INTERNATIONAL JOURNAL OF EDUCATION AND INFORMATION TECHNOLOGIES
DOI: 10.46300/9109.2021.15.9 Volume 15, 2021

Information theory-based approaches to detect DDoS attacks on software-defined networking controller a review

Mohammad A. Aladaileh[1,*], Mohammed Anbar[1,*], Iznan H. Hasbullah [1], Yousef K. Sanjalawe[2]
[1]National Advanced IPv6 Centre of Excellence, Universiti Sains Malaysia, Penang, Malaysia
[2]Computer Sciences Department, Northern Border University, Ar'ar, the Kingdom of Saudi Arabia

Received: February 26, 2021. Revised: April 13, 2021. Accepted: April 15, 2021.
Published: April 22, 2021.

Abstract—The number of network users and devices has exponentially increased in the last few decades, giving rise to sophisticated security threats while processing users' and devices' network data. Software-Defined Networking (SDN) introduces many new features, but none is more revolutionary than separating the control plane from the data plane. The separation helps DDoS attack detection mechanisms by introducing novel features and functionalities. Since the controller is the most critical part of the SDN network, its ability to control and monitor network traffic flow behavior ensures the network functions properly and smoothly. However, the controller's importance to the SDN network makes it an attractive target for attackers. Distributed Denial of Service (DDoS) attack is one of the major threats to network security. This paper presents a comprehensive review of information theory-based approaches to detect low-rate and high-rate DDoS attacks on SDN controllers. Additionally, this paper provides a qualitative comparison between this work and the existing reviews on DDoS attack detection approaches using various metrics to highlight this work's uniqueness. Moreover, this paper provides in-depth discussion and insight into the existing DDoS attack detection approaches to point out their weaknesses that open the avenue for future research directions. Meanwhile, the finding of this paper can be used by other researchers to propose a new or enhanced approach to protect SDN controllers from the threats of DDoS attacks by accurately detecting both low-rate and high-rate DDoS attacks.

Keywords — Software-defined networking; SDN controller; DDoS attack; information theory; entropy; Rényi joint entropy; and joint entropy.

I. Introduction

The last few decades have witnessed a proliferation and rapid growth of information and communication technology which spurred the astronomical increase of network traffic, which added more complexity to the operations to process the massive data [1]. Soon, the existing conventional network architecture might not be able to cope with the tremendous amount of network traffic, leading to security and privacy issues as some packets may be lost or dropped in transit. Many researchers attempted to solve this issue, including proposing a new revolutionary network architecture, such as software-defined networking (SDN) designed to be more secure and flexible, easier to manage, and programmable [2], [3]. SDN changes the approach to managing the network and provides innovative solutions to conventional network problems. Consequently, several factors differentiate SDN from the traditional networks. One of the main differences is separating the control plane from the data plane. The separation provides the SDN with the ability to centrally and flexibly manage the entire network using a centralized controller [4]–[6]

On the other hand, the SDN controller deals with the network traffic packets either in a proactive or reactive mode. The SDN controller operating in proactive mode has a more significant effect on the SDN performance than the reactive mode in protecting the SDN network from malicious attacks because the rules are pre-installed in the switch table (flow rule) to process the packets [7]. In contrast, in the reactive mode, the rules will only be created and installed to the switch whenever new incoming packets do not have matching rules in the switch table. Furthermore, SDN helps data centers to control costs by increasing the efficiency of managing network traffic. Cisco reported in 2018 that a large percentage of data centers would

adopt SDN globally, partially or entirely, to manage their network traffic flows soon [8], as shown in Fig. 1.

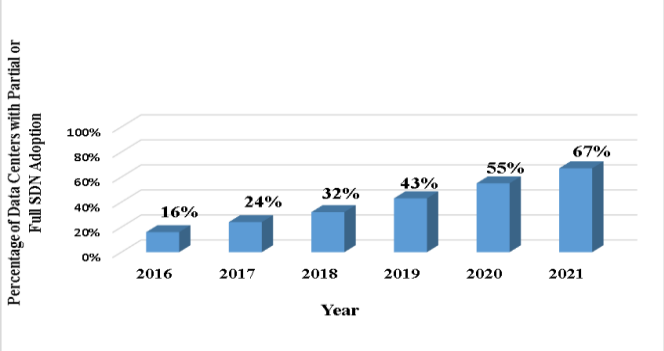

Fig. 1 adoption of SDN from 2016 to 2021

Fig. 1 shows that by 2021, most data centers will be using SDN technology since it makes the management and control of network traffic more efficient, thus less costly. This projection is a strong indicator of the importance of SDN in information technology and data exchange.

The controller simplifies network operations by utilizing the centralized control feature for improving the network through monitoring network devices and routing a flow path based on the flow entry rule or instruction in the switch's flow table. Furthermore, the SDN controller gathers the required information from the network packets for analysis to detect DDoS attacks. The controller is a crucial component in any effort to improve network performance. The controller plays different roles by using various modules to gather network traffic's statistical data and identify each part's tasks in the network [9]. Notably, the controller simplifies network operations by utilizing the centralized control feature for improving the network through monitoring network devices and routing a flow path according to the flow entry (rules/instructions) in the switch's flow table. Furthermore, the SDN controller collects the required information from the network packets for analysis to detect DDoS attacks.

Fig. 2 illustrates the SDN architecture's three-layer overview [10].

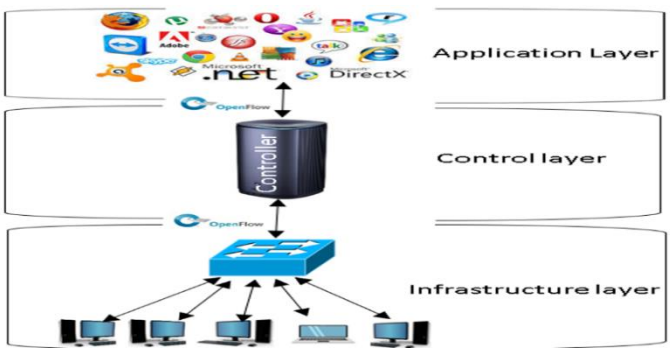

Fig. 2 SDN architecture

The SDN depends on a centralized controller to control the entire network by allowing the applications to have a network-wide view, establishing centralized visibility to manage the network traffic flow [11] . Moreover, it can virtualize the entire network infrastructure to simplify configuring and managing the network [12].

However, the centralized controller also faces challenges and issues that affect network performance in terms of security, reliability, and scalability [13].

The DDoS attack is one of the security challenges to the SDN network that could bring down the entire network, denying the legitimate user access to network services or resources. DDoS is one of the most common types of attacks that target the SDN controllers [14], [15]. In a DDoS attack, attackers flood the network or the controller with a large volume of traffic to the point where the controller's resources are exhausted and unable to process any more incoming packets. Therefore, many researchers have proposed approaches to tackle DDoS attack issues. Information theory-based approaches are considered the most common approaches for detecting DDoS attacks on SDN networks.

The contributions of this paper are: (i) a comprehensive review of various information theory-based approaches to detect low-rate and high-rate DDoS attacks on the SDN controller; (ii) qualitative comparison of this work with existing reviews related to similar fields in terms of information theory used and classifies the detection approaches; and (iii) in-depth discussion and insight on the existing detection approaches' gaps.

The rest of the paper is structured as follows. Section II presents the background on the SDN controller and security issues in SDN, including DDoS attacks' impacts on the SDN controller. Section III discusses the result of a qualitative comparison of existing reviews on the existing detection approach of low-rate and high-rate DDoS attack traffic on the SDN controller. Section IV explains the importance of information theory. Section V discusses the approaches to detect DDoS attacks on the SDN controller by studying the technique used and analyzing the finding in terms of strength and the drawbacks of each approach. Section VI discusses detection approaches gaps for detecting low-rate and high-rate DDoS attacks on the SDN controller. Finally, Section VII and VIII provide future research directions and conclusions, respectively).

II. Background

A. Software-Defined Networking

The SDN offers many advantageous features, such as network programmability, that enable the SDN networks to be deployed quickly and managed dynamically compared to traditional networks, which take longer to deploy and harder to manage [16], [3]. The SDN depends on a centralized controller to control the entire network. It enables the applications to have a network-wide view by establishing centralized visibility to manage the network traffic flow [4]. Moreover, it also provides the capability to virtualize the entire network infrastructure to simplify configuring and managing the network. SDN promises to reduce the network complexity by dividing the data plane

from the control plane [16], [17]. Table 1 presents the benefit of the SDN versus the traditional network.

Table 1. SDN vs. traditional network

Criteria	SDN	Traditional Network
Network management	Easy	Difficult
Global network view	Easy	Difficult
Maintenance cost	Low	High
Time for update/error handling	Quick	Slow
Attack detection and mitigation	Easy	Difficult
Controllers' and applications' authenticity	Important	Not Applicable
Integrity and consistency of forwarding table and network state	Important	Important
Availability of controller	Important	Not Applicable

SDN removes the controllability feature from the data plane. Instead, it puts it at the control plane, allowing better control of the network configurations to improve network performance and driving future innovations on the network architecture and security [18]. Moreover, it provides real-time network status updates, making efficient control and flow handling procedures possible while ensuring the control plane remains flexible and intelligent [19].

The SDN controller handles many vital functions in the network, such as configuring the flow table, monitoring networking devices via secure connections, and updating the flow table's rules or instructions in the infrastructure layer (switch's table) to identify new traffic flow [20]. Also, the controller could position itself between the infrastructure layer and application layer to manage all traffic flow via open API southbound, northbound, and east/westbound interfaces [21], [22]. By assuming a manager's role, the controller decides whether a traffic flow is normal or abnormal based on the controller's network statistics used as a baseline input (information) to an attack detection method.

B. Security Issues in Software-Defined Networking

The SDN is a novel network architecture that provides a better solution to overcome the traditional network drawbacks. In particular, SDN provides flexibility in programming the network by isolating the control plane from the data plane [7], [23]. This isolation allows efficient network management and provides network operators with the flexibility to program their network and control, and permits new approaches to solve the conventional networks' problems.

Any new network will encounter many security issues, especially with the increase in the Internet's use annually. Besides, no network defense technique can be guaranteed to be safe, including the SDN, because the controller is a very attractive target for attackers that aim to breakdown the entire SDN network operations. Akamai's report showed a high increase in DDoS attacks in the first three months of 2016, reaching 126% compared to the same period a year prior [24].

The SDN platform could potentially bring many new security challenges, especially since its centralized controller is the most critical component of the SDN network that acts as its operating system. Thereby, any unaddressed threat to the centralized controller may lead to a network breakdown. Therefore, a DDoS attack's main objective is to overwhelm the controller until its resources are exhausted by flooding the network with spoofed IP packets, thereby congesting the controller that causes the entire SDN network to collapse [25], [26]. Since security is an essential feature of any communication network, coupled with the SDN controller's importance to the SDN architecture, the SDN controller's security has attracted network security researchers' attention [27].

The centralized controller could play an essential role in securing the networks if configured according to network security best-practices to solve security issues [26], [28] and [29]. Thus, the SDN provides many advantages that contribute to protecting the network from malicious attacks that exploit SDN features. However, at the same time, some of the features also attract the attention of adversaries intending to disrupt or break down the network by targeting the SDN controller, whether directly or indirectly, to deny legitimate users access to the network or network services [30]. On the other hand, the controller becomes a single point of failure [31], which directly affects the network's performance, scalability, reliability, and security. Table 2 shows significant SDN related issues and challenges that affect the controller from the exploitation of the centralized control feature that affects the network performance or even breakdown the whole network. These issues call for more effort to be made to prevent and protect the SDN controller from breakdown.

Table 2. Issues and challenges an SDN controller

No	Issue	Description
1.	Reliability	The reliability between the controller and the switch by extent secure the data exchange and mitigates controller failures' impact.
2.	Scalability	The ability to increase the number of hardware devices connected to a controller or the number of flow requests processed by a single controller.
3.	Performance	Security threats remain to affect the SDN controller performance, considered one of the critical issues in SDN to locate security threats before the entire network's collapse by a new type of network attack.
4.	Security	The decoupling of the data plane from the control plane provides centralized control, attracting new types of attacks (DDoS) that target the network.

Several factors threaten SDN security [5]. The most important factors that affect SDN security are faked traffic

flows, exploited SDN vulnerabilities, exploited centralized controllers, exhaustion of controller resources, and malicious attacks on the SDN controller. One of the most widespread threats against the SDN controller is the DDoS attack.

C. Distributed Denial of Services Attack

Security and privacy issues are a growing concern in SDN, particularly on the SDN controller. The DDoS attack is considered one of the most serious threats to the SDN because of its devastating effects on the entire network. DDoS attacks perpetrator's primary goal is to deny legitimate users' access to network services or network resources [32], [33]. Indeed, DDoS attack principles depend on the attackers to launch attacks. The attacker sends massive numbers of packets towards a destination IP address from a unique source IP address to flood the entire network, denying the legitimate users from reaching their destinations [23]. Typically, DDoS attacks spoofed the source IP address of the attacking hosts by using many different faked source IP addresses to launch the attack without the risk of being discovered [34].

In an SDN network, the hosts (users, computers, nodes) communicate with each other by exchanging data through switches. Every SDN switch has flow table entries that contain flow entries (instructions) for forwarding matching incoming packets to their destinations. However, suppose there are no matching instructions in the flow table entry (switch table). In that case, the switch will forward these packets to the controller using OpenFlow protocol for further processing to obtain new instructions or new rules [35]. The controller treats all received packets as new incoming packets, so the controller will either drop the packets or send them to their destination based on the controller's statistics. The attackers exploit this feature by sending a flood of distinct packets with spoofed source IP addresses to overwhelm the controller and bring it down.

According to [36], DDoS attack has many types, such as ICMP flooding, TCP flooding, and UDP flooding. Each DDoS type has its way of sending the attack traffic to the target destinations. The DDoS attack can be made more efficient and robust by varying its attack traffic rates (low or high), the number of targets (single or multiple victims), or the number of attackers (one or multiple sources) launching the attack.

As mentioned, the attacks' target is to destroy the network relying on the traffic type characteristics. Therefore, the attackers' targets in this research are damaging the controller. The attackers use different types of attack traffic to launch their attack, which is difficult for the controller to distinguish the attack traffic from the regular traffic.

However, the enormous numbers of packets contribute to exhaust controller resources, making it challenging to handle all these packets. Meanwhile, if the controller crashes, the entire SDN will crash too. This drawback is considered a single point of failure in SDN. It is a security vulnerability, and the DDoS attack perpetrator exploits this vulnerability to launch more attacks that cause massive damages. Thus, triggering spoofing DDoS attacks prevents legitimate users from accessing the network resources by sending an enormous number of packets by spoofing the source IP addresses to distinct victims' destinations. The attacker launched numerous spoofed attack packets towards the target (SDN controller) using agents or botnets to deny legitimate users of network services [37]. This paper's primary goal is to classify DDoS attack detection approaches, which depend on Information Theory, by categorizing them according to their method, strengths, and weaknesses based on their category.

III. QUALITATIVE COMPARISON WITH EXISTING REVIEWS ON DETECTION APPROACHES TO DETECT DDOS ATTACKS ON SDN CONTROLLER

This section provides a qualitative comparison to benchmark this work with other existing information theory-based detection approaches designed to detect low-rate and high-rate DDoS attacks on the controller using different metrics to highlight this study's uniqueness, as shown in Table 3. The used metrics are: (i) number of theories used, (ii) approaches classified. These metrics are defined based on intensive study of many existing detection approaches. This comparison aims to understand the critical issues related to detecting DDoS attacks on the controller to find an efficient detection approach. Furthermore, it could be a guideline for future researchers in a similar domain. This review is compared with three existing reviews from [10], [38] and [39].

Table 3. Qualitative comparison with the existing reviews

Criteria		This work	[10]	[38]	[39]
Theory used	Entropy	9	6	4	3
	Joint Entropy	2	-	-	-
	Rényi Entropy	2	-	-	-
Approaches classified.		yes	yes	yes	yes
The number of approaches depended on a single packet header feature.		11	9	17	14
The number of approaches depended on multiple packet header features.		2	2	11	10
The number of detection approaches deployed on the controller.		13	5	19	11

IV. INFORMATION THEORY

Most existing detection approaches rely on methods to detect either low-rate or high-rate DDoS attacks, but none consider both [32], [40] and [41]. Therefore, there are no approaches that can detect various rates of DDoS attacks on SDN controllers (low-rate and high-rate attack traffic) with high detection accuracy and low false-positive rate.

An information theory-based approach attempts to detect the different DDoS attack rates in the network traffic flow by relying on statistical methods to calculate incoming packets' randomness in the network traffic flows. Incoming packets' randomness is one of the DDoS attacks' indicators, which is due to the attackers continuously sending packets with spoofed IP addresses towards the SDN controller. Furthermore, information theory-based DDoS attack detection is more effective in terms of detection accuracy and false-positive rate.

V. INFORMATION THEORY-BASED ALGORITHMS TO DETECT DDOS ATTACKS ON SDN CONTROLLER

This section discusses information theory-based algorithms commonly used to detect low-rate or high-rate DDoS attacks on SDN controllers based on analyzing network traffic statistics from traffic flows. The subsection below explains information theory such as entropy, Rényi entropy, and joint entropy. This paper is the first attempt at classifying some of the existing DDoS attack detection approaches that have used information theory-based algorithms, as shown in Fig. 3.

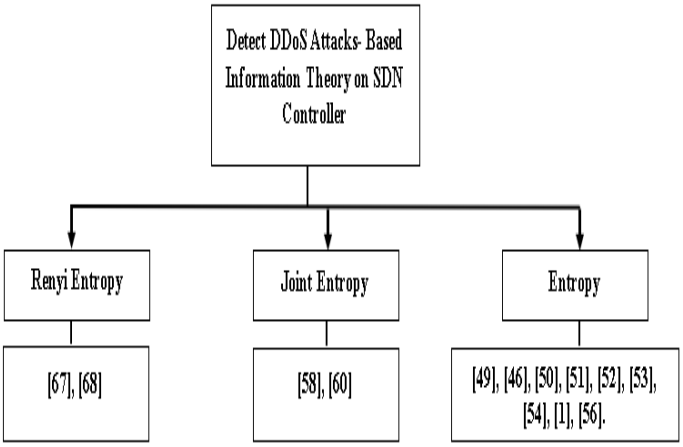

Fig. 3 Classification of DDoS detection approaches

A. Entropy

Entropy is a method used to measure a random variable's probability within a specific time [42] and [43]. The entropy method-based detection approach focuses on calculating the randomness of packets in the network using different packet header features of the network traffic flows, such as source IP address, destination IP address, source port, etc. [44]. Furthermore, entropy-based metrics are used for network traffic flows analysis based on the packet header features to detect abnormal packets in the traffic flows. A maximum entropy occurs when all packets are equally distributed to all host destinations, whereas minimum entropy occurs when all packets are destined toward a specific destination host [23]and [45]–[48]. Equation 1 shows the general formula of the entropy method.

$$H = -\sum_{i=1}^{n} p_i log_2 p_i \quad (1)$$

Where p_i is the probability of a specific feature from the packet header features, n is the total number of features in the network traffic within a particular time, and H is the entropy value.

Indeed, many security techniques rely on the entropy method to detect DDoS attacks against the SDN network [49]. They used the entropy method to detect attacks against the controller, which depends on the relative frequencies to the destination IP address feature to measure the incoming packets' randomness within a fixed window size. [48]–[50] used the entropy method to define the packet randomness in network traffic as an indicator of the DDoS attack presence. Thus, the entropy method has a high ability to detect and issue an alarm on the presence of a DDoS attack in the network traffic flow.

A lightweight detection approach was proposed based on the entropy method to detect DDoS attacks against the controller early by analyzing the network traffic flow statistics by [51]. However, the proposed approach used a fixed threshold, leading to an increase in the false-positive rate and introducing extra overhead on the SDN controller in dealing with new incoming packets. Furthermore, the proposed approach is only able to detect the DDoS attacks that target a single victim.

An entropy method is used to propose an approach to detect DDoS attacks with high accuracy and low false-positive rate. It analyzes the gathered network traffic flows statistics from the switch table to identify the packets' randomness in the network traffic flow [48]. However, the proposed approach relies on the SDN switch to collect the traffic statistics instead of the SDN controller, which delays the controller's response time to detect the attack due to extra efforts needed to gather the data. It is also unable to detect low-rate DDoS attacks that target multiple victims and use a fixed threshold that increases the false-positive rate and lowers the detection rate.

Furthermore, [52] proposed an entropy-based DDoS defense mechanism (EDDM) against the SDN controller. EDDM tries to keep legitimate packets from being dropped during flash crowd events and thus prevents denial of service to legitimate users on the network by tracing the attack traffic within the flash crowd traffic. The EDDM comprises three phases to detect DDoS attacks based on traffic statistics collected within a specific window size to calculate the entropy value. The entropy value is compared with a static threshold in the sFlow to display the traffic packet rate that targets a particular victim. However, there is a delay in processing new packets, and also

the detection accuracy is low for attacks that target multiple victims.

[53] proposed a scheme that depends on a security gateway that uses the entropy method to detect DDoS attacks. The security gateway receives unmatched packets from the switches for a further check to see if the entropy value is lower than the threshold value. If true, the security gateway creates new rules without making any decision about the new packets. Then, the security gateway sends the results related to the new packet to the controller to decide if the packet will be forwarded to their intended destination or dropped. The scheme depends on entropy value calculation that uses three features, such as protocol type, destination IP, and source IP address. However, it is time-consuming to process new network packets, which explains low detection accuracy, especially with low rate attack traffic. Besides, the scheme also relies on a static threshold.

[54] proposed a FADM approach that depends on traffic flow analysis of the network traffic statistics collected by the controller's sFlow agent. The proposed approach adopted the entropy method to calculate the packet features' probabilities to measure traffic changes. Additionally, a machine learning algorithm (SVM) is used for detecting DDoS attacks. However, the false-positive rate increases significantly when the DDoS attack is triggered by a burst of attack traffic within a short time that targets multiple victims.

A detection approach called SAFETY was proposed for early detection and mitigation of TCP SYN flooding by adopting the entropy theory to calculate the randomness of the destination IP and TCP flags. The proposed approach adapted a threshold based on the variation of traffic rates and types. The threshold is compared with the entropy value to detect DDoS attacks against the controller [55]. However, SAFETY only handles a single victim host; and will render the network unstable if multiple concurrent victims are involved.

Proposed an efficient and lightweight approach to detect DDoS attacks against the SDN controller (EDDSC) that depends on the entropy method [56]. The proposed approach relies on just one packet header feature (destination IP address) to analyze the network traffic flow statistics. The traffic flow statistics are used to calculate the packet feature's probability in the traffic flow, which is then fed to the entropy method to calculate the packets' randomness in the network traffic flows. Moreover, it depends on a fixed threshold to decide if the traffic flows contain any attack traffic by comparing the entropy value with the threshold value. Although the proposed method attempts to detect DDoS attacks early, its limitations reduce its reliability due to reliance on a fixed threshold, which is not suitable for detecting DDoS attacks with variable attack rates, thus increases the false-positive rate. Also, the attackers launch the DDoS attacks from a single host (single attacker) to multiple victims resulting in a reduced detection rate of the low-rate DDoS attack and increase the false-positive rate. Furthermore, the proposed approach only depends on a single feature to collect network traffic flow statistics, rendering the collected traffic flow data insufficient to distinguish the attack traffic from the regular traffic.

A new statistical-based approach to detect DDoS attacks was proposed by [1]. The proposed approach is designed to detect the presence of DDoS attacks accurately, reduce false-positive flow rates, and minimize the complexity of targeting SDN controllers according to a statistical analysis of packet features depending on the entropy method. However, the controller is still overwhelmed by incoming packets because the proposed technique works in the controller and needs time to process all incoming packet flows.

A stateful model was adopted to protect the end-host from DDoS attacks in [57]. They proposed a novel approach based on in-switch processing capabilities to monitor, detect, and mitigate DDoS to avoid risking controller overload or failure during the processing of new packets. The proposed key objective a quick reaction time and reduction of the overhead include on the controller channel by communication between a switch and its controller. The proposed relies on the entropy-based algorithm with such monitoring features (source IP, destination IP, Source Port, Destination Port). StateSec detects DDoS and port scan with high accuracy. Furthermore, it achieves highly accurate detection and limiting the controller overhead. However, StateSec approach is neither efficient nor scalable.

B. Joint Entropy

As mentioned, the Shannon entropy formula depends on a single feature for detecting the DDoS attacks, and it ignored other packet header features. Otherwise, there is a new method known as Joint-entropy which depends on multiple features for detecting DDoS attacks, which increases the accuracy of detecting DDoS attacks as compared with the entropy method, such as [58]–[60]

Joint entropy is a method used to measure the probability of random variables that depends on two packet header features, such as source IP *(X*) and source port *(Y),* and their calculated probabilities *P(X,Y)* within a specific time. For example, the attack detection approaches depend on joint entropy for detecting DDoS attacks by calculating the distribution of two random variables in the network traffic flow $p(x_i y_j)$. So, the joint entropy formula is defined by Equation 2.

$$H(X) = -\sum_{i=1}^{N}\sum_{j=1}^{M} p(x_i y_i) log_2 p(x_i y_j) \qquad (2)$$

Where $H(x)$ is joint entropy, $p(x_i y_j)$ is the probability of the event (X= X_i , Y= y_j), $i = 1, 2, 3, ..., N$ and $j = 1, 2, 3, ...M$. Thus, the proportion of the number of frequencies an event can occur relative to the total number of possible outcomes. It is the probability of that event happening. Hence, taking the form of a positive fractional number between 0 and 1, and hence whose logarithm will always be a negative number due to the probabilities that we are dealing with are all positive values. [58] proposed an approach that detects DDoS

attacks based on the joint entropy method and conditional entropy. The proposed approach depends on two features (attributes), unlike the entropy method that relies only on a single feature. Thus, the false-positive rate of the approaches that use the joint entropy method will be lower than the entropy method.

A joint-entropy method was adopted in a DDoS attack detection approach by [59]. It used information theory that depends on several packet header features (e.g., flow duration, source IP address, packet length, and destination port) to calculate packet traffic flow randomness using a joint entropy method. The proposed method is effective in reducing the false-positive rate and increase the detection accuracy. However, it still suffers from low rate DDoS attack detection and cannot detect DDoS attacks on a controller with different attack traffic rates (low and high), and used a static threshold.

Additionally, Kalkan et al. [61] proposed a new security scheme that depends on Joint Entropy Security Scheme (JESS). The JESS comprises three stages: nominal, preparatory, and active mitigation stage. The first stage collects statistical information about the network traffic features in a non-attack period. The second stage calculates the controller's incoming packets' randomness by joint entropy method and compares with a threshold to detect DDoS attack in the network traffic. The third stage mitigates attack traffic by following instructions from the controller for action based on the flow table. However, the proposed approach added overhead on the controller and uses a static threshold, which increases the false-positive rate and decreases detection accuracy for network traffic flows with different attack traffic rates (low and high).

C. Rényi Entropy

The Rényi Entropy formula is generalized based on the Shannon entropy method [62]. The Shannon entropy assumes a trade-off between contributions from the main mass of the distribution and the tail. This research used two parameterized Shannon entropy generalizations to control the trade-off. These two parameters (Rényi and Tsallis) are derived from Kolmogorov-Nagumo [63]. Consequently, Rényi entropy method performance is considered one of the best methods to detect DDoS attacks that depend on one feature of the packet header features. Equation 3 shows the Rényi entropy formula.

$$H_{R\alpha}(x) = log_a \left(\sum_{i=1}^{N} p(x_i)^{\alpha} \right) \tag{3}$$

Where α is a positive parameter and exposes the main mass (concentration of events that occur often) and $p(x_i)$ the probability that x event will occur from a total number of events within a specific time.

This research assumes that the Rényi entropy's value satisfies the same postulates as the Shannon entropy in the following relations as shown below.

$H_{R\alpha_1} \geq H_{R\alpha_2}(x)$ where $\alpha_1 < 1$ and $\alpha_2 > 1$. As one can see Rényi converge to Shannon for $\alpha \to 1$.

$$\lim_{\alpha \to 1} \frac{1}{2} log_2 \left(\sum_{i=1}^{N} p(x_i)^{\alpha} \right) = H(x) = -\sum_{i=1}^{N} P(x_i) log_2 (x_i) \tag{4}$$

An information distance and generalized entropy metrics are proposed by [64] to detect low rate DDoS attacks with lower false-positive rate than the detection approaches that depend on the Shannon entropy method by differentiating between the legitimate traffic and the attack traffic. However, the method does not consider the detection of high-rate DDoS attacks with flash crowd events. [65] uses the Rényi entropy method combined with the EWMA theory for detecting DDoS attacks. In the proposed approach, the Rényi entropy method was more effective than the Shannon entropy method in terms of detection accuracy. Moreover, Rényi entropy can calculate network traffic intensity to differentiate normal traffic and abnormal traffic (attack).

[66] proposed an effective method to detect DDoS attacks based on the Rényi entropy method to distinguish DDoS attacks from flash crowd events. Rényi entropy methods used depend on the time series to analyze network traffic similarities based on particular traffic features.

Wang et al. proposed an approach called HHM-R to improve the detection of low-rate DDoS attacks against the controller, which combines a hidden Markov model and the Rényi entropy method [67]. The proposed approach comprises four modules: data preprocessing, model initializing, model training, and model detecting, which consume more time to detect Low-rate DDoS attacks. However, the false-positive rate increases, and the detection accuracy decreases when the Low-rate DDoS attacks target multiple victim hosts.

Moreover, [68] proposed an approach that combines information distance with the generalized (Rényi) entropy for detecting low-rate DDoS attacks against the SDN controller. The probability distribution is considered the DDoS attack's detection metrics by setting a specific window size for the incoming packets. It then periodically extracts the packet features from the flow table (switch table). The difference in the probability distributions is the indicator of a DDoS attack in the network, but the false-positive rate increases under different attack traffic rates. The proposed approach experiments showed that the generalized entropy combined with the information distance accurately detects low-rate DDoS attacks. However, it is difficult to set the dynamic threshold because of different attack traffic rates in the traffic flow and the dependency of the proposed approach on the switch table to extract the traffic statistics instead of the controller, which may miss some information on the traffic flow. Furthermore, the proposed approach depends on one packet feature of the packet header features to collect traffic flow statistics, which increases the false-positive detection rate.

Also, Zhai et al. proposed an approach to detect and mitigate DDoS attacks using the Rényi entropy method to calculate

packets' randomness in network traffic flows by relying on one header packet feature to analyze network traffic behavior [69]. However, it used a fixed IP address to trigger attacks toward the SDN controller, and the approach only deals with high-rate DDoS attacks (fixed traffic rate).

VI. RESEARCH GAPS AND DISCUSSION

The previous sections provided a comprehensive review of existing DDoS attack detection approaches on the SDN controller to highlight each attack detection approach's strengths and weaknesses. Many related works in the literature address multiple DDoS detection requirements. Yet, they are still plagued with many issues, such as low detection rate and low false-positive rate due to many reasons, such as (i) reliance on a single packet header feature and (ii) usage of a static threshold. Although some of the approaches rely on two packet header features, they are either focused on detecting low-rate DDoS attacks or high-rate DDoS attacks, thus failing to detect DDoS attacks when varying rate attacks occurred simultaneously in the network.

The complexity of DDoS attacks and novel ways to deny legitimate users access to network services increased the attack traffic intensity. This type of attack is called a high-rate DDoS attack, which commonly focuses on a single victim to maximize the attack's effectiveness. As a result, many of the existing approaches were proposed to detect this type of attack, such as in [70], [71] and [52].

Many existing approaches dealing with low-rate DDoS attack that targets a single victim have a good detection accuracy. However, some approaches, such as [54], [55] struggle when multiple victims are involved. Attackers might exploit this drawback to launch DDoS attacks against the SDN controller to disrupt or bring down the entire SDN network. Meanwhile, [56], [68] tried to secure the SDN controller from both low-rate and high-rate DDoS attacks regardless of the number of targeted victims with high accuracy and low false-positive rate. However, these approaches generally rely only on a single packet header feature to collect traffic flow statistics for DDoS attack detection, which negatively impacts detection accuracy. Also, these approaches may result in a substantial increase in the controller's resource consumption that increases the controller's overhead. Table 4 summarizes the existing attack detection approaches by presenting their strengths and weaknesses based on their category.

As mentioned earlier, most of the existing detection approaches of DDoS attacks against the SDN controller are designed only to detect either low-rate DDoS attacks or high-rate DDoS attacks. Thus, it is necessary to propose an approach to detect low-rate and high-rate DDoS attacks. However, the existing approaches that can detect DDoS attacks with varying attack rates still suffer from low detection accuracy, especially when the low-rate DDoS attacks are targetting multiple victims.

Table 4. Summary of DDoS attack on SDN controller detection Approaches

	Ref	Strengths	Drawbacks
ENTROPY	[51]	- Early detection of DDoS attacks. - Lightweight method.	- Static threshold. - Only detect DDoS attack that targets a single victim.
	[48]	- Early detect DDoS attacks. - Lightweight method.	- Static threshold. - Unable to detect low rate DDoS attack targeting multiple victims.
	[52]	- keep legitimate packets from being dropped during flash crowd events.	-Detection accuracy is low for attacks that target multiple victims.
	[53]	- Predicts the number of incoming packets (unmatched packets) - Avoids flooding the controller.	- Static threshold. - Low attack detection accuracy.
	[54]	- Classifies traffic flow as attack or normal. - Detects and mitigates DDoS attacks.	- Only handle low-rate attack. - False-positive rate increases during high traffic rate in short time targets multiple victims.
	[55]	- Detect an attack at the edge of switches.	- Detects attacks on a single victim only.
	[56]	- Early detect DDoS attacks. - Lightweight method.	- Static threshold. - Does not able to detect low DDoS attack traffic rate with a high accuracy ratio.
	[1]	- Lightweight method. - Early detect DDoS attacks.	- Static threshold. - Unable to detect different attack traffic rates.
	[57]	- Avoid overload and failing a controller processing.	- Does not able to detect low DDoS attack traffic rate with a high accuracy ratio.
JOINT ENTROPY	[59]	-High detection accuracy.	- Unable to detect DDoS attacks that have different attack traffic rates. - Static threshold.
	[61]	-Mitigate DDoS attacks.	-Static threshold. -High detection accuracy to different attack traffic rates.
RÉNYI ENTROPY	[68]	- Detects DDoS attacks targeting single victims efficiently. - Distinguish normal traffic from attack traffic.	- Difficult to set dynamic threshold. - Low detection accuracy when an attack targets multiple victims, specifically with a low rate DDoS attack.
	[67]	-Detect DDoS attacks on the data plane. - Mitigation DDoS attacks on SDN controller.	-Used a fixed IP address to launch DDoS attacks. - Detect high-rate DDoS attacks only.

Table 5 presents the research gaps in each existing detection approach.

Table 5. Gaps in existing DDoS detection approaches

Reference	Evaluation			Threshold		Features	
	False Positive	Detect a Low Attack Rate	Detect a High Attack Rate	Static Threshold	Dynamic Threshold	Single Feature	Multiple Features
[50]	√	X	√√	√√	-	√√	-
[47]	√√	XXX	√√	√√	-	√√	-
[51]	√√	XXX	√√	√√	-	√√	-
[52]	√√	XX	√√	√√	-	√√	-
[53]	√√	√√	√	√√	-	√ √ √	-
[54]	√√	XXX	√√	-	√√	√ √ √	-
[55]	√√	X XX	√√	√√	-	√ √ √	-
[1]	√	XXX	√√	√√	-	√√	-
[56]	√	XXX	√√	√√	-	√√	-
[58]	√√	√√	√√	-	-	-	√√
[60]	√√	√√	√√	√√	-	-	√√
[67]	√√	√√	√√	√√	-	√√ √	-
[68]	√√	√√	√√	-	√√ √	√√ √	-

√√√: highly efficient, √√: efficient, √: relatively efficient, XXX: highly inefficient, XX: inefficient, X: relatively inefficient, -: not considered/not addressed.

Table 5 lists the gaps or weaknesses of the existing DDoS detection approaches. The gaps are grouped based on the detection approach's evaluation (false-positive rate, low-rate attack, high-rate attack), the threshold used (static or dynamic), and the packet header feature selected (single or multiple). The majority of existing detection approaches that are highly efficient in detecting high-rate DDoS attacks that target a single victim depend on a single packet header feature and a static threshold value. The dependency on the single packet header feature and static threshold value renders them inefficient in detecting other DDoS attacks with a high detection rate and low false-positive rate. These gaps correlate with the challenges listed in Table 4.

The observation of the reviewed existing information theory-based approaches reveals the following:

- Some of the existing approaches rely on a single packet header feature to detect DDoS attacks against an SDN controller triggered from a single host and attack single or multiple targets with a low detection rate and high false-positive detection rate.
- Some of the existing approaches to detect DDoS attacks against SDN controller using two packet header features are incapable of detecting low-rate DDoS attacks on multiple targets to achieve a high detection rate and low false-positive rate.
- Some of the existing detection approaches rely only on static threshold values, making them inefficient in detecting DDoS attacks with variable traffic rates to achieve a high detection rate and a low false-positive rate.

Therefore, the future DDoS attack detection approach should address the abovementioned drawbacks of the existing approaches and fulfills the following requirements:

1. High accuracy: The DDoS attack detection process should consume minimal resources to avoid overloading the controller, thus reducing its effectiveness.
2. Low false-positive detection error: The approach should distinguish abnormal traffic from the normal or legitimate traffic with confidence regardless of the DDoS attack traffic rates (low or high).
3. Make a decision: The approach should decide on the existence of a DDoS attack against the SDN controller from abnormal traffic flow behavior based on the statistical data collected from the controller instead of the switches.
4. Able to detect both low-rate and high-rate DDoS attacks regardless of the number of victims (single or multiple).

VII. CONCLUSION

This paper reviewed information theory-based detection approaches to detect DDoS attacks on SDN controllers. It is noted that most existing information theory-based detection approaches have varying detection accuracy in detecting DDoS attacks against the SDN controllers. Most approaches achieved high detection accuracy in detecting high-rate DDoS attacks; however, the approaches' accuracy is lower when low-rate DDoS attacks are involved or multiple victims are targeted. This paper also identified the weaknesses of the existing approaches so that interested researchers may consider addressing them in their future research. Also, we intend to conduct a systematic literature review on information theory-based detection approaches for DDoS attacks against the SDN controller that cover a more comprehensive range of research.

Acknowledgment: This work was supported by Universiti Sains Malaysia under the external grant (Grant number 304/PNAV/650958/U154).

References

[1] M. Al-adaileh, M. Anbar, Y. Chong, and A. Al-ani, "Proposed Statistical-Based Approach for Detecting Distribute Denial of Service Against the Controller of Software Defined Network (SADDCS)," *1st Int. Conf. Ind. Electr. Electron. 2018)*, vol. 218, no. MATEC Web Conference, Anyer, Indonesia, pp. 1–8, 2018.

[2] S. Scott-Hayward, S. Natarajan, and S. Sezer, "A Survey of Security in Software Defined Networks," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 623–654, 2016.

[3] D. He, S. Chan, X. Ni, and M. Guizani, "Software-Defined-Networking-Enabled Traffic Anomaly Detection and Mitigation," *IEEE Internet Things J.*, vol. 4(6), pp. 1890–1898, 2017.

[4] X. Xia, Wen, Foh, Niyato, "Survey on software-defined networking," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes*

Bioinformatics), vol. 9106, no. 1, pp. 115–124, 2015.

[5] D. Kreutz, F. M. V Ramos, P. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-Defined Networking : A Comprehensive Survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[6] S. Scott-Hayward, S. Natarajan, and S. Sezer, "Survey of Security in Software Defined Networks," *Surv. Tutorials*, vol. 18, no. 1, pp. 623–654, 2015.

[7] O. Salman, I. Elhajj, A. Kayssi, and A. Chehab, "SDN Controllers: A Comparative Study," *2016 18th Mediterr. Electrotech. Conf.*, no. Cyprus, pp. 1–6, 2016.

[8] Cisco, "Cisco Cloud Index: Data Center SDN to Skyrocket by 2021.," 2018. [Online]. Available: https://www.sdxcentral.com/articles/news/cisco-cloud-index-data-center-sdn-skyrocket-2021/2018/02/. [Accessed: 24-Mar-2019].

[9] B. Görkemli, A. Parlakışık, and S. Civanlar, "Dynamic Management of Control Plane Performance in Software-Defined Networks," *2016 IEEE NetSoft Conf. Work.*, no. Seoul, Korea, pp. 68–72, 2016.

[10] M. ALAdaileh, M. Anbar, I. Hasbullah, C. Wey, and Y. Sanjalawe, "Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking Controller–A Review," *IEEE Access*, vol. 8.PP, pp. 143985–143995, 2020.

[11] S. K. Abdullah Gani, A. W. Ainuddin Wahid, A. Abdelaziz, K. Ko, M. K. Khan, and M. Guizani, "Software-Defined Network Forensics : Motivation , Potential Locations , Requirements , and Challenges," *IEEE Netw.*, vol. 30(6), pp. 6–13, 2016.

[12] D. Kreutz, F. M. V. Ramos, and P. Verissimo, "Towards Secure and Dependable Software-Defined Networks," *Proc. Second ACM SIGCOMM Work. Hot Top. Softw. Defin. Netw.*, pp. 55–60, 2013.

[13] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A Survey on Software-Defined Networking," *IEEE Commun. Surv. Tutorials*, vol. 17, no. 1, pp. 27–51, 2015.

[14] M. R. Haque *et al.*, "Motivation of DDoS Attack-Aware in Software Defined Networking Controller Placement," *2017 Int. Conf. Comput. Appl. ICCA 2017*, no. Doha, United Arab Emirates, pp. 36–42, 2017.

[15] S. Sindian, "An enhanced deep autoencoder-based approach for ddos attack detection," *WSEAS Trans. Syst. Control*, vol. 15, no. December, pp. 716–724, 2020.

[16] J. Chen, X. Zheng, and C. Rong, "Survey on software-defined networking," in *International Conference on Cloud Computing and Big Data in Asia*, 2015, pp. 115–124.

[17] S. Khan, A. Gani, A. W. Abdul Wahab, M. Guizani, and M. Khan, "Topology Discovery in Software Defined Networks : Threats , Taxonomy , and State-of-the-art," *IEEE Commun. Surv. Tutorials*, vol. 19(1), pp. 303–324, 2016.

[18] S. Shin, L. Xu, S. Hong, and G. Gu, "Enhancing Network Security Through Software Defined Networking (SDN)," *2016 25th Int. Conf. Comput. Commun. networks*, no. USA Waikoloa, pp. 1–9, 2016.

[19] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, "Software-Defined Networking Security: Pros and Cons," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 73–79, 2015.

[20] H. T. N. Tri and K. Kim, "Resource Attack Based on Flow Table Limitation in SDN," *Proc. Korea Inf. Process. Soc. Conf.*, no. Korea, pp. 215–217, 2014.

[21] S. Al-Mashhadi, M. Anbar, R. A. Jalal, and A. Al-Ani, "Design of Cloud Computing Load Balance System Based on SDN Technology," *Lect. Notes Electr. Eng.*, vol. 603, no. January, pp. 123–133, 2020.

[22] Y. Jarraya, T. Madi, and M. Debbabi, "A Survey and a Layered Taxonomy of Software-Defined Networking," *IEEE Commun. Surv. Tutorials*, vol. 16, no. 4, pp. 1955–1980, 2014.

[23] N. Z. Bawany, J. A. Shamsi, and K. Salah, "DDoS Attack Detection and Mitigation Using SDN: Methods, Practices, and Solutions," *Arab. J. Sci. Eng.*, vol. 42, no. 2, pp. 425–441, 2017.

[24] Akamai, "The State of the Internet," 2016.

[25] C. Bouras, A. Kollia, and A. Papazois, "SDN & NFV in 5G: Advancements and Challenges," *2017 20th Conf. Innov. Clouds, Internet Networks*, no. ICIN Conference, Paris, France, pp. 107–111, 2017.

[26] S. Scott-Hayward, G. O'Callaghan, and S. Sezer, "SDN Security: A Survey," *SDN4FNS 2013 - 2013 Work. Softw. Defin. Networks Futur. Networks Serv.*, 2013.

[27] H. D. Zubaydi, M. Anbar, and C. Wey, "Review on Detection Techniques Against DDoS Attacks on a Software-Defined Networking Controller," *2017 Palest. Int. Conf. Inf. Commun. Technol.*, pp. 10–16, 2017.

[28] 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, 2016.

[29] V. Patil, C. Patil, and R. N. Awale, "Security Challenges in Software Defined Network and their Solutions," *2017 8th Int. Conf. Comput. Commun. Netw. Technol.*, vol. 1, no. Delhi, India, pp. 1–5, 2017.

[30] 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, 2015.

[31] M. Azab and J. A. B. Fortes, "Towards Proactive SDN-Controller Attack and Failure Resilience," *Int. Conf. Comput. Netw. Commun.*, vol. 26, no. ICNC, Silicon Valley, USA, 2017, p. (pp. 442-448), 2017.

[32] Y. Cui *et al.*, "SD-Anti-DDoS: Fast and Efficient DDoS Defense in Software-Defined Networks," *J. Netw. Comput. Appl.*, vol. 68, pp. 65–79, 2016.

[33] M. Yang, Y. Li, D. Jin, L. Zeng, X. Wu, and A. V. Vasilakos, "Software-Defined and Virtualized Future Mobile and Wireless Networks: A Survey," *Mob. Networks Appl.*, vol. 20, no. 1, pp. 4–18, 2015.

[34] I. Alsmadi and D. Xu, "Security of Software Defined Networks: A Survey," *Comput. Secur.*, vol. 53, pp. 79–108, 2015.

[35] K. Kalkan, G. Gur, and F. Alagoz, "Defense Mechanisms Against DDoS Attacks in SDN

Environment," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 175–179, 2017.

[36] 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.

[37] 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. 1736–1739, 2016.

[38] Y. Hande and A. Muddana, "A survey on intrusion detection system for software defined networks (SDN)," *Int. J. Bus. Data Commun. Netw.*, vol. 16, no. 1, pp. 28–47, 2020.

[39] T. Jafarian, M. Masdari, A. Ghaffari, and K. Majidzadeh, "A survey and classification of the security anomaly detection mechanisms in software defined networks," *Cluster Comput.*, vol. 1, 2020.

[40] 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. Networks*, 2018.

[41] 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," *2016 IEEE Int. Conf. Commun.*, no. ICC, Kuala Lumpur, Malaysia, pp. 1–6, 2016.

[42] R. R. Coifman and M. V. Wickerhauser, "Entropy-Based Algorithms for Best Basis Selection," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 713–718, 1992.

[43] A. X. Liu, "An Advanced Entropy-Based DDoS Detection Scheme," *2010 Int. Conf. Information, Netw. Autom.*, vol. 2, no. China, Kunming, pp. V2-67-V2-71, 2010.

[44] L. Li, J. Zhou, and N. Xiao, "DDoS Attack Detection Algorithms Based on Entropy Computing," *Int. Conf. Inf. Commun. Secur.*, no. Springer, Berlin, Heidelberg, pp. 452–466, 2007.

[45] S. Khan, A. Gani, A. W. Abdul Wahab, and P. K. Singh, "Feature Selection of Denial-of-Service Attacks Using Entropy and Granular Computing," *Arab. J. Sci. Eng.*, vol. 43(2), pp. 499–508, 2017.

[46] M. Kia, "Early Detection and Mitigation of DDoS Attacks in Software Defined Networks," *Master's Thesis, Ryerson Univ. Toronto, ON, Canada,2015*, 2015.

[47] T. H. Lee and J. De He, "Entropy-Based Profiling of Network Traffic for Detection of Security Attack," *Inst. Electr. Electron. Eng. Reg. Asia Pacific, Inst. Electr. Electron. Eng.*, no. Singapore, pp. 1–5, 2009.

[48] R. Wang, Z. Jia, and L. Ju, "An Entropy-Based Distributed DDoS Detection Mechanism in Software-Defined Networking," *2015 IEEE Trust.*, vol. 1, no. Finland Helsinki, pp. 310–317, 2015.

[49] S. M. Mousavi and M. St-Hilaire, "Early Detection of DDoS Attacks Against SDN Controllers," *2015 Int. Conf. Comput. Netw. Commun.*, no. Anaheim, California, USA, pp. 77–81, 2015.

[50] S. Oshima, T. Nakashima, and T. Sueyoshi, "DDoS Detection Technique Using Statistical Analysis to Generate Quick Response Time," *2010 Int. Conf. Broadband, Wirel. Comput. Commun. Appl.*, no. Fukuoka, Fukuoka Prefecture Japan, pp. 672–677, 2010.

[51] S. M. Mousavi, "Early Detection of DDoS Attacks in Software Defined Networks Controller," *Master Diss. Carlet. Univ. Ottawa, 2014*, 2014.

[52] Y. Jiang, X. Zhang, Q. Zhou, and Z. Cheng, "An Entropy-Based DDoS Defense Mechanism in Software Defined Networks," *Int. Conf. Commun. Netw. China*, vol. 1, no. Springer, Cham, pp. 169–178, 2016.

[53] X. Huang, X. Du, and B. Song, "An Effective DDoS Defense Scheme for SDN," *2017 IEEE Int. Conf. Commun.*, no. ICC Conference, Paris, France, pp. 1–6, 2017.

[54] D. Hu, P. Hong, and Y. Chen, "FADM: DDoS Flooding Attack Detection and Mitigation System in Software-Defined Networking," *GLOBECOM 2017-2017 IEEE Glob. Commun. Conf.*, no. Singapore, pp. 1–7, 2017.

[55] 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. Serv. Manag.*, vol. 15(4), pp. 1545–1559, 2018.

[56] S. Mousavi and M. St-Hilaire, "Early Detection of DDoS Attacks Against Software Defined Network Controllers," *J. Netw. Syst. Manag.*, vol. 26(3), pp. 573–591, 2018.

[57] J. Boite, P. A. Nardin, F. Rebecchi, M. Bouet, and V. Conan, "Statesec: Stateful Monitoring for DDoS Protection in Software Defined Networks," *In2017 IEEE Conf. Netw. Softwarization*, no. EEE, 2017, Bologna, Italy, p. (pp. 1-9), 2017.

[58] G. Yonghao and W. Weiming, "DDoS Detection and Prevention Based on Joint Entropy and Conditional Entropy," *Key Eng. Mater.*, vol. 474, pp. 2129–2133, 2011.

[59] J. Mao, W. Deng, and F. Shen, "DDoS Flooding Attack Detection Based on Joint-Entropy with Multiple Traffic Features," *IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. IEEE Int. Conf. Big Data Sci. Eng.*, vol. New York, pp. 237–243, 2018.

[60] H. Rahmani, N. Sahli, and F. Kammoun, "Joint Entropy Analysis Model for DDoS Attack Detection," *2009 Fifth Int. Conf. Inf. Assur. Secur.*, vol. 2, no. Xi'An, China, pp. 267–271, 2009.

[61] K. Kalkan, L. Altay, G. Gur, and F. Alagoz, "JESS: Joint Entropy Based DDoS Defense Scheme in SDN," *IEEE J. Sel. Areas Commun.*, vol. 36(10), pp. 2358–2372, 2018.

[62] P. Bereziński, B. Jasiul, and M. Szpyrka, "An Entropy-Based Network Anomaly Detection Method," *Entropy*, vol. 17, no. 4, pp. 2367–2408, 2015.

[63] M. Masi, "A Step Beyond Tsallis and Rényi Entropies," *Phys. Lett. Sect. A Gen. At. Solid State Phys.*, vol. 338, no. 3–5, pp. 217–224, 2005.

[64] Y. Xiang, K. Li, and W. Zhou, "Low-Rate DDoS Attacks Detection and Traceback by Using New Information Metrics," *IEEE Trans. Inf. Forensics Secur.*, vol. 6, no. 2, pp. 426–437, 2011.

[65] R. Yan, "Combining Renyi Entropy and EWMA to

Detect Common Attacks in Network," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 30, no. 10, pp. 1–23, 2016.

[66] R. Yan, G. Xu, and X. J. Qin, "Detect and Identify DDoS Attacks from Flash Crowd Based on Self-Similarity and Renyi Entropy," *2017 Chinese Autom. Congr.*, pp. 7188–7194, 2017.

[67] W. Wang, X. Ke, and L. Wang, "A HMM-R Approach to Detect L-DDoS Attack Adaptively on SDN Controller," *Futur. Internet*, vol. 10, no. 9, p. 83, 2018.

[68] K. S. Sahoo, D. Puthal, M. Tiwary, J. Rodrigues, B. Sahoo, and R. Dash, "An Early Detection of Low Rate DDoS Attack to SDN Based Data Center Networks Using Information Distance Metrics," *Futur. Gener. Comput. Syst.*, vol. 89, pp. 685–697, 2018.

[69] P. Zhai, Y. Song, X. Zhu, L. Cao, J. Zhang, and C. Yang, "Distributed Denial of Service Defense in Software Defined Network Using OpenFlow," *2020 IEEE/CIC Int. Conf. Commun. China, ICCC 2020*, no. Iccc, pp. 1274–1279, 2020.

[70] G. A. Ajaeiya, N. Adalian, I. H. Elhajj, A. Kayssi, and A. Chehab, "Flow-Based Intrusion Detection System for SDN," *Proc. 2017 IEEE Symp. Comput. Commun. (ISCC), Heraklion, Greece*, vol. 3–6, pp. 787–793, 2017.

[71] J. Cui, J. He, Y. Xu, and H. Zhong, "TDDAD: Time-Based Detection and Defense Scheme Against DDoS Attack on SDN Controller," *Australas. Conf. Inf. Secur. Priv.*, vol. 10946, no. ACISP Conference, Wollongong, Australia, pp. 649–665, 2018.