Flow Table Overflow Attacks in Software Defined Networks: A Survey

Changqing Zhao[1], Ling Xia Liao[1], Han-Chieh Chao[2], Roy Xiaorong Lai[3], Miao Zhang[4]*

[1] School of Electronic Information and Automation, Guilin University of Aerospace Technology, China
[2] Department of Electric Engineering, National Dong Hwa University, Taiwan
[3] Confederal Networks Inc., USA
[4] Quanzhou University of Information Engineering, China
zhaochq@guat.edu.cn, liaolx@guat.edu.cn, hcc@gms.ndhu.edu.tw, roy.lai@ieee.org, zm@qzuie.edu.cn

Abstract

While Software-Defined Networks (SDNs) have separated control and data planes and completely decouple the flow control from the data forwarding to enable network flexibility, programmability, and innovation, they also raise serious security concerns in each plane and the interfaces between the two planes. This paper, instead of studying the security issues in the SDN control plane as many literatures have done in current research, focuses on the security issues in the SDN data plane, aiming at the state of the art mechanims to identify, detect, and mitigate them. Specifically, this paper reviews the typical models, detections, and mitigations of SDN flow table overflow attacks. After reviewing the various vulnerabilities in SDNs, this paper categorizes the flow table overflow attacks into saturation, low-rate table exhaustion, and slow saturation attacks, and summarizes the attack models, detections, and mitigations of each category. It reviews the typical attacks that can overflow the flow tables and provides the main challenges and open issues for the future research.

Keywords: SDN, Saturation attack, Low-rate table exhaustion attack, Slow saturation attack

1 Introduction

By decoupling network control from data forwarding to form layered architecture, Software-Defined Networks (SDNs) enable flexible network configuration, better network performance, and centralized monitoring and automation [1]. SDNs can meet the stringent requirements of low network latency, Quality of Servise (QoS), and Service Level Agreement (SLA), and have been adopted in 5G moble [2], Internet of Things (IoT) [3], and Industrial IoT systems [4].

SDNs typically have a layered architecture including the application layer, the control layer, and the infrastructure layer, as shown in Figure 1. The application and control layers form the control plane and the infrastructure layer is the data plane. The application layer includes applications such as switching, routing, and load balancing, the control layer has controllers, and the infrastructure layer consists of packet forwarders (also called switches) and hosts. Between the application and control layers, SDNs introduce a northbound Application Program Interface (API) to allow the applications to collect network status and enforce management policies. Between the control and infrastructure layers, SDNs standardize on a southbound interface such as OpenFlow [5], to program the behavior of flows and switches and to maintain a global network view of the network. Specifically, switches in the infrastructure layer do not have "brain" and rely on the forwarding rules (also called the flow entries in SDNs) to determine the forwarding of packets. Flow entries are flow-based. Controllers generate them and install them in the flow tables located in the Ternary Content Addressable Memory (TCAM) of the switches. TCAM supports efficient matching of flows to flow entries for the fine-grained packet forwarding.

Although SDNs provide new primitives and facilitate network programming and application innovation, SDNs raise the security concerns on the layered architecture that can be exploited to disrupt the normal operation of SDNs [6].

The major vulnerabilities of SDNs can be traced back to the flow setup procedure, which sets up flow entries for new flows, and the limited resources of the participants in this procedure [7]. Figure 1 shows the main participants in this procedure: 1) the switches that receive the flow packets; 2) the flow entries that are looked up to determine the behavior of switches corresponding to the received packets; 3) the communication channel through which the switches forward the received packets to the controller if no matching flow entries are found in the flow table; 4) the controller that receives the forwarded packets, generates the matching flow entries, and installs them on the corresponding switch via the communication channel; and 5) the flow table that is checked to determine if it has enough space for the received entries. The entries are stored in the flow table and the received packets are forwarded out if there is enough space, and dropped otherwise. SDNs rely on this process to operate the network normally and provide advanced features.

All the entities involved in the flow setup procedure can be the targets of cyber attacks. For instance, the applications in the application layer are often provided by the third party. Attackers can take advantage of this and insert malicious applications and packets to overload the controllers, switches, and the communication channel between them. The logically centralized controller in the control layer is responsible for determining the behavior of all switches and flow packets in the network by setting the configuration of switches and

*Corresponding Author: Miao Zhang; E-mail: zm@qzuie.edu.cn
DOI: 10.53106/160792642023122407001

generating flow entries for switches, creating a notable single point of failure. The communication channel between the controllers and switches has the limited bandwidth but should move control packets in a reliable and efficient manner, making it the major target of cyber attacks [8].

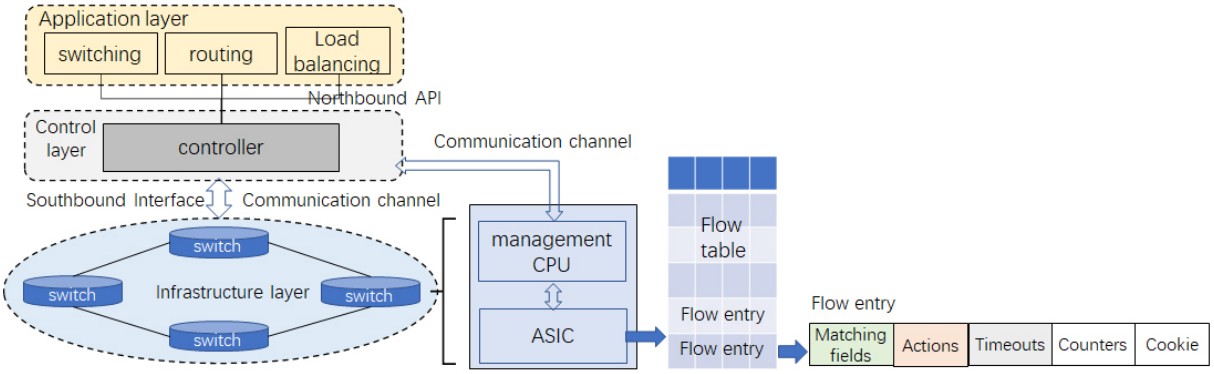

Figure 1. The architecture of software-defined networks

Regarding the vulnerabilities in the infrastructure layer and the communication channel between the control and infrastructure layers, the limited bandwidth of the communication channel may be congested when in-bind control plane is enabled. Although the bandwidth problem can be significantly mitigated by using the out-of-bind control plane, additional money must be invested to build the dedicated control infrastructure to move the control traffic [5].

The second vulnerability is the limited computing and storage resources of SDN switches. The local control Central Processing Unit (CPU) in an SDN switch is weak, so the communication channel between the local control CPU and the Application Specific Integrated Circuit (ASIC) in a switch is shallow and can be easily congested by malicious flow packets [7]. More importantly, SDN switches incorporate TCAM to store flow entries and enable efficient and fine-grained flow matching. Since the size of TCAM is limited due to its cost, footprint, and power consumption, while the number of forwarding entries required by an SDN switch is much larger than that required by a traditional switch, and each SDN flow entry typically consumes much more memory than the forwarding rules in the traditional networks, the flow table in SDN switches typically suffers from space shortage and can be easily overflowed by normal burst flows or attack flows [7].

Current research has made great efforts to study the mechanisms attacking SDNs, and many literatures have studied the attack models, detection and mitigation mechanisms for the control plane [8-9]. For instance, the weakness of the application and control layers, the various attack mechanisms, such as faking or spoofing traffic flow, deploying untrusted applications, overloading the controllers [10]. However, the strudy of security vulnerabilities in the infrastructure layer is still in its infancy [11-12].

This paper focuses on data plane security issues. In particular, it reviews the state of the art in flow table overflow attacks. It first categorizes the typical flow table overflow attacks into saturation, low-rate table exhaustion, and slow saturation attacks, and then summarizes the attack models and proposals for each attack type. It introduces the state of the art of mechanisms to detect and mitigate such attacks.

Although some researchers have made comprehensive suverys on saturation attacks on SDNs [13] and the LDoS attacks on switches [14-15], to the best of our knowledge, this work is the first effort on reviewing the typical attack models, detections, and mitigations for the space shortage of flow tables. The major contributions of this paper are three folds.

1. The typical attacks overflowing flow tables are reviewed.
2. The state of the art of mechanisms in detecting and mitigating flow table overflow attacks is summarized.
3. The major challenges and open issues in the area of secure flow tables are presented.

The rest of this paper is organized as follows. While the related surveys are summarized in Section 2, the major vulnerabilities in the SDN data plane are introduced in Section 3. Section 4 surveys the flow table overflow attacks, and Sections 5 and 6 summarize the state of the art in detection and mitigation of attacks, respectively. Section 7 proposes the challenges and open issues related to the attacks followed by the conclusion in Section 8.

2 Related Surveys

SDN security has been a hot research area since the introduction of SDN architecure, various security drawbacks and challenges in the architecture have been identified and studied [6, 10], and the potential solutions have been discussed and surveyed [16]. As listed in Table 1, the security issues in different networking scenarios such as SDN-based vanet [17], SDN-based IoT [18], and wireless networks [19] have been reviewed. While the cyber attacks for SDNs have been extensively studied [6], the security of SDN control plane and data plane have been discussed by [9] and [11], respectively. Although the traditional saturation attacks and low-rate table exhaustion attacks have been comprehensively surveyed by [13] and [14], respectively, this paper targets the security issues in the space shortage of SDN flow tables.

Table 1. The summary of related surveys

Reference	Year published	Purposes
[6]	2015	SDN security drawbacks, challenges, and cyber attacks
[9]	2019	Cyber attacks for the SDN control plane
[10]	2020	SDN security drawbac and challenges
[11]	2017	Cyber attacks for the SDN data plane
[13]	2015	Traditional saturation attacks
[14]	2022	Traditional low-rate exhaustion attacks
[16]	2020	Solutions for SDN securty concerns
[17]	2020	Security issues in SDN-based Vanet
[18]	2018	Security issues in SDN-based IoT
[19]	2019	Security issues in wireless networks
This survey		SDN table overflow attacks including saturaton, low-rate exaustion, and slow saturation.

3 Volnerabilities in SDN Data Plane

The SDN architecure is designed to simplify the control and management of large-scale networks. As shown in Figure 1, the architecture typically consists of 3 layers: the infrastructure layer supporting the data plane operation, the control layer consisting of controllers, and the application layer containing various network applications. Two interfaces between the layers are also provided: the northbound API and the southbound interface. The former is exposed to SDN application developers to hide the network complexity, and the latter, such as the OpenFlow protocol, abstracts the entire network for network programming [5].

The forwarding rules in SDNs are called flow entries. They are stored in the flow tables of SDN switches. Each flow entry consists of 5 main parts: 1) the matching rule that specifies a flow by matching the combination of selected header fields from layer 2 to layer 4; 2) the actions that the matched flows should take; 3) the counters that collect the statistics of the matched flows; 4) the timeouts that define the lifetime of the entry in the flow table; and 5) the cookies that allow interaction between the switch and its controller. TCAM is used to store flow tables to support wildcard matching, providing fast, flexible, and fine-grained flow management.

3.1 Vulnerabilities in Application and Control layers

The logically centralized controller in the control plane is one of the major vulnerabilities of SDNs. The compromised controller can program the network and manipulate the resources by inserting fake flow rules or leaking the key network parameters.

The authorization attacks can lead to the cyber attacks on the controller and applications. Current SDNs do not have a strong mechanism for the authorization and authentication. The accountability usually relies on the third party to consider the consumption of network resources.

The separation of the control and application layers and the interface between the layers exploit the vulnerability of congestion. As listed in Table 2, the applications, the northbound API, and the controllers become targets for various cyber attacks such as saturation and Man-in-the-Middle (MiM) attacks due to the limited capabilities.

Table 2. Volnerabilities in SDNs

Reference	Category	Features
[20]	Southbound interface	lack of certificates in the handshake phase of the authentication process can lead to MiM attacks on the communication channel between controllers and switches
[21]	Southbound interface	lack of TLS configuration can lead to gain the access to the forwarding information and rules
[20, 22-23]	Southbound interface	Extending current southbound interface protocols or developing new protocols to mitigate the security issues in southbound interfaces.
[7]	Southbound interface	Resource shortage in flow table space, the local Management CPU, and the communication channel bandwidth between control and data planes and the bandwidth between the management CPU and the ASIC inside switches
[6]	Control plane	Centralized controller is the network single point of failure, compromized controller can leak network parameters. Controllers and applications suffer from cyber attacks
[14]	Data plane	Flow table setup procedure and the resource shortage in switches

3.2 Vulnerabilities in Communication Channel between Control and Infrastructure Layers

The southbound interface such as the OpenFlow protocol has no technical security issues. Optionally, secure communication links can be established between switches and their controllers using secure protocols such as Transport Layer Security (TLS). However, the lack of certificates in the handshake phase of the authentication process can lead to MiM attacks on the communication channel between controllers and switches [20]. Also, the attacker can take advantage of the lack of TLS configuration to gain the access to the forwarding information and rules [21]. In this sense, to mitigate the MiM attacks in the southbound interface, the extensions to the TLS protocol [20] or new protocols [22-23] can be involved to mitigate such security issues.

The communication channel between controllers and switches has finite bandwidth and computational capacity, which limits the rate at which flows can be set up. The best implementations we know of in the current literature can only set up a few hundred flows per second, which is insufficient for flow setup in a high-performance network. Inside an SDN switch, as shown in Figure 1, the local management CPU and the communication channel between the local CPU and the ASIC allow a slow path for flow forwarding, while the path inside ASIC is very fast, exposing the entire communication channel and controllers to saturation attacks [7].

3.3 Vulnerabilities in Infrastructure Layer

Because SDNs need to enable wildcard flow matching, flow entries are stored in the TCAM. Since TCAM is expensive and power hungry, an SDN switch has a limited TCAM resource and supports a very limited number of flow entries. However, the number of flow entries required by SDNs is much larger than that required by traditional networks due to the support of fine-grained network forwarding and management, in addition, each SDN flow entry costs more bits, which leads to a huge flow table space shortage and exposes flow tables to the overflow attacks in practice.

4 Flow Table Overflow Attacks

4.1 Attack Models

To enable fine-grained and flexible network management, SDNs have a flow setup procedure that allows controllers to reactively set up flow entries for the new flows in the network. In this procedure, each new flow in the network must trigger the controller to create the appropriate flow entry. If the flow table runs out of space, the matched flow entries newly created by the controller won't have room to fit, and the corresponding flow packets will have to be dropped, causing a significant packet loss, increasing the network forwarding delay, and degrading the network QoS and SLA.

Assume that flow entries are generated and installed reactively by controllers. Suppose an attacker controls some of the hosts that connected to the target switch. The attacker can use saturation, low-rate table exhaustion, and slow saturation attacks without any prior network configuration to overflow the flow table of the targe switch.

4.1.1 Saturation Attacks

A saturation attack simply floods the flow table of the target switch by rapidly sending a large number of packets in a very short period of time without knowing and inferring any network settings, as shown in Figure 2(a). Saturation attacks are also called Denial-of-Service (DoS) attacks because they deny the normal network services [24]. Saturation attacks significantly increase the network packet rate, which is the most important network metric in the detection mechanisms. Saturation attacks can target controllers, switches, and the communication channel between controllers and switches. This paper discusses the saturation attacks that overflow the flow tables of switches.

4.1.2 Low-rate Table Exhaustion Attacks

A low-rate table exhaustion attack must first infer the flow entry timeout using the existing SDN timeout probing approaches, and then craft its packets accordingly using the gathered information to convertly overflow the flow table of the target switches.

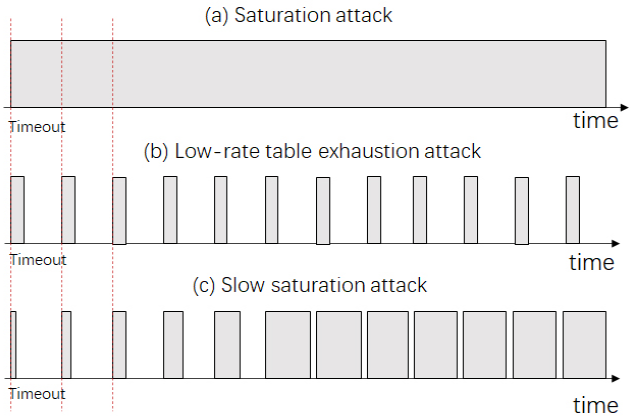

Figure 2. Flow table overflow attack models

Low-rate table exhaustion attacks typically consist of two phases: 1) the probe phase and 2) the launch phase. The former allows the attacker to use the existing time probing approachers to infer and identify the target network settings, such as the flow entry timeout (To) and the flow table capacity of the target switch (Nr). With To and Nr, the attacker is able to set the attack packet generation rate (Nps) and attack packet retransmission rate (Npr) so that they not only have minimal impact on the overall packet rate of the network, but also continuously install flow entries for the attack flows in the target flow table to keep the flow table full. After setting the parameters for the attack, the attacker executes the attack by crafting and sending the packets accordingly [15, 26].

In a low-rate table exhaustion attack, the controller and the switch operate normally, but they are forced to serve only the flow entries installed before the attack and the flow entries installed by the attacker, thus denying the service to new legitimate flow packets. The attack flows can mimic the normal short flows or IoT data flows in the networks, and their packet rate is too low to trigger traditional saturation detection approaches. Therefore, low-rate table exhaustion attacks are also called Low-rate DoS (LDoS) attacks to

distinguish them from DoS in their packet rates [14].

4.1.3 Slow Saturation Attacks

Slow saturation attacks are performed by combining a low-rate table exhaustion attack and a low packet rate saturation attack [15]. Slow saturation attacks inject attack flows into the network at a much lower packet rate than that the saturation attacks, but can disrupt the target switch by dropping the flow entries of incoming legitimate flows, resulting in the denial of service to those flows [27].

A slow saturation attack also consists of two phases: the probing phase and the launching phase. In the probing phase, the attacker must discover the flow entry timeout using the existing tools and techniques. The launch phase typically involves the following steps: 1) controlling a sufficient number of bots, which is typically greater than the half rule capacity of the target switch (Nr/2), 2) each bot sends an attack packet to the target switch without using spoofed IP addresses. Whenever the switch receives the first packet of a new flow, two new flow entries for the incoming and the outgoing flows, are eventually installed, 3) the attack packet generation rate (Nps) is controlled so that an attacker can overflow the target switch's flow table quickly enough while keeping the rate at which new flow entries are installed not too high, 4) each bot keeps re-sending attack packets to the target switch within its flow entry idle timeout (To) at the packet rate of Npr. After the flow table of the target switch is full of bot entries and the removed flow entries are always installed by the new bot entries, the target flow table will remain full, 5) Finally, by coordinating the Nps, Npr, and To, the attacker can keep the flow table of the target switch in the full state indefinitely and deny service to the legitimate flow packets [26].

4.1.4 Summary

Saturation attacks use a high traffic rate to flood the flow table of the target switch without knowing the network settings. Low-rate table exhaustion attacks and slow saturation attacks flood the target switch's flow table with a low traffic rate by coordinating attack flows with the network settings. While low-rate table exhaustion attacks keep the attack traffic rate below 0.2 times the normal traffic rate, slow saturation attacks slowly increase the attack traffic rate to survive saturation detection mechanisms.

4.2 Flow Table Overflow Attack Proposals

4.2.1 Sniffing Network Settings

Network settings must be discovered in the low-rate table exhaustion attacks and the slow saturation attacks. Attackers must sniff SDN settings using existing SDN timeout probing approaches.

- **Hearder fields scan.** SDN SCANNER [28] scans header fields to determine which header fields are used to match flow entries. Particulary, SDN SCANNER sents two (or more) specially crafted packets to a target network and first records the response time of each packet. Let T1 be the response time for the first packet and T2 be the time for the second packet. SDN scanner repeated this process by changing one field of the packet header. Finally, SDN scanner collected T1 and T2 for each different header field. References [29-30], and [31] also proposed the similar methods that used the information from the Round Trip Time (RTT) and packet-pair dispersion of the exchanged packets and launched fingerprint attacks on SDN networks to succeed with overwhelming probability.
- **Timeouts inference.** T-test [32] is a statistical test tool to infer the imeouts of flow entry. After the samples of T1 and T2 are collected by the SDN scanner, t-test can involve to tell whether two sample sets are significantly different from each other or not with a high confidence via the time difference of T2-T1.
- **Flow table capacity and usage discovery**. Reference [30] introduced a model for inferring the flow table capacity and usage. As similar as the SDN SCANNER, it first discovered the header fields for flow entry matching, then probed packets in the network to infer the state of flow table and flow entry. Third, the inferred states were used as control signals to compute the flow table capacity and usage. Reference [31] proposed a method to predict the flow table size. Firstly, it constructed the attack to occupy n flow rules, and then inferred whether the flow tables are full by sending some packets to measure the RTT differences. The flow table is full if the measured RTTs of these packets are significantly deviated. Otherwise, another round of probing should be initiated to occupy another n flow rules.

4.2.2 Attack Flows

Saturation attacks do not actually construct flows but must send the attack packets at a high enough rate to flood the flow table, although the saturation attacks that overload the communication channel between controllers and switches may need craft attack flows [24]. Saturation attack packets often have no real payload with a real or faked IP address to conserve attackers' resources [24].

Low-rate flow table exhaustion attacks must craft flows according to the sniffed network settings. The attack flows must ensure that each attack packet can effectively trigger a unique flow entry installation in flow tables. Since the match fields have been sniffed along with the wildcard in the probing phase, the header fields of each attack packet can be easily modified, and the minimum number of packets to overflow flow tables of a switch can be achieved. Attack packets can maintain the minimum size as they do not need to include any payload [31].

Two types of flows are typically used, the long attack flows and the short attack flows, as shown in Figure 3(a) and Figure 3(b). The packet inter-arrival time of such attack flows should be slightly shorter than the flow entry timeout. The idea behind this is to ensure that each long flow continuously occupies the space of a flow entry, or to ensure that the space of a timed out flow entry can be occupied by the entry of a short attack flow [33].

Slow saturation attacks must ensure that the attack packet rate is increased slowly to avoid triggering the saturation detection mechanisms. Therefore, the slow saturation attack packets should take the flow entry timeout as the period and slowly increase the number of attack packets in each period, as shown in Figure 3(c). The idea behind this is to preserve

the flow entry space taken in the previous periods, while slowly consuming more flow table space.

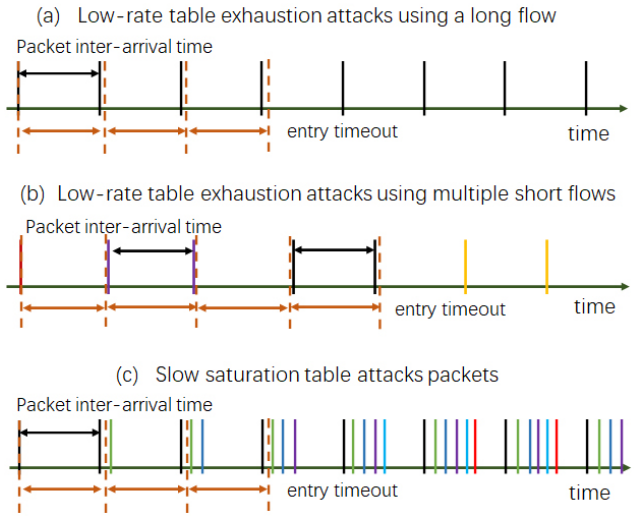

Figure 3. Attack flows

4.2.3 Attack Proposals

Saturation attacks are the typical attacks to overload SDN resources such as controllers, control channel, and switches. As listed in Table 3, while reference [34] studied the saturation attacks for controllers, reference [24] simulated the saturation attacks for controllers and control channel. Although reference [25] proposed the saturation attacks targeting both the control channel and the SDN flow table, references [35-36], and [37] proposed the brute-force and high-rate saturation attacks on SDN flow tables. Reference [38] also introduced an advanced saturation attack targeting flow tables consisting of flow entries with dynamic timeouts. All of these attacks generated a large number of crafted or random packets per second. Saturation attacks signficantly increase network packet rate.

Low-rate flow table exhaustion attacks overflow the flow table of the target switch in two phases. Although reference [28] proposed SDN SCANNER to fingerprint the network settings in the probing phase, it did not provided details to launch the attacks. Reference [29] investigated the ability of a fingerpring attack to identify the interaction between the controller and the switches triggered by an attack packet. The experimental settings such as the testbed, networking topology, and traffic load are provided, and some network metrics such as packet-parity dispersion and RTT are evaluated. Reference [30] developed the FIFO and LRU inference algorithms to predict the flow table size and usage based on the measured RTT data. Unlike the attacks described above, which typically focus on discovering the network settings, LOFT proposed by reference [31] introduced the complete process of a low-rate flow table exhaustion attack. LOFT could accurately sniff the network settings of flow entries using only a small number of probing packets. By measuring the RTTs of the probing packets, an attacker can accurately infer the flow entry settings such as the match fields and the wildcards that indicate which packets will trigger the installation of new flow entries, and the timeouts that define the lifetime of the entries in the flow table. In the attack phase, LOFT generates low-rate attack traffic to overflow flow tables according to the inferred flow entry settings. It constructs different attack packets using some specific match fields so that each such packet can trigger a new flow rule installation. Meanwhile, based on the timeout configurations, it can calculate the minimum packet rate to keep the flow tables overflowed over time.

Table 3. Flow table attacks, detections, and mitigtions

Reference	Category		Features
[24, 34]	Attacks	Saturation	Saturation attacks for controllers
[25, 35-38]			Saturation attacks for data plane
[24-25]			Saturation attacks for control channel
[28-30]		Low-rate table exhaustion	Fingerprint network setting
[31]			Complete attack process
[15]		Slow saturation	An attack consisting of a countinuous slow saturation attack and a burst slow saturation attack
[43-47]	Detections	Saturation	Use machine learning approaches to detection saturaton attacks
[33]		Low-rate table exhaustion	Strategies of table overflow prediction and flow entry deletion
[48]			Use the number of proactive flow entries to filter malicious flows
[49]			Use the features of flow entries and factorization machine to detect
[50]			Use of traffic features and machine learning approaches to detect
[42]	Mitigations		Limit the flow entry setup speed
[51]			Redirect flow entries to other switches
[52]			Sharing flow table space among switches
[26]			Randomly delete flow entries
[15]			Moving target defense

Slow saturation attacks are the combination of saturation attacks and low-rate flow table exhaustion attacks. Reference [15] introduced a slow saturation attack consisting of a countinuous slow saturation attack and a burst slow saturation attack. In the continuous slow saturation attack, both the low-rate flow table exhaustion attack and the saturation attack were performed simultaneously; in the burst slow saturation attack, the low rate flow table exhaustion attack was performed during the whole duration of the attack, but the saturation attack alternated between two bursts in which both the low-rate flow table exhaustion and the saturation traffics were sent; and sleeping periods in which the attack sent only low-rate flow table exhaustion attack traffic. The experiments showed that the attacker has a number of options when performing a slow saturation attack, such as selecting the type of saturation attack (continuous or burst), the intensity, and the intervals between bursts. Many currently proposed LDoS flow table overflow attacks should be categorized as low-rate flow table exhaustion attacks rather than slow saturation attacks.

5 Table Overflow Attack Detection

5.1 Saturation Attack Detection

Existing approaches to flow table overflow often consider saturation attacks. They typically assume that the attack packets are sent at a very high rate combining IP spoofing and flooding techniques. The main approaches to avoid these attacks are.

1) Optimizing flow entry timeouts to remove obsolete entries [28, 39];
2) Monitoring unpaired flow entries to detect IP spoofing [15, 40];
3) Monitoring the installation rate of flow entry [36]. When the rate of new entries is high, the countermeasures are triggered.
4) Monitoring the CPU and memory of SDN controllers and the buffer of switches [41]. When they start to consume too more computational resources, the network may be under saturation attacks as they deal with the increased traffic [15]. The growth of foreign flow consumption, the deviation of amount, and the commonness of flow entry can indicate the saturation attacks as such attacks sent a large number of flows in a short time [42].
5) Use of machine learning and statistical analysis techniques. Reference [43] proposed the use of neural networks, entropy, and more sophisticated traffic analysis methods to help the controller to determine whether the network was under saturation attack or not. Reference [44] proposed three supervised classifiers and four semi-supervised classifiers for five types of saturation attacks (TCP-SYN, UDP, ICMP, IP-Spoofing, and TCP-SARFU) and their combinations, although not all of these attacks were used to overflow flow tables. Other machine learning based approaches to saturation attack detection are discussed in references [45-46], and [47].

5.2 Low-rate Table Exhaustion Attack Detection

Because low-rate table exhaustion attacks can set a low rate of flow entry creation, the defenses against saturation attacks are not effective in detecting low-rate table exhaustion attacks. Cao et al. [31] gave two simple methods to prevent the network configuration sniffing, but no effective scheme for low-rate table exhaustion detection and mitigation. Xie et al. [33] proposed the SAIA by introducing the strategies of table overflow prediction and flow entry deletion. Kong et al. [48] proposed to use the number of proactive flow rules in the flow table as a detection metric, and applied a statistical approach to help filter malicious flows, because proactive flows from the attacked port always occupy a stable proportion in the flow table regardless of the attack form. Therefore, Kong's approach could be used to detect any types of flow table overflow attacks. Wu et al. [49] extracted several features from the flow rules, and proposed a LDoS attack detection method based on factorization machine over such features. The experiments show that the method could effectively detect the LDoS attack with an accuracy reaching 95.80%. Reference [50] extracted the essential traffic features such as the distribution of packet interval, the number of packets, the duration of the flow and the relative distribution of match bytes, and proposed SVM, C4.5 and Naïve Bayes based models to detect the LDoS attacks. The approaches proposed in references [49] and [50] could detect many types of low-rate DoS attacks including the low-rate flow table overflow exhaustion attacks.

5.3 Slow Saturation Attack Detection

Current research has not proposed detection approaches specifically for slow saturation attacks. However, since the slow saturation attacks are the combination of saturation attacks and low-rate table exhaustion attacks, the detection approaches for the slow saturation attacks should consider the characteristics of the saturation and the low-rate table exhaustion attacks.

5.4 Summary

Detecting flow table overflow attacks often rely on the features of traffic, flow entries, and switches to model attacks based on machine learning algorithms. Saturation attacks is easy to detect due to its high traffic rate, but low-rate table exhaustion and slow saturation attacks are not because of their low traffic rate.

6 Table Overflow Attack Mitigation

Mitigation mechanisms are often used to reduce the overhead of the flow table after the attacks are detected. The mitigation mechanisms reduce the overflow of the flow table regardless of the type of table overflow attacks.

- Reference [42] limited the speed of installing flow entries from the controller using token bucket, so as to avoid exhausting the flow table space by saturation attacks.
- Reference [51] remodeled the flow table to maximize the use of the table, and proposed redirection mechanisms to detour packets to other switches

to balance the size of the flow table. Similarly, reference [52] allowed switches to share their unused TCAM memory space with other switches, so that the attacked switches could redirect their flows to other peer switches according to parameters such as TCAM usage, proximity to the attacked switch, how busy a switch is, and how a switch is connected to other switches.

- Reference [26] randomly selected rules to be dropped when the system is overloaded to mitigate the low-rate table exhaustion attack.
- Moving Target Defense (MTD) are proactive defences that randomly change network settings, parameters, or topology. They dynamically change the attack surface and impose advantages on mitigating the slow saturation attack [15]. Reference [53] reviewed the general MTD approaches. In SDNs, MTD approaches could randomize the timeout values for some flow rules [54-55], dynamically change the IP and MAC addresses of controllers [56], and maintain a pool of healthy controllers to be replaced with faulty ones [57], to increase the difficulty of probing network settings and thus launching flow table overflow attacks. MTD approaches can defend against all types of flow table overflow attacks.

7 Challenges and Open Issues

Although SDN plays an important role in enabling network management flexibility and application innovation, SDN is still in its early adoption stage and security is one of the major concerns to make the transition to SDN. Regarding the security issues in the SDN infrastructure layer and the southbound interface, the challenges and open issues include

- Mutual authentication mechanisms. Authentication mechanism leads to trust management and secure identification among the SDN entities such as controllers, switches, and the southbound interface. Secure communication protocols along with the role-based access control and auditing must be done to look for unauthorized access to the controllers [58-59].
- Application development. Today's networking applications are typically developed and deployed by independent third parties. Because networking applications typically reside at the application layer and affect switches through controllers and the northbound API, these applications must be authenticated and authorized to prevent them from becoming flow table overflow attackers [60-61].
- Flow table optimization. Since timeout values are often set for flow entries when entries are generated, the timeout values of flow entries should be asserted in the flow tables. Besides, an encryption strategy can be included to prevent data leakage among switches [62-64].
- Attack-tolerant networks. Considering that SDN has security concerns in controllers, switches, and southbound/northbound APIs, fault-tolerant SDNs aim to operate correctly or provide service that meet the requirements of SLAs when systems are under attack. Although some efforts have been made in traditional networks, how to use the features of SDNs to design and deploy attack-tolerant systems is still an open issue in current research [65-66].

8 Conclusions

This paper provides an overview of the flow table overflow attacks in SDNs. It summarizes the major vulnerabilities in SDN control and data planes and the communication channel between the planes. It points out the flow setup procedure and the limited space of TCAM are the root causing the attacks. It categorizes the attacks into the saturation, the low-rate table exhaustion, and the slow saturation attacks, and further reviews the typical attack models, proposals, detections, and mitigations. It also provides challenges and open issues in this research area.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (61962016) , the Ministry of Science and Technology of China (G2022033002L), the Natural Science Foundation of Guangxi (2022JJA170057), Guangxi Education Department's Project on Improving the Basic Research Ability of Young and Middle-aged Teachers in Universities (grant titile: Research on Statistical Network Delay Predictions in Large-scale SDNs, grant no: 2023ky0812), and Guilin University of Aerospace Technology (grant no: XJ22KT20)

References

[1] Open Networking Fundation, Software-Defined Networking: The New Norm for Networks, *ONF White Paper*, April, 2012.

[2] P. P. Ray, N. Kumar, SDN/NFV architectures for edge-cloud oriented IoT: A systematic review, *Computer Communications*, Vol. 169, pp. 129-153, March, 2021.

[3] S. Wijethilaka, M. Liyanage, Survey on network slicing for Internet of Things realization in 5G networks, *IEEE Communications Surveys & Tutorials*, Vol. 23, No. 2, pp. 957-994, Secondquarter, 2021.

[4] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiguzzaman, D. O. Wu, Edge computing in industrial internet of things: Architecture, advances and challenges, *IEEE Communications Surveys & Tutorials*, Vol. 22, No. 4, pp. 2462-2488, Fourthquarter, 2020.

[5] Open Networking Fundation, OpenFlow specification *Version 1.0.0 (Wire Protocol 0×01)*, [online] Available: https://opennetworking.org/wp-content/uploads/2013/04/openflow-spec-v1.0.0.pdf, *ONF Write Paper*, December, 2009.

[6] R. Deb, S. Roy, A comprehensive survey of vulnerability and information security in SDN, *Computer Networks*,

Vol. 206, Article No. 108802, April, 2022.

[7] A. Curtis, J. C. Mogul, J. Tourrihes, P. Yalagandula, P. Sharma, S. Banerjee, DevoFlow: Scaling flow management for high-performance networks, *2011 ACM SIGCOMM Conference*, Toronto, Ontario, Canada, August, 2011, pp. 254-265.

[8] A. Abdou, P. C. V. Oorschot, T. Wan, Comparative analysis of control plane security of SDN and conventional networks, *IEEE Communications Surveys & Tutorials*, Vol. 20, No. 4, pp. 3542-3559, Fourthquarter, 2018.

[9] T. Han, S. R. U. Jan, Z. Tan, M. Usman, M. A. Jan, R. Khan, Y. Xu, A comprehensive survey of security threats and their mitigation techniques for next-generation SDN controllers, *Concurrency and Computation: Practice and Experience*, Vol. 32, No. 16, Article No. e5300, August, 2020.

[10] T. Ubale, A. K. Jain, Survey on DDoS attack techniques and solutions in software-defined network, in: B. Gupta, G. Perez, D. Agrawal, D. Gupta (Eds.), *Handbook of computer networks and cyber security: Principles and paradigms*, Springer, Cham, 2020, pp. 389-419.

[11] T. Dargahi, A. Caponi, M. Ambrosin, G. Bianchi, M. Conti, A survey on the security of stateful SDN data planes, *IEEE Communications Surveys & Tutorials*, Vol. 19, No. 3, pp. 1701-1725, Thirdquarter, 2017.

[12] A. Shaghaghi, M. A. Kaafar, R. Buyya, S. Jha, Software-defined network (SDN) data plane security: issues, solutions, and future directions, in: B. Gupta, G. Perez, D. Agrawal, D. Gupta (Eds.), *Handbook of Computer Networks and Cyber Security: Principles and paradigms*, Springer, Cham, 2020, pp. 341-387.

[13] Q. Yan, F. R. Yu, Q. Gong, 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 communications surveys & tutorials*, Vol. 18, No. 1, pp. 602-622, Firstquarter, 2016.

[14] A. A. Alashhab, M. S. M. Zahid, M. A. Azim, M. Y. Daha, B. Isyaku, S. Ali, A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks, *Symmetry*, Vol. 14, No. 8, Article No. 1563, August, 2022.

[15] T. A. Pascoal, I. E. Fonseca, V. Nigam, Slow denial-of-service attacks on software defined networks, *Computer Networks*, Vol. 173, Article No. 107223, May, 2020.

[16] J. C. C. Chica, J. C. Imbachi, J. F. Botero Vega, Security in SDN: A comprehensive survey, *Journal of Network and Computer Applications*, Vol. 159, Article No. 102595, June, 2020.

[17] O. S. Al-Heety, Z. Zakaria, M. Ismail, M. M. Shakir, S. Alani, H. Alsariera, A comprehensive survey: Benefits, services, recent works, challenges, security, and use cases for sdn-vanet, *IEEE Access*, Vol. 8, pp. 91028-91047, May, 2020.

[18] I. Farris, T. Taleb, Y. Khettab, J. Song, A survey on emerging SDN and NFV security mechanisms for IoT systems, *IEEE Communications Surveys & Tutorials*, Vol. 21, No. 1, pp. 812-837, Firstquarter, 2019.

[19] R. Khan, P. Kumar, D. N. K. Jayakody, M. Liyanage, A survey on security and privacy of 5G technologies: Potential solutions, recent advancements, and future directions, *IEEE Communications Surveys & Tutorials*, Vol. 22, No. 1, pp. 196-248, Firstquarter, 2020.

[20] B. Agborubere, E. Sanchez-Velazquez, OpenFlow communications and TLS security in software-defined networks, *IEEE International Conference on on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Exeter, UK, June, 2017, pp. 560–566.

[21] K. Benton, L. J. Camp, C. Small, Openflow vulnerability assessment, *2nd ACM SIGCOMM Workshop on Hot Topics in Software Defined Netw*orking, Hong Kong China, August, 2013, pp. 151-152.

[22] J. H. Lam, S. G. Lee, H. J. Lee, Y. E. Oktian, Securing distributed SDN with IBC, *7th International Conference on Ubiquitous and Future Networks*, Sapporo, Japan, July, 2015, pp. 921-925.

[23] J. H. Lam, S. G. Lee, H. J. Lee, Y. E. Oktian, Securing SDN southbound and data plane communication with IBC, *Mobile Information Systems*, Vol. 2016, Article No. 1708970, August, 2016.

[24] R. Kandoi, M. Antikainen, Denial-of-service attacks in OpenFlow SDN networks, *2015 IFIP/IEEE International Symposium on Integrated Network Management*, Ottawa, Ontario, Canada, May, 2015, pp. 1322-1326.

[25] J. Xu, L. Wang, Z. Xu, An enhanced saturation attack and its mitigation mechanism in software-defined networking, *Computer Networks*, Vol. 169, Article No. 107092, March, 2020.

[26] T. A. Pascoal, Y. G. Dantas, L. E. Fonseca, V. Nigam, Slow TCAM exhaustion DDoS attack, *ICT Systems Security and Privacy Protection: 32nd IFIP TC 11 International Conference, SEC 2017*, Rome, Italy, May, 2017, pp. 17-31.

[27] S. Khorsandroo, A. S. Tosun, White box analysis at the service of low rate saturation attacks on virtual sdn data plane, *IEEE 44th LCN Symposium on Emerging Topics in Networking*, Osnabrueck, Germany, October, 2019, pp. 100-107.

[28] S. Shin, G. Gu, Attacking software-defined networks: a first feasibility study, *2nd ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking*, Hongkong, China, August, 2013, pp. 165-166.

[29] [29]H. Cui, G. O. Karame, F. Klaedtke, R. Bifulco, On the fingerprinting of software-defined networks, *IEEE Transactions on Information Forensics and Secur*ity, Vol. 11, No. 10, pp. 2160-2173, October, 2016.

[30] J. Leng, Y. Zhou, J. Zhang, C. Hu, An inference attack model for flow table capacity and usage: exploiting the vulnerability of flow table overflow in software-defined network, April, 2015, https://arxiv.org/abs/1504.03095.

[31] J. Cao, M. Xu, Q. Li, K. Sun, Y. Yang, J. Zheng, Disrupting SDN via the data plane: a low-rate flow table overflow attack, *13th International Conference on Security and Privacy in Communication Systems,* Niagara Falls, Ontario, Canada, October, 2017, pp. 356-

376.

[32] J. F. Box, Guinness, gosset, fisher, and small samples, *Statistical Science*, Vol. 2, No. 1, pp. 45-52, February, 1987.

[33] S. Xie, C. Xing, G. Zhang, J. Zhao, A table overflow LDoS attack defending mechanism in software-defined networks, *Security and Communication Networks*, Vol. 2021, Article No. 6667922, January, 2021.

[34] M. A. Aladaileh, M. Anbar, I. H. Hasbullah, Y.-W. Chong, Y. K. Sanjalawe, Detection techniques of distributed denial of service attacks on software-defined networking controller–a review, *IEEE Access*, Vol. 8, pp. 143985-143995, August, 2020.

[35] Y. Qian, W. You, K. Qian, Openflow flow table overflow attacks and countermeasures, *European Conference on Networks and Communications*, Athens, Greece, June, 2016, pp. 205-209.

[36] M. Dhawan, R. Poddar, K. Mahajan, V. Mann, SPHINX: detecting security attacks in software-defined networks, *22nd Annual Network and Distributed System Security Symposium, NDSS 2015*, San Diego, California, USA, 2015, pp. 8-11.

[37] R. Klöti, V. Kotronis, P. Smith, OpenFlow: a security analysis, *2013 21st IEEE International Conference on Network Protocols (ICNP)*, Goettingen, Germany, October, 2013, pp. 1-6.

[38] Y. Shen, C. Wu, D. Kong, Q. Cheng, Flow Table Saturation Attack against Dynamic Timeout Mechanisms in SDN, *Applied Sciences*, Vol. 13, No. 12, Article No. 7210, June, 2023.

[39] L. Dridi, M. F. Zhani, SDN-guard: DoS attacks mitigation in SDN networks, *2016 5th IEEE International Conference on Cloud Networking (Cloudnet)*, Pisa, Italy, October, 2016, pp. 212-217.

[40] L. X. Liao, X. Ma, C. Zhao, Z. Li, H.-C. Chao, FEAROL: Aging Flow Entries Based on Local Staircase Randomized Response for Secure SDN Flow Tables, *Applied Sciences*, Vol. 13, No. 5, Article No. 2985, March, 2023.

[41] H. Wang, L. Xu, G. Gu, Floodguard: a dos attack prevention extension in software-defined networks, *45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, Rio de Janeiro, Brazil, June, 2015, pp. 239-250.

[42] T. Xu, D. Gao, P. Dong, C. H. Foh, H. Zhang, Mitigating the table-overflow attack in software-defined networking, *IEEE Transactions on Network and Service Management*, Vol. 14, No. 4, pp. 1086-1097, December, 2017.

[43] M. Wang, H. Zhou, J. Chen, B. Tong, An approach for protecting the OpenFlow switch from the saturation attack, *4th National Conference on Electrical, Electronics and Computer Engineering*, Xi'an, China, December, 2015, pp. 729-734.

[44] S. Khamaiseh, E. Serra, D. Xu, Vswitchguard: Defending openflow switches against saturation attacks, *IEEE 44th Annual Computers, Software, and Applications Conference*, Madrid, Spain, July, 2020, pp. 851-860.

[45] M. Dominguez-Limaico, E. Maya-Olalla, C. Bosmediano-Cardenas, C. Escobar-Teran, J. F. Chafla-Altamirano, A. Bedon-Chamorro, Machine Learning in an SDN Network Environment for DoS Attacks, *International Conference on Knowledge Society: Technology, Sustainability and Educational Innovation*, Ibarra, Ecuador, July, 2019, pp. 231-243.

[46] M. S. Elsayed, N. A. Le-Khac, S. Dev, A. D. Jurcut, Ddosnet: A deep-learning model for detecting network attacks, *IEEE 21st International Symposium on A World of Wireless, Mobile and Multimedia Networks*, Cork, Ireland, August, 2020, pp. 391-396.

[47] B. Nugraha, N. Kulkarni, A. Gopikrishnan, Detecting adversarial DDoS attacks in software-defined networking using deep learning techniques and adversarial training, *2021 IEEE International Conference on Cyber Security and Resilience*, Rhodes, Greece, July, 2021, pp. 448-454.

[48] D. Kong, C. Wu, Y. Shen, X. Chen, H. Liu, D. Zhang, TableGuard: A Novel Security Mechanism Against Flow Table Overflow Attacks in SDN, *GLOBECOM 2022- IEEE Global Communications Conference*, Rio de Janeiro, Brazil, December, 2022, pp. 4167-4172.

[49] Z. Wu, Q. Xu, J. Wang, M. Yue, L. Liu, Low-rate DDoS attack detection based on factorization machine in software defined network, *IEEE Access*, Vol. 8, pp. 17404-17418, January, 2020.

[50] K. M. Sudar, P. Deepalakshmi, Flow-Based Detection and Mitigation of Low-Rate DDOS Attack in SDN Environment Using Machine Learning Techniques, in: P. Nayak, S. Pal, S. L. Peng (Eds.), *IoT and Analytics for Sensor Networks*, *Vol. 244*, Springer, Singapore, 2022, pp. 193-205.

[51] S. Gao, Z. Peng, B. Xiao, A. Hu, K. Ren, FloodDefender: Protecting data and control plane resources under SDN-aimed DoS attacks, *IEEE INFOCOM 2017- IEEE Conference on Computer Communications*, Atlanta, GA, USA, May, 2017, pp. 1-9.

[52] B. Yuan, D. Zou, S. Yu, H. Jin, W. Qiang, J. Shen, Defending against flow table overloading attack in software-defined networks, *IEEE Transactions on Services Computing*, Vol. 12, No. 2, pp. 231-246, March-April, 2019.

[53] J. H. Cho, D. P. Sharma, H. Alavizadeh, S. Yoon, N. Ben-Asher, T. J. Moore, D. S. Kim, K. Lim, F. F. Nelson, Toward proactive, adaptive defense: A survey on moving target defense, *IEEE Communications Surveys & Tutorials*, Vol. 22, No. 1, pp. 709-745, Firstquarter, 2020.

[54] P. Kampanakis, H. Perros, T. Beyene, SDN-based solutions for moving target defense network protection, *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Sydney, NSW, Australia, June, 2014, pp. 1-6.

[55] J. H. Jafarian, E. Al-Shaer, Q. Duan, An effective address mutation approach for disrupting reconnaissance attacks, *IEEE Transactions on Information Forensics and Security*, Vol. 10, No. 12, pp. 2562-2577, December, 2015.

[56] D. C. MacFarland, C. A. Shue, The SDN shuffle:

creating a moving-target defense using host-based software-defined networking, *2nd ACM Workshop on Moving Target Defense*, Denver, Colorado, USA , October, 2015, pp. 37-41.

[57] S. Banerjee, K. Kannan, Tag-in-tag: efficient flow table management in SDN switches, *10th IEEE International Conference on Network and Service Management*, Rio de Janeiro, Brazil, November, 2014, pp. 109-117.

[58] S. Garg, K. Kaur, G. Kaddoum, S. H. Ahmed, D. N. K. Jayakody, SDN-based secure and privacy-preserving scheme for vehicular networks: A 5G perspective, *IEEE Transactions on Vehicular Technology*, Vol. 68, No. 9, pp. 8421-8434, September, 2019.

[59] S. A. Latif, F. B. X. Wen, C. Iwendi, L.-L. F. Wang, S. M. Mohsin, Z. Han, S. S. Band, AI-empowered, blockchain and SDN integrated security architecture for IoT network of cyber physical systems, *Computer Communications*, Vol. 181, pp. 274-283, January, 2022.

[60] N. Z. Bawany, J. A. Shamsi, SEAL: SDN based secure and agile framework for protecting smart city applications from DDoS attacks, *Journal of Network and Computer Applications*, Vol. 145, Article No. 102381, November, 2019.

[61] T. Hu, Z. Zhang, P. Yi, D. Liang, Z. Li, Q. Ren, Y. Hu, J. Lan, SEAPP: A secure application management framework based on REST API access control in SDN-enabled cloud environment, *Journal of Parallel and Distributed Computing*, Vol. 147, pp. 108-123, January, 2021.

[62] B. Leng, L. Huang, C. Qiao, H. Xu, X. Wang, FTRS: A mechanism for reducing flow table entries in software defined networks, *Computer Networks*, Vol. 122, pp. 1-15, July, 2017.

[63] C. Wang, H. Y. Youn, Entry Aggregation and Early Match Using Hidden Markov Model of Flow Table in SDN, *Sensors*, Vol. 19, No. 10, Article No. 2341, May, 2019.

[64] B. Isyaku, M. B. Kamat, K. B. A. Bakar, M. S. M. Zahid, F. A. Ghaleb, Ihta: Dynamic Idle-Hard Timeout Allocation Algorithm Based Openflow Switch, *IEEE 10th Symposium on Computer Applications & Industrial Electronics*, Peneng, Malaysia, April, 2020, pp. 170-175.

[65] M. Du, K. Wang, An SDN-enabled pseudo-honeypot strategy for distributed denial of service attacks in industrial Internet of Things, *IEEE Transactions on Industrial Informatics*, Vol. 16, No. 1, pp. 648-657, January, 2020.

[66] N. O. Ahmed, B. Bhargava, From byzantine fault-tolerance to fault-avoidance: An architectural transformation to attack and failure resiliency, *IEEE Transactions on Cloud Computing*, Vol. 8, No. 3, pp. 847-860, July-September, 2020.

Biographies

Changqing Zhao, senior lecturer with the School of Electronic Information and Automation of Guilin University of Aerospace Technology, China. His research interests include digital image/speech processing and network management and optimization.

Ling Xia Liao, professor in the School of Electronic Information and Automation, Guilin University of Aerospace Technology, China. Her research interests include intelligent network management and optimization, distributed systems, and edge computing.

Han-Chieh Chao, professor and chair of the Department of Electrical Engineering, National Dong Hwa University, Taiwan. His research interests include high speed networks, wireless networks, IPv6 based networks and digital divide.

Roy Xiaorong Lai, co-founder and Chairman of Confederal Networks Inc., Seattle, WA, USA. He is a SM of IEEE. His research interests include wireless networking, artificial intelligence algorithms and applications, and blackchain.

Miao Zhang, professor at Quanzhou University of Information Engineering, China. His research interests include Internet of Things, distributed networks, blockchain, and artificial intelligence algorithms and applications.