allowing a centralized controller to manage and control network traffic flows for the entire network, as shown in Figure 2. The SDN manages the entire network via application programming interfaces (APIs) located between the layers to connect the networks together [22]. By contrast, the nature of traditional network design as a singular package makes it difficult to manage the data traffic effectively and to enhance its performance further [15].

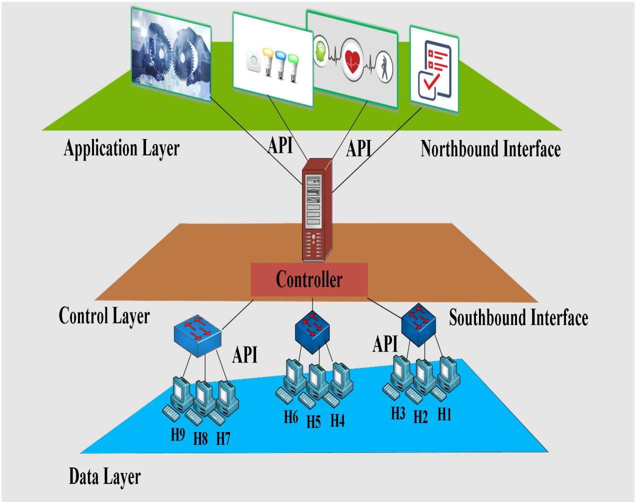

FIGURE 2. A general SDN layered architecture [23].

The performance of SDN networks is highly dependent on the controller since the operation of the control plane fully relies on the controller. Essentially, the switch table requests new or updated flow entries (rules/instructions) from the controller. Rule updates and the awareness of network status require frequent communication between the controller and the switch devices. The first 'station' that deals with incoming packets within an SDN environment is a network switch that inspects all packet header features against the entry table (flow table rules). If no match is found, then the packets are forwarded to the controller through secure channels for further processing. Thus, this paper expounds on the contributions of the SDN controller in managing and processing incoming packets and the roles of the SDN controller in network security.

B. SOFTWARE-DEFINED NETWORKING (SDN) CONTROLLER

SDN controller plays many considerable roles in the network such as configuring flow table; monitoring networking devices by establishing secure connections; and updating instructions to the flow table in the infrastructure layer (switch's table) to identify new traffic flow. The update to the flow table is required before the switch could handle new incoming packets and it is only possible due to the ability of the controller to create new rules and update the switch [24]. In addition, the controller could manage the entire traffic flow by assuming the role of a manager between the infrastructure layer and application layer through open API southbound, northbound and east/westbound interfaces [25], and decides whether the traffic flow is normal or abnormal by making use of the network traffic flow statistics collected by the controller as a baseline input (information) to an attack detection method. To do that, the controller uses various ways to gather statistical information about network traffic [26]. Therefore, the controller plays a very important role in any effort to enhance and improve the SDN security against malicious attacks.

In summary, the controller simplifies the operations of the network by centralizing the control of the entire network. All incoming packets will be inspected according to the switch's flow table that receives policies and instructions from the controller. If a match is found, the packet is forwarded to the destination accordingly; else the packet will be forwarded to the controller for further processing or dropped [27]. In other words, the controller acts as the brain of the network by making forwarding decisions for all incoming packets into the network.

III. SECURITY ISSUES IN SDN

SDN controller is one of the more robust security solutions that provides protection for networks against threats. However, as the number of users and volume of network usage increase, the probability of potential security issues also increases [28]. However, so far, there has been no effective approach to detect low-rate DDoS attack with high accuracy and low false positive rate; and given that the controller being the key and focal component of SDN, any problem occurring at the controller may degrade or even collapses the entire network [1], [17], [29], [30]. Therefore, many researchers who have been studying the security issues and challenges of SDN architecture had proposed many suggestions or potential solutions to address some of the issues. One of the issues is the poor performance of the SDN controller when overwhelmed by a huge number of incoming packets or flows that impedes the controller's ability to process incoming packets. The question on how to further improve the performance of the SDN controller certainly needs further research to answer [31]. Another challenge is the accuracy of attack detection because attackers keep changing the attack behavior especially when the attack traffic is made to resemble normal traffic and thus hardly distinguishable [32]. As aforementioned, the attack aims to overwhelm network device resources by flooding the SDN-switch with a huge volume of unmatched packets. These unmatched packets are treated as new packets by the switch and will be forwarded to the controller [33], [34]. On other hand, SDN also faces challenges in other aspects of security, such as malicious applications, data modification, misconfiguration issues and denial of service [4], [35]–[37].

Several existing techniques to detect DDoS attack on SDN architecture, as explained in section VI, invoke a lot of back and forth communication over the controller-switch communication channel for the purpose of acquiring network statistics that result in a jump in bandwidth consumption and resource utilization. However, a DDoS attack would overwhelm the controller by either duplicating packets that would flood the controller as new packets or spoofing the source address. Any packet that does not have a matching entry in the flow table (switch table) will be sent to the controller for further processing. This forces the controller to send back a new instruction to the switch to deal with subsequent incoming packets [38]. But when the number of new incoming packets is much larger than the bandwidth and controller's ability to process them, it will result in the collapse of the entire network [39]. Table 2 shows some of the security issues of SDN.

TABLE 2. Security Issues on SDN.

Security Issue	Description	Solutions
Scalability	Switch-controller latency burden the infrastructure and controller that cause delay in processing new incoming packets [40].	Avoid controller bottleneck by switch table. Put a device between the data plane and the control plane.
Reliability	A single centralized controller could fail under bombardment of packets. Thus, using a single controller is an unreliable method to detect D/DoS attack. [4].	Using multiple controllers to tackle new incoming packets.
Programmability	Easy for attackers to read un-complicated codes that allow modification to attack behavior [40].	Make codes more complex to prevent attackers from changing attack behavior to evade detection.
Dependability	Many gaps in security enable exploitation of SDN vulnerabilities such as limited memory size, and lacked a mechanism to detect abnormal traffic behavior [41].	Work to improve accuracy to detect abnormal behavior and achieve high security.

Meanwhile, few proposed approaches [42]–[44] re-monitor the traffic flow between the controller and switches in order to access the controller's rules to the switches' tables to intercept DDoS attacks at the boundary switches of SDN. However, these approaches are incapable of detecting DDoS attacks that continuously change their attack behaviors due to the reliance on static switch rules. Static switch rules or switch flow tables are not effective in dealing with attackers that keep changing the attack traffic behavior to resemble normal traffic. Recently, a powerful SDN controller security middleware has been proposed as a promising way to deal with suspicious traffic flow [45], [46]. However, most SDN switches are not built with sufficient intelligence to be able to recognize the fluctuation of traffic flow early enough in order to detect DDoS attacks; thus, making it impractical to rely on switches to secure the network from DDoS attacks [34].

A. DISTRIBUTED DENIAL OF SERVICES (DDOS) ATTACKS

DDoS attack is a serious threat to network stability and security due to a huge resource asymmetry between the network and the victim because the attacks usually come not only from multiple sources but also distributed geographically [47]–[50].

Attackers typically initiate their attack by scanning the network to search for security hole or vulnerable host that could be exploited. If found, the vulnerability is then exploited to gain control and to infect them with malicious programs.

Attackers continuously change the DDoS attack methods to evade detection and discovery using novel techniques. Oftentimes, the identity of the subverted host is hidden by spoofing the source IP address in attack packets to prevent discovery. Thus, DDoS attack poses a serious threat to SDN network quality, especially if it affects the SDN controller, whether directly or indirectly [51]. DDoS attack attempts to prevent legitimate users from accessing network resources or deny their access to network services [52]. Most DDoS attacks are diversified to avoid detection as well as to increase the chances of reaching the targeted victim by using different attack scenarios or utilizing different methods of DDoS attack, such as ICMP flooding, TCP flooding [53] and UDP flooding [54]. The most common types of DDoS attack are listed in Table 3.

TABLE 3. Common Types of DDoS Attack.

Type of DDoS attacks	Principles of Operation	Target
SYN	Transmits many SYN packets to the victim using TCP connection to prevent return of ACK to the victim causing resources to be exhausted [55].	Victim's machine
HTTP Flood	Transmits massive number of requests to web server to overwhelm it and unresponsive to legitimate user request [56].	Server
UDP	Transmits a huge number of packets to random ports of victim causing the machine to look for applications on these ports. Thus, the machine has to send Unreachable Destination packet in response to each incoming packet. When incoming packets increase, the delays also increase [57].	Inaccessible machine
DNS	Transmits spoofed IPs and asks for the response which is more than what the victim when it is directed to it. Thus, they change the source IP address. It causes massive traffic [58].	Victim
ICMP	Transmits a huge number of ICMP pings to the victim to exhaust the victim's resources [59].	Server and Victim

SDN architecture provides potential improvements in networking security aspect by providing the network with programmability features that can bring benefits to existing intelligent systems such as intrusion detection system (IDS) and intrusion prevention system (IPS) [60].

Although the roles of SDN properties in improving network security have been well highlighted, the methods to secure the SDN controller are still not properly addressed, thus exposing the network to threats [61].

DDoS attacks against SDN controller are hard to detect using traditional DDoS attack detection techniques because these attacks have similar features as the flooding attacks. Figure 3 illustrates one of the mechanisms of DDoS attack on the SDN controller. In addition, DDoS attacks against SDN controller could be easily executed using affordable tools and do not require high performance computing or much effort from the attackers [62].

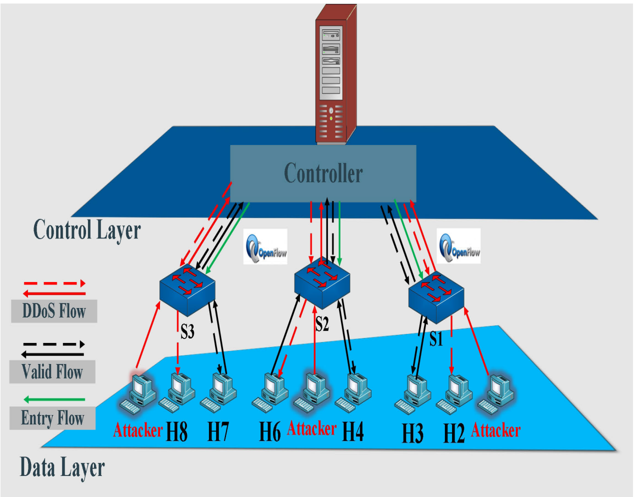

FIGURE 3. DDoS Attacks on SDN Controller [64].

DDoS attacks can be classified based on the protocol used or layer targeted. Attacks on the control layer mostly use TCP, ICMP or UDP packets in order to exhaust the victim bandwidth. Attacks on application layer usually attempt to deny legitimate users of services by exhausting the resource of the server that provides the particular service [63].

DDoS attacks on the SDN controller are launched by sending a massive amount of network traffic with spoofed source IP addresses from different sources (hosts' attacks). In fact, the power of DDoS attack to devastate the victim lies in the use of multiple types of attack scenarios against the target [32]. A sophisticated DDoS attack not only uses less bandwidth and low number of packets but also mimics the behavior of normal traffic and varies the traffic rate (e.g. low or high) to evade detection which increases the effectiveness of the attack.

IV. QUALITATIVE COMPARISON WITH EXISTING REVIEWS ON DETECTION TECHNIQUES OF DDOS ATTACKS AGAINST SDN CONTROLLER

There are many reviews that discuss existing detection techniques of DDoS attacks against SDN controller. Therefore, in this section, a qualitative comparison is conducted to point out the uniqueness of this review compared to others. The qualitative comparison is based on the metrics as shown in Table 4. The metrics are: (i) Number of techniques, (ii) Mechanisms classified, (iii) Entropy-based detection technique, (iv) Low rate traffic flow detection technique, (v) Time-based detection technique, (vi) Intrusion detection system, and (vii) Machine learning techniques. These metrics are author-defined based on intensive review of many existing detection techniques. Such comparison is important in order to understand the critical issues related to DDoS attack against SDN for the purpose of finding a more effective detection technique. In addition, it could serve as a guideline for future researchers who conduct research in a similar field. This review is benchmarked with three existing reviews from [34], [39], [45].

TABLE 4. Qualitative comparison with the existing reviews.

Criteria	This work	[34]	[45]	[39]
Number of techniques	15	8	6	9
Mechanisms classified	Yes	Yes	No	No
DETECTION TECHNIQUE				
Entropy-Based Detection Technique	6	2	3	2
Low Rate Traffic Flow Detection Technique	3	1	-	-
Time-based Detection Technique	2	3	-	-
Intrusion detection system	2	-	1	-
Machine Learning Technique	2	-	1	2

As previously mentioned, the SDN architecture provides a suitable environment and tools to simplify management and control of the traffic flows via the controller that could potentially help in finding solution to some of the challenges in detecting DDoS attack that has been confronting the traditional network architecture.

However, due to the vital role of SDN controller to the network, a failure or problem occurring at the controller may degrade and even collapse the entire SDN network. Therefore, an efficient and high-performance method of DDoS detection is very much needed to identify, evaluate, and respond to incidents before they could negatively affect the network.

Many techniques have been proposed to detect DDoS attack against the SDN controller to secure the SDN network. In this paper, each approach is scrutinized according to performance, accuracy, detection duration, and traffic flow rate. Each technique has specific characteristics based on the criteria listed in Table 4. This section explains the existing works related to the detection method of DDoS attacks against SDN and summarizes the findings (results) and drawbacks for each approach in Table 5. This paper is the first attempt at classifying some of the existing DDoS attack detection approaches based on the technique and features used; threshold nature; and the location of where the approach is deployed in the SDN environment, which are summarized in Table 6.

TABLE 5. Summary of Existing DDoS Attacks Detection Techniques in SDN.

Ref.	Findings	Drawbacks
[45]	Detect DDoS attack in its early stages. Low false positive and false negative.	Overload the controller. Unable to deal with large window size (over 50 packets).
[44]	Capable of early detection.	Only handles one type of DDoS attack. Only handle single victim.
[65]	Effectively protect the controller from DDoS attack.	Delay when handling enormous number of incoming packets
[42]	Able to distinguish attack traffic from legitimate traffic. Low false positive.	Only handles low traffic rate and threshold is fixed.
[66]	Traces attack source. Reduces workload on controller in early stage.	Works after controller received traffic flow which leads to flooding of controller by new incoming packets.
[67]	Reduce overload between controller and switches.	Unable to separate legitimate traffic from attack traffic.
[6]	Able to distinguish DDoS attack from flash crowd.	Overload the controller. Delay in DDoS attack detection.
[43]	Reduces controller overload. Quick reaction in detecting DDoS attacks.	Unable to deal with complex traffic flow in the switch.
[46]	Reduced resource consumption. High detection ratio on DDoS attack.	Requires high computing resources and processing power of SDN controller.
[61]	Prompt, versatile and accurate detection of DDoS attack. Limits false positive and false negative rate.	High resource consumption on the controller. Does not care in the temporal characteristics in order to accelerate detection process.
[68]	Reduces congestion of incoming packet at controller.	Consume time to process new network packets.
[69]	- Low false positive rate. - Increase detection accuracy.	Difficult to detect unknown or new type of DDoS attacks. Need time to detect the attack. Overload the controller

A prompt and accurate detection of any type of attack on SDN controller is a challenging task. However, it is crucial to quickly detect the attack before it reaches the controller to prevent the degradation or even the collapse of the entire network. Different techniques have been proposed to detect DDoS attacks. Each technique uses different approach and parameters; and has specific characteristics, advantages, and limitations.

V. TECHNIQUES TO DETECT DDOS ATTACKS AGAINST SDN

Since networks, especially SDN, has been subjected to constant barrage of DDoS attacks for the past 10 years, researchers have at their disposal many methods to detect DDoS attacks. Many studies on DDoS attack detection techniques are in the literature [6], [41], [43], [45], [46], [61], [65]–[67]. In this survey, as tabulated in Table 6, DDoS attack detection techniques are classified into two broad categories based on the location of the deployment.

TABLE 6. Detection Techniques of DDoS Attack In Existing Approaches.

Ref.	Techniques	Features	Threshold	Deployment location
[45]	Entropy	Dst_IP	Fixed	Controller
[44]	Entropy	Dst_IP	Fixed	Out controller
[65]	TDDAD	Time feature	Number of thresholds	Out controller
[42]	Entropy	Dst_IP	Fixed	Out controller
[67]	Entropy	Dst_IP	Fixed	Controller
[6]	EDDM	Dst_IP	Fixed	Controller
[43]	Statesec	Dst_ IP, dst_ port, src_IP and src_ port	Fixed	Out controller
[46]	SOM	Src_ IP, dst_port	Fixed	Controller
[61]	SORT	Dst_IP address	-	Out controller
[68]	Entropy	Dst_ IP	Fixed	Out controller
[69]	Entropy	Flow duration, src_ IP, packet length, dst_port	Fixed	Controller

1) SOURCE-BASED TECHNIQUE

The source-based techniques are deployed near the source of DDoS attack to pre-empt the attack at its onset. DDoS attack usually overwhelms the switch table by flooding the network with spoofed IP packets until the arrival rate of incoming packets is beyond the ability of the controller to handle and its resources depleted. Thus, the centralized SDN controller becomes a single point of failure in such situation.

The authors in [42] employed information distance (ID) to detect DDoS attack with minimum number of features within a certain detection period. They also applied an entropy method based on dest_IP occurrences of incoming packets within a certain window size. Their method depended on two thresholds: entropy value and information distance (ID). However, this technique only handles low-rate DDoS attack and the threshold is fixed which may reduce the detection accuracy and increase false positive rate.

The switch receives new instructions from the controller for every new flow without matching rules to be added into the switch's table. But this procedure may overload the controller as all flows without matching rules in the switch table have to be forwarded to the controller. However, there are efforts to make the switches smarter in detecting DDoS attacks without resorting to the controller through a novel approach presented in [43] called StateSec. This approach depends on an entropy method and the traffic monitoring of relevant features. The authors claimed that their proposed approach had a fast reaction time, prevented controller overload and had a high detection accuracy. However, there might be a delay in the attack detection because the switch is not only responsible for the collection of the statistics for the source IP, but also required to process the decision to detect DDoS attacks instead of the controller which requires the switch to perform complex computations.

A novel detection approach was proposed for early detection and mitigation of TCP SYN flooding by harnessing the programmability and wide visibility of the SDN through an entropy method in order to determine the randomness of the flow [44]. The entropy is calculated using the destination IP address and a few selected attributes of the TCP flags. The proposed approach was evaluated in term of average detection accuracy rate, average attack detection time and average false positive rates. However, the entropy-based approach that is based on a single feature from packet header (e.g. source IP or destination IP) does not help the controller to accurately determine whether it is under DDoS attack or not. Therefore, the use of multiple features is preferable as it will have a significant contribution in enhancing the accuracy of the attack detection [61].

Time-based detection and defense scheme against DDoS (TDDAD) [65] is a DDoS attack detection method based on time feature by extracting the temporal behavior of an attack. The use of time feature is for the purpose of detecting and defending against DDoS attack quickly and effectively. An attacker uses the OpenFlow switches to overwhelm the controller with a huge number of packets instead of attacking the controller directly since the incoming packets will be automatically forwarded to the controller for processing. However, since this method uses the content feature to detect an attack, the detection could be bypassed with some modification to the content of the malicious packet. TDDAD consists of five modules: statistics collection, feature extraction, attack detection, attack defense and port recovery.

The controller is responsible for updating flow rules and configuring new rules according to the flow. However, attackers could exploit the gap in the controller's reaction time in handling new network packets to launch their attack on the SDN controller by sending a large number of requests to the controller within this time window. A new method to filter the requests could decrease the entropy value by sending any new network packet directly to the security gateway instead of the controller to detect DDoS attack by an entropy method, and to generate the rules for these new requests to the switch flow table [68]. This method depends on three features to calculate three kinds of entropies: protocol, source IP address, and destination IP address. However, the detection method consumes time to process new packet flows.

2) DESTINATION-BASED TECHNIQUES

The destination-based detection techniques mostly employ the detection and defense at the target of the attack. In this survey, since the SDN controller is the target, it is considered the destination of the attack.

The strategic location of the controller within the network is an advantage that was capitalized by [45] to detect DDoS attacks in an effective and lightweight manner. The proposed approach employed an entropy method to calculate the probability of random incoming packets to detect an attack in the early stage by choosing a fixed threshold. The entropy is calculated using the destination IP address. However, a single point of failure situation could occur in two ways. First, excessive traffic causing a bottleneck in the switch-controller communication channel that blocks legitimate traffic from reaching the destination. Second, the number of inbound packets that reach the controller exceeds its processing capability. Hence, the occurrence of any one of the two will result in the exhaustion of the controller resources that prevent legitimate packets from reaching the controller. Therefore, the entropy method allows the controller to evaluate the rate of incoming packets destined to a specific host or subnet.

The emergence of SDN not only makes network management more flexible and programmable, but it also presents an attractive target for attackers. By constantly bombarding the controller with attack packets, attacker forcefully puts extra burden and strain on the controller that needs to process all incoming traffic packets in order to detect or prevent any potential DDoS attack. Until a reliable and effective solution is found that is able to secure the SDN network from DDoS attack, this issue will continue to attract the attention of researchers in the industry and academia alike.

One of the new methods that has been proposed to solve this issues is based on self-organizing mapping (SOM) network that gives early alert according to the probability of occurrence of packets in an event [46]. This method manages to decrease resource consumption, lower false positive rates and improve detection ratio for UDP and TCP traffic flow, but has a high false positive rate for ICMP traffic flows.

A different technique to detect DDoS attack was proposed by [66] with the objectives to identify the path where attack traffic passed through, to achieve rapid response from the detection module and to cope with the limitations of fixed detection loop approach. It comprises four modules: attack detection trigger, attack detection, attack traceback and attack mitigation, that achieves rapid detection of DDoS attacks thus helps to reduce the workload of controllers.

Several researchers proposed a method to detect DDoS attacks against SDN controller by collecting and analyzing statistics from switches' tables. However, the large amount of data that needs to be exchanged between hosts, especially in a large network, increases the burden of data collection process. Besides, the increase in contact frequency between the hosts could overload the switch-controller communication channel that makes it difficult to capture all communication between the switches and the controller. However, many efforts to collect statistics from the switches are now handled by a flow statistics process in the switch [67]. The authors used a lightweight entropy-based DDoS flooding attack detection method running in OpenFlow and reduces the flow collection load to the controller to lessen the overhead from frequent flow collection and make the switches more intelligent to proactively detect DDoS attack at the switch. Thus, the frequency of communication between the controller and the switches is reduced.

Many prior researches were concentrated on the defense against DDoS attack on SDN. Some existing researches suffer from false positive. The similarity between DDoS attack traffic behavior and flash crowd behavior was analyzed by [6]. They proposed an entropy-based DDoS defense mechanism (EDDM) that runs on the controller. EDDM tries to keep legitimate packets from being dropped during flash crowd events and thus prevent denial of service to legitimate users on the network. That makes the operation of DDoS attack detection more accurate and reduces false positive rate. At the same time, this mechanism depends solely on one wind ow to detect abnormal flows according to the entropy method that needs to produce a result in a short time. Thus, this mechanism may overload the controller and delays the detection of DDoS attack.

Many solutions to detect DDoS attack against the SDN controller depend on a single packet feature for efficiency reason. However, the use of a single feature puts many restrictions on DDoS detection operation. To eliminate the restrictions, a new approach was proposed in [69] that used multiple packet header features based on joint-entropy method. The authors utilized information theory in their method because it is more scalable, less complex and gives more accurate result. Furthermore, the method could detect both spoofed and non-spoofed DDoS attacks in online mode by making use of flow duration, source IP address, packet length and destination port as features to reduce false positive rate and improve detection accuracy. However, the setback is it requires longer time to detect an attack.

The Difficulty in Detecting DDoS Attack in SDN

As mentioned earlier, the SDN offers network administrators ease of management and programmability by decoupling the control plane from the data plane [1]. Although there have been many excellent works done by researchers and security communities to detect DDoS attack in SDN environments, the trend of DDoS attacks is still rising; and some of them with new forms and characteristics. Some difficulties in defending against DDoS attacks in SDN environments are described below.

1- Statistical data: Most DDoS attack detection techniques need to collect data from the infrastructure layer switches) to construct their approaches, such as in the method to extract the necessary features of packet header to detect abnormal behavior. Since the frequency of DDoS attack keep increasing, collecting statistical data from the traffic flows becomes more difficult and challenging, especially when involving low-rate DDoS attack. Moreover, there are techniques that distribute the data collection tasks across multiple switches in the SDN network to balance the data collection loads. However, this makes it harder to collect precise data for use in detecting DDoS attack against SDN networks.

2- Algorithm selection: The diversification of DDoS attack behavior complicates the detection of abnormal traffic in SDN environment. Thus, many algorithms have turn to artificial neural network, Bayesian classification, fuzzy logic, etc. to detect DDoS attack behaviors. However, there is no single algorithm that is able to deal with all variations of DDoS attack behaviors.

3- Prompt response: Prompt response to DDoS attack is extremely important for SDN controller to maintain the availability of the network. However, under DDoS attack, the controller deals with massive amount of traffic that could exhaust all of its resources which cripple its ability to respond to requests from legitimate users.

In response to the above-mentioned difficulties, several techniques have been proposed to detect DDoS attacks against SDN controller. However, the problems with existing DDoS attack detection approaches are many which include heavy burden on the controller to process an overwhelming number of ingress packets within a short time frame; inability to detect low rate DDoS attack; and high network resource consumption. In addition, invalid packets also put additional processing burden on the controller that cause delay in attack detection. Some approaches were meant only for low traffic flow, thereby resulting in high false positive rate.

VI. FUTURE RESEARCH

The existing detection techniques to secure SDN controller against DDoS attacks, as explained in Section VI, still suffer from various issues. Future research in the field should pay close attention to the following aspects:

1- Fixed threshold: Many researchers attempt to secure the SDN controller against DDoS attack by proposing detection techniques that use a fixed threshold based on a certain number of packets within a time interval (e.g. 500 packets per *t* time). Therefore, a high false positive rate remains an issue. Thus, a method to dynamically calculate the threshold needs to be developed and investigated. Researchers could also apply dynamic threshold features based on the traffic statistics.

2- Low-rate DDoS attack detection: DDoS attack with low traffic rates is almost impossible to detect using a technique that solely relies on a single packet header feature [70]. From the attacker's point-of-view, a huge number of packets with spoofed source IP addresses are generated and blasted to a single host in the network. As a result, the targeted host will be flooded with these packets at roughly the same time which will eventually exhaust its resources.

The controller has difficulty in determining whether the particular traffic is part of DDoS attack or not because the packets seem to be coming from multiple sources at a relatively 'normal' rate, thus resulted in a high false positive rate and low detection accuracy. To enhance the detection and identification of DDoS attack with a low traffic flow rate, the detection technique needs to rely on multiple packet header features instead of a single feature.

3- Controller overhead: Some SDN security approaches are deployed at the controller. To make the matter worse, some of these approaches require analysis of the entire traffic flow to detect DDoS attacks which would add unnecessary burden and overhead on the controller since traffic flow analysis is a very resource-intensive process. Thus, selecting appropriate packet features and deploying the detection approach at a separate location other than the controller would reduce the overhead of the controller.

VII. CONCLUSION

This paper provides an overview of SDN concept by illustrating the importance of SDN features' in managing, monitoring, and programming the network using SDN controller. The controller is also playing a central role in securing the network from various threats as elaborated in section II. Section III of this paper discusses some of the security issues that plague the SDN controller; explains the effect of DDoS attacks, particularly on SDN controller; and elaborates on some of the common types of DDoS attacks. Section IV gives a comprehensive analysis of existing DDoS detection techniques and compares it with three other existing surveys according to certain criteria. In addition, this section provides a comprehensive study on the detection techniques of DDoS attacks in SDN and summarizes the findings and drawbacks. Furthermore, this paper is the first to classify some of the existing DDoS attack detection approaches based on the technique and features used; the threshold nature; and the location where the approaches have been deployed in the SDN environment.

This paper highlights the limitations of several DDoS attack detection approaches which could be solved by utilizing a better and more efficient technique that increases detection accuracy and reduces false positive rate. Finally, researchers could also exploit the advantages or strengths of existing approaches by hybridizing them to obtain a more comprehensive detection approach against DDoS attacks.

REFERENCES

[1] J. Chen, X. Zheng, and C. Rong, "Survey on software-defined networking," in *Cloud Computing and Big Data. CloudCom-Asia* (Lecture Notes in Computer Science), vol. 9106, W. Qiang, X. Zheng, and C. H. Hsu, Eds. Cham, Switzerland: Springer, 2015, doi: 10.1007/978-3-319-28430-9_9.

[2] B. M. Leiner, V. G. Cerf, D. D. Clark, R. E. Kahn, L. Kleinrock, D. C. Lynch, J. Postel, L. G. Roberts, and S. Wolff, "A brief history of the Internet," *SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 5, pp. 22–31, Oct. 2009.

[3] A. Samson and N. P. Gopalan, "Software defined networking: Identification of pathways for security threats," in *Proc. Int. Conf. Informat. Anal. ICIA*, 2016, p. 16.

[4] S. Scott-Hayward, S. Natarajan, and S. Sezer, "A survey of security in software defined networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 623–654, 1st Quart., 2016.

[5] M. A. AL-Adaileh, M. Anbar, Y.-W. Chong, and A. Al-Ani, "Proposed statistical-based approach for detecting distribute denial of service against the controller of software defined network (SADDCS)," in *Proc. MATEC Web Conferences*, vol. 218, 2018, p. 2012.

[6] Y. Jiang, X. Zhang, Q. Zhou, and Z. Cheng, "An entropy-based DDoS defense mechanism in software defined networks," in *Communications and Networking. ChinaCom* (Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering), vol. 209, Q. Chen, W. Meng, and L. Zhao, Eds. Cham, Switzerland: Springer, 2018, doi: 10.1007/978-3-319-66625-9_17.

[7] C. Bouras, A. Kollia, and A. Papazois, "SDN & NFV in 5G: Advancements and challenges," in *Proc. Innov. Clouds, Internet Netw. (ICIN) 20th Conf.*, 2017, pp. 107–111.

[8] Tom Bienkowski. (2018). *1.7tbps DDoS Attack Makes History | NETSCOUT*. Accessed: Dec. 25, 2018. [Online]. Available: https://www.netscout.com/news/blog/security-17tbps-ddos-attack-makes-history

[9] T. Jose and J. Kurian, "Survey on SDN security mechanisms," *Int. J. Comput. Appl.*, vol. 132, no. 14, pp. 32–35, Dec. 2015.

[10] A. Abdelaziz, A. T. Fong, A. Gani, U. Garba, S. Khan, A. Akhunzada, H. Talebian, and K.-K. R. Choo, "Distributed controller clustering in software defined networks," *PLoS ONE*, vol. 12, no. 4, pp. 1–19, 2017.

[11] S. Shin, V. Yegneswaran, P. Porras, and G. Gu, "AVANT-GUARD: Scalable and vigilant switch flow management in software-defined networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. CCS*, 2013, pp. 413–424.

[12] S. Khan, A. Gani, A. W. A. Wahab, M. Guizani, and M. K. Khan, "Topology discovery in software defined networks: Threats, taxonomy, and state-of-the-art," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 303–324, 1st Quart., 2017.

[13] C. Yoon, T. Park, S. Lee, H. Kang, S. Shin, and Z. Zhang, "Enabling security functions with SDN: A feasibility study," *Comput. Netw.*, vol. 85, pp. 19–35, Jul. 2015.

[14] D. Kreutz, F. M. V. Ramos, and P. Verissimo, "Towards secure and dependable software-defined networks," in *Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw. HotSDN*, 2013, p. 55.

[15] D. B. Rawat and S. R. Reddy, "Software defined networking architecture, security and energy efficiency: A survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 325–346, 1st Quart., 2017.

[16] S. Shin, L. Xu, S. Hong, and G. Gu, "Enhancing network security through software defined networking (SDN)," in *Proc. 25th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Aug. 2016, pp. 1–9.

[17] S. Sezer, S. Scott-Hayward, P. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, and N. Rao, "Are we ready for SDN? Implementation challenges for software-defined networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 36–43, Jul. 2013.

[18] Cisco. (2018). *Cisco Cloud Index: Data Center SDN to Skyrocket by 2021*. Accessed: Mar. 24, 2020. [Online]. Available: https://www.sdxcentral.com/articles/news/cisco-cloud-index-data-center-sdn-skyrocket-2021/2018/02/

[19] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1617–1634, 3rd Quart., 2014.

[20] D. He, S. Chan, X. Ni, and M. Guizani, "Software-defined-networking-enabled traffic anomaly detection and mitigation," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1890–1898, Dec. 2017.

[21] F. L. Sell, "Some of the pros and cons of central banking supervision by the ECB," *CESifo Forum*, vol. 13, no. 4, pp. 40–45, 2012.

[22] N. C. S. N. Iyengar, A. Banerjee, and G. Ganapathy, "A fuzzy logic based defense mechanism against distributed denial of services attack in cloud environment," *Int. J. Commun. Netw. Inf. Secur.*, vol. 6, no. 3, p. 233, 2014.

[23] W. Li, W. Meng, and L. F. Kwok, "A survey on OpenFlow-based software defined networks: Security challenges and countermeasures," *J. Netw. Comput. Appl.*, vol. 68, pp. 126–139, Jun. 2016.

[24] H. T. N. Tri and K. Kim, "Resource attack based on flow table limitation in SDN," in *Proc. Korea Inf. Process. Soc. Conf.*, 2014, pp. 215–217.

[25] Y. Jarraya, T. Madi, and M. Debbabi, "A survey and a layered taxonomy of software-defined networking," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1955–1980, 4th Quart., 2014.

[26] B. Gorkemli, A. M. Parlakisik, S. Civanlar, A. Ulas, and A. M. Tekalp, "Dynamic management of control plane performance in software-defined networks," in *Proc. IEEE NetSoft Conf. Workshops (NetSoft)*, Jun. 2016, pp. 68–72.

[27] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A survey on software-defined networking," *Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 27–51, 1st Quart., 2014.

[28] O. Salman, I. H. Elhajj, A. Kayssi, and A. Chehab, "SDN controllers: A comparative study," in *Proc. 18th Medit. Electrotech. Conf. (MELECON)*, Apr. 2016, pp. 18–20.

[29] S. Revathi, A. Geetha, and others, "A survey of applications and security issues in software defined networking," *Int. J. Comput. Netw. Inf. Secur.*, vol. 9, no. 3, p. 21, 2017.

[30] S. Luo, J. Wu, J. Li, and B. Pei, "A defense mechanism for distributed denial of service attack in software-defined networks," in *Proc. 9th Int. Conf. Frontier Comput. Sci. Technol.*, Aug. 2015, pp. 325–329.

[31] S. Latre, M. Charalambides, J. Francois, C. Schmitt, and B. Stiller, Eds., *Intelligent Mechanisms for Network Configuration and Security*. Berlin, Germany: Springer-Verlag, 2015.

[32] M. H. Bhuyan, H. J. Kashyap, D. K. Bhattacharyya, and J. K. Kalita, "Detecting distributed denial of service attacks: Methods, tools and future directions," *Comput. J.*, vol. 57, no. 4, pp. 537–556, Apr. 2014.

[33] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: Classification and state-of-the-art," *Comput. Netw.*, vol. 44, no. 5, pp. 643–666, Apr. 2004.

[34] H. D. Zubaydi, M. Anbar, and C. Y. Wey, "Review on detection techniques against DDoS attacks on a software-defined networking controller," in *Proc. Palestinian Int. Conf. Inf. Commun. Technol. (PICICT)*, May 2017, pp. 10–16.

[35] S. Shin and G. Gu, "Attacking software-defined networks: A first feasibility study," in *Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw. HotSDN*, 2013, pp. 165–166.

[36] R. Horvath, D. Nedbal, and M. Stieninger, "A literature review on challenges and effects of software defined networking," *Procedia Comput. Sci.*, vol. 64, pp. 552–561, Jan. 2015.

[37] N. E. Kolobova, Z. P. Valueva, and M. Y. Solodova, "Synthesis of formylcyclopentadienyltricarbonylrhenium and some of its properties," *Bull. Acad. Sci. USSR Division Chem. Sci.*, vol. 29, no. 10, pp. 1701–1705, Oct. 1980.

[38] S. M. Mousavi, "Early detection of DDoS attacks in software defined networks controller," Ph.D. dissertation, Carleton Univ., Ottawa, ON, Canada, 2014.

[39] M. Kia, *Early Detection and Mitigation of DDoS Attacks in Software Defined Networks*. Toronto, ON, Canada: Ryerson Univ., 2015.

[40] M. Karakus and A. Durresi, "A survey: Control plane scalability issues and approaches in software-defined networking (SDN)," *Comput. Netw.*, vol. 112, pp. 279–293, Jan. 2017.

[41] P. E. Heegaard, B. E. Helvik, and V. B. Mendiratta, "Achieving dependability in software-defined networking—A perspective," in *Proc. 7th Int. Workshop Reliable Netw. Design Modeling (RNDM)*, Oct. 2015, pp. 63–70.

[42] K. S. Sahoo, D. Puthal, M. Tiwary, J. J. P. C. Rodrigues, B. Sahoo, and R. Dash, "An early detection of low rate DDoS attack to SDN based data center networks using information distance metrics," *Future Gener. Comput. Syst.*, vol. 89, pp. 685–697, Dec. 2018.

[43] J. Boite, P.-A. Nardin, F. Rebecchi, M. Bouet, and V. Conan, "Statesec: Stateful monitoring for DDoS protection in software defined networks," in *Proc. IEEE Conf. Netw. Softwarization (NetSoft)*, Jul. 2017, pp. 1–9.

[44] P. Kumar, M. Tripathi, A. Nehra, M. Conti, and C. Lal, "SAFETY: Early detection and mitigation of TCP SYN flood utilizing entropy in SDN," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 4, pp. 1545–1559, Dec. 2018.

[45] S. M. Mousavi and M. St-Hilaire, "Early detection of DDoS attacks against software defined network controllers," *J. Netw. Syst. Manag.*, vol. 26, pp. 573–591, Jul. 2018.

[46] J. Ye, X. Cheng, J. Zhu, L. Feng, and L. Song, "A DDoS attack detection method based on SVM in software defined network," *Secur. Commun. Netw.*, vol. 2018, pp. 2–5, Apr. 2018.

[47] I. Chawla, "DDoS Attacks in Cloud and Mitigation Techniques," *Int. J. Innov. Sci., Eng. Technol.*, vol. 2, no. 7, pp. 596–600, 2015.

[48] M. H. H. Khairi, "A review of anomaly detection techniques and distributed denial of Service (DDoS) on software defined network (SDN)," *Eng., Technol. Appl. Sci. Res.*, vol. 8, no. 2, pp. 2724–2730, 2018.

[49] Y. Zhao, W. Zhang, Y. Feng, and B. Yu, "A classification detection algorithm based on joint entropy vector against application-layer DDoS attack," *Secur. Commun. Netw.*, vol. 2018, pp. 1–8, Aug. 2018.

[50] A. S. Syed Navaz, V. Sangeetha, and C. Prabhadevi, "Entropy based anomaly detection system to prevent DDoS attacks in cloud," *Int. J. Comput. Appl.*, vol. 62, no. 15, pp. 8887–8975, 2013.

[51] Q. Yan and F. R. Yu, "Distributed denial of service attacks in software-defined networking with cloud computing," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 52–59, Apr. 2015.

[52] Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 602–622, 1st Quart., 2016.

[53] S. Deore and A. Patil, "Survey denial of service classification and attack with protect mechanism for TCP SYN flooding attacks," *IRJET*, vol. 3, no. 5, pp. 1–4, 2016.

[54] M. N. Rajkumar, "A survey on latest DoS attacks: Classification and defense mechanisms," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 1, no. 8, pp. 1847–1860, 2013.

[55] M. Nugraha, I. Paramita, A. Musa, D. Choi, and B. Cho, "Utilizing OpenFlow and sFlow to detect and mitigate SYN flooding attack," *J. Korea Multimedia Soc.*, vol. 17, no. 8, pp. 988–994, Aug. 2014.

[56] Y. Hayashi, J. Yong Zhen, S. Nishiyama, and A. Misawa, "Method for detecting low-rate attacks on basis of burst-state duration using quick packet-matching function," in *Proc. IEEE Int. Symp. Local Metrop. Area Netw. (LANMAN)*, Jun. 2017, pp. 7–8.

[57] B. B. Gupta, R. C. Joshi, and M. Misra, "Defending against distributed denial of service attacks: Issues and challenges," *Inf. Secur. J., A Global Perspective*, vol. 18, no. 5, pp. 224–247, Nov. 2009.

[58] A. A. Aizuddin, M. Atan, M. Norulazmi, M. M. Noor, S. Akimi, and Z. Abidin, "DNS amplification attack detection and mitigation via sFlow with security-centric SDN," in *Proc. 11th Int. Conf. Ubiquitous Inf. Manage. Commun. IMCOM*, 2017, pp. 1–7.

[59] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 2, p. 39, 2004.

[60] S. Scott-Hayward, G. O'Callaghan, and S. Sezer, "SDN security: A survey," in *Proc. IEEE SDN for Future Netw. Services (SDN4FNS)*, Nov. 2013, pp. 1–7.

[61] P. Dong, X. Du, H. Zhang, and T. Xu, "A detection method for a novel DDoS attack against SDN controllers by vast new low-traffic flows," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6.

[62] M. T. Manavi, "Defense mechanisms against distributed denial of service attacks : A survey," *Comput. Electr. Eng.*, vol. 72, pp. 26–38, Nov. 2018.

[63] N. Z. Bawany, J. A. Shamsi, and K. Salah, "DDoS attack detection and mitigation using SDN: Methods, practices, and solutions," *Arabian J. Sci. Eng.*, vol. 42, no. 2, pp. 425–441, Feb. 2017.

[64] B. H. Al-Mafrachi, "Detection of DDoS attacks against the SDN controller using statistical approaches," M.S. thesis, Wright State Univ., Dayton, OH, USA, 2017.

[65] J. Cui, J. He, Y. Xu, and H. Zhong, "TDDAD: Time-based detection and defense scheme against DDoS attack on SDN controller," in *Proc. Australas. Conf. Inf. Secur. Privacy*, 2018, pp. 649–665.

[66] Y. Cui, L. Yan, S. Li, H. Xing, W. Pan, J. Zhu, and X. Zheng, "SD-Anti-DDoS: Fast and efficient DDoS defense in software-defined networks," *J. Netw. Comput. Appl.*, vol. 68, pp. 65–79, Jun. 2016.

[67] R. Wang, Z. Jia, and L. Ju, "An entropy-based distributed DDoS detection mechanism in software-defined networking," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, Aug. 2015, pp. 310–317.

[68] X. Huang, X. Du, and B. Song, "An effective DDoS defense scheme for SDN," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[69] J. Mao, W. Deng, and F. Shen, "DDoS flooding attack detection based on joint-entropy with multiple traffic features," *Proc. 17th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. 12th IEEE Int. Conf. Big Data Sci. Eng.*, Aug. 2018, pp. 237–243.

[70] M. Şimşek and A. Şentürk, "Fast and lightweight detection and filtering method for low-rate TCP targeted distributed denial of service (LDDoS) attacks," *Int. J. Commun. Syst.*, vol. 31, no. 18, Dec. 2018, Art. no. e3823.

MOHAMMAD A. ALADAILEH received the B.S. degree in computer science from Mutah University and the M.Sc. degree in information technology from Universiti Utara Malaysia (UUM), in 2016. He is currently pursuing the Ph.D. degree with the National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia (USM). His research interests include computer networks, network security, and software defined networks (SDN).

MOHAMMED ANBAR (Member, IEEE) received the Ph.D. degree in advanced computer network from University Sains Malaysia (USM). He is currently a Senior Lecturer with the National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia. His current research interests include malware detection, web security, intrusion detection systems (IDS), intrusion prevention systems (IPS), network monitoring, the Internet of Things (IoT), and IPv6 security.

IZNAN H. HASBULLAH received the B.Sc. degree in electrical engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA. He is currently pursuing the M.Sc. degree in advanced network security. He worked as a Software Developer, the Research and Development Consultant, a CTO, and a Network Security Auditor prior to joining the National Advanced IPv6 Centre (NAv6), in 2010, as a Research Officer. His research interests include unified communication, network security, network protocols, and next generation networks.

YUNG-WEY CHONG (Member, IEEE) is currently a Senior Lecturer with the National Advanced IPv6 Centre of Excellence, Universiti Sains Malaysia, where she has been a Faculty Member, since 2012. She worked at telecommunication industry before joining USM. Her research interests include Industry Revolution 4.0, ranging from embedded systems, wireless communication, cloud computing, and software defined networking. She is a Committee Member of SOI Asia (www.soi.asia), a project that utilizes satellite-based Internet to support interactive multimedia communications between partner universities. She has collaborated actively with international and local researchers in multi-disciplinary projects. She has been involved in many collaborative research projects financed by various instances including the European Commission, the Royal Academy of Engineering, U.K., the National Information Communication Technology, Japan, and the National Science Foundation, South Korea.

YOUSEF K. SANJALAWE is currently pursuing the Ph.D. degree with the National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia. His research interests include data mining, information retrieval, the IoT, and cloud computing.