IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 31 March 2023, accepted 30 April 2023, date of publication 9 May 2023, date of current version 17 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3274577

RESEARCH ARTICLE

Physical Assessment of an SDN-Based Security Framework for DDoS Attack Mitigation: Introducing the SDN-SlowRate-DDoS Dataset

NOE M. YUNGAICELA-NAULA[1], (Student Member, IEEE), CESAR VARGAS-ROSALES[1], (Senior Member, IEEE), JESUS ARTURO PEREZ-DIAZ[1], (Member, IEEE), EDUARDO JACOB[2], (Senior Member, IEEE), AND CARLOS MARTINEZ-CAGNAZZO[3]

[1]Tecnologico de Monterrey, School of Engineering and Sciences, Monterrey 64849, Mexico
[2]University of the Basque Country UPV/EHU, Faculty of Engineering, 48013 Bilbao, Spain
[3]LACNIC, Technology Management, Montevideo 11400, Uruguay

Corresponding author: Noe M. Yungaicela-Naula (a00821711@tec.mx)

This work was supported in part by the Fondo Regional para la Innovación Digital en America Latina y el Caribe (FRIDA); in part by the Project Red Temática Ciencia y Tecnología para el Desarrollo (CYTED) under Grant 519RT0580; in part by the Ibero-American Science and Technology Program for Development CYTED; in part by the 2020 Seed Fund Award from Tecnologico de Monterrey and the Center for Information Technology Research in the Interest of Society (CITRIS); in part by the Banatao Institute, University of California; in part by the School of Engineering and Sciences, Tecnologico de Monterrey; and in part by the Telecommunications Research Group, Tecnologico de Monterrey.

ABSTRACT Slow-read Distributed Denial of Service (DDoS) attacks are complex to detect and mitigate. Although existing tools allow one to identify these attacks, these tools mainly generate alerts. However, in real scenarios, a large number of attack detection alerts will put the security workforce in a bottleneck, as they will not be able to implement mitigation actions in a complete and timely manner. Furthermore, since most existing security solutions for DDoS attack mitigation are tested using datasets and simulated scenarios, their applicability to production networks could be unfeasible or ineffective due to possibly incomplete assumptions in their design. Therefore, automated security solutions against DDoS attacks are needed not only to be designed but also to be implemented and evaluated in real scenarios. This study presents a Software-Defined Networking (SDN)-based security framework, which automates the monitoring, detection, and mitigation of slow-rate DDoS attacks. The framework is implemented in a physical network that uses equipment from the European Experimental Facility Smart Networks for Industry (SN4I). The results demonstrate that the framework effectively mitigates malicious connections, with a mitigation efficiency between 91.66% – 100% for different conditions of the number of attackers and victims. In addition, the SDN-SlowRate-DDoS dataset is presented, which contains multiple experiments of slow-rate DDoS attacks performed on the real testbed. The resources provided in this security dataset are useful to the scientific and industry communities in designing and testing realistic solutions for intrusion detection systems.

INDEX TERMS Dataset, deep learning, slow-rate DDoS, software defined networking (SDN), intrusion detection system (IDS), intrusion prevention system (IPS).

I. INTRODUCTION

Distributed denial of service (DDoS) attacks cause critical complications in traditional and next-generation networks, such as the fifth-generation communication network (5G). Current solutions of intrusion detection systems (IDSs) alleviate part of the task to countermeasure these attacks, generating detection alerts to the security administration workforce [1]. However, given the huge amount of alerts these systems face, they could become useless, since humans will

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

not be able to handle all alerts in a timely and complete way. Therefore, automated tools to mitigate DDoS attacks are more important than ever [2].

Various efforts have been made to create intelligent mechanisms to detect and mitigate DDoS attacks. Among the most promising solutions are those based on Software-Defined Networking (SDN) technology [3]. However, the lack of collaboration between academia and industry has obstructed the development of security tools that work effectively in production environments. As a result, most of the proposed security solutions have been tested using benchmark datasets or simulated environments [4], [5], [6]. Integrating these solutions into real networks could be unfeasible or ineffective due to potentially inappropriate assumptions considered in their design.

Most research groups and institutions do not have experimental facilities or a physical network to test their solutions and have reported the use of security datasets [7], [8]. However, most existing datasets contain synthetic data captured from simulated networks. Testing detection and mitigation systems with these datasets will result in ineffective or incomplete solutions when deployed in real scenarios. Therefore, a real testbed-based security dataset is fundamental for the design of realistic security solutions. Additionally, researchers interested in designing SDN-based security solutions do not have access to SDN-based datasets. That is, a dataset containing controller-based flow statistics is needed, such that researches can test centralized mechanisms based of these statistics.

In this study, we deploy and assess an automated framework to mitigate DDoS attacks. The framework includes a deep learning (DL)- based IDS to detect attacks and an intrusion prevention system (IPS) to autonomously mitigate detected attacks. The design of this framework is based on our previous study presented in [9] and [10]. Although this framework can be configured to detect any type of DDoS attacks, this study focuses on slow-rate DDoS which turn to be more recent and complex than high-rate DDoS attacks.

A relevant contribution of this study is that the performance of the proposed framework is showcased using physical equipment from the European facility Smart Networks for Industry (SN4I) [11]. In particular, a data center topology is configured using numerous microservers and minicomputers. The proposed automated framework demonstrates effectiveness on maximizing the attack connections mitigated when the number of attackers and victims are varied.

Another important contribution of this work is that we provide an SDN-based dataset for slow-rate DDoS attacks, named SDN-SlowRate-DDoS dataset. This dataset has two components: (i) raw pcap files and (ii) SDN controller-based network flow statistics, which contains traffic information from the real testbed based on SN4I. The resources of the provided dataset contribute to the development of realistic IDS solutions.

In summary, the contributions of this work are as follows.

1) Realistic assessment of an automated DDoS attack mitigation framework using real equipment from the SN4I facility. The results demonstrated a mitigation efficiency between 91.66%–100% for different number of attackers and victims.
2) SDN-based dataset for slow-rate DDoS attacks containing traffic generated using the SN4I facility. The dataset is named SDN-SlowRate-DDoS and is available on [12]. This dataset allows researchers to validate up-to-date realistic solutions of IDS.

The remainder of this paper is organized as follows. Section II explores similar work and highlights the contribution of this study. The proposed framework is described in Section III. Section IV reports the experimental results, including the generation of the SDN-based dataset and the evaluation of the framework. Section V discusses the findings of this study. Finally, the conclusion and future research are presented in Section VI.

II. SIMILAR WORKS

A. SDN-BASED DATASETS FOR SLOW-RATE DDoS ATTACKS

The existence of up-to-date security datasets is fundamental for the development of cutting-edge security solutions to secure traditional and next-generation networks from latest attacks. However, most existing DDoS security datasets contain traditional high-rate or volumetric DDoS attacks, such as TCP-SYN and UDP- flood attacks. These attacks are bandwidth intensive and large scale which use protocols from layer 3 and 4 to flood the network. Although these attacks are still critical to networks, many datasets are already available for researchers. That is the case of KDD99,[1] NSL-KDD [13], ICMPv6-based dataset [14], LATAM-DDoS-IoT [15], CF2-based dataset [16], and CICDDoS2019 [17].

It is worth nothing that a limited number of datasets include more recent and complex DDoS attacks, known as slow-rate DDoS attacks. Slow-rate DDoS attacks use application layer protocols to overwhelm servers. For example, in the slow HTTP read attack,[2] a malicious user sends a pertinent number of connection requests to a victim server. Afterward, the user reads the server's response slowly, but also preventing the server to incur in idle connection timeout. As a result, all server's resources are occupied by the malicious user and denied to legitimate requests. CICDoS2017 dataset [18] is an example of a slow-rate DDoS dataset, which contains 24 hours of network traffic with six application layer DDoS attacks. In spite of providing real traces of most recent DDoS attacks, CICDoS2017 dataset is not oriented to SDN-based security applications. In contrast, among other resources, our dataset provides SDN controller-based statistics which serve to design and test security solutions specific for SDN environments.

Realistic datasets are also of relevant importance in designing and testing effective IDSs. Using simulated

[1] https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[2] https://www.netscout.com/what-is-ddos/slow-read-attacks

network-based datasets could not reflect realistic normal and attack scenarios, and researchers will find difficulties on developing effective and practical IDSs [16]. Most existing datasets captured samples from real testbed, for example, KDD99, NLS-KDD, LATAM-DDoS-IoT, CF20-based dataset, CICDDoS2019, and CICDoS2017. Nevertheless, different from our SDN-SlowRate-DDoS, no previous dataset used an SDN-based testbed with real equipment, which is of high importance to design security applications that are SDN-specific.

Table 1 compares the SDN-SlowRate-DDoS dataset with existing datasets. Unlike most existing datasets, the SDN-SlowRate-DDoS dataset contains the most recent DDoS attacks, namely slow-rate DDoS attacks. Additionally, different from previous datasets, out dataset is obtained using an SDN-based testbed. Furthermore, to promote the development of realistic IDSs, the data are captured from a testbed using real equipment. Finally, the resources provided by our dataset serves to design three types of realistic security solutions to slow-rate DDoS attacks: (i) packet statistics-based IDSs, (ii) Flow statistics-based IDSs, and (iii) SDN-specific IDSs.

B. REALISTIC ASSESSMENT OF IPS

The use of datasets to evaluate IDSs is valuable. However, IPSs' assessment requires network deployment. Although simulated networks, such as using Mininet [19], [20], are useful to effectively evaluate IPSs, they do not allow one to replicate real conditions, such as the behavior of legitimate traffic.

Different authors have recognized the importance of using real scenarios to validate their security solutions. An approach is to use real testbeds deployed with development board-based SDN switches, such as Zodiac FX [21], [22]. Although these devices are designed to be used in laboratories providing real traffic on physical hardware, these devices do not fulfill real conditions such as processing and memory capacity, and number of ports of real devices. Finally, more complete testbed, such as GENI [23], have helped the evaluation of security solutions [24]. These testbeds allow us to test the real-time capabilities of innovative security solutions.

In this study, SN4I facilities were used to evaluate a DDoS mitigation framework. SN4I runs on an Network Function Virtualization (NFV) and SDN-enabled network that interconnects the University of the Basque Country (UPV/EHU) with the Aeronautics Advanced Manufacturing Centre (CFAA) and the Rectorate of the University of the Basque Country. An isolated network slice was created over this infrastructure, including an external SDN controller to manage the created network. In this slice, the proposed framework which includes the IDS and IPS was tested, demonstrating its performance in real scenarios.

III. DL-BASED FRAMEWORK FOR DDoS MITIGATION

Figure 1 shows the proposed framework to detect and minimize DDoS attacks. The five elements of the architecture work as follows. The Routing module performs reactive traffic forwarding and traffic mirroring to the Monitor module. The Monitor processes network packets to obtain traffic flows using CICFlowMeter.[4] These flows are identified by a five-tuple key: (Source IP, Source Port, Destination IP, Destination Port, Protocol), where IP refers to the Internet protocol address. Furthermore, each flow contains 76 features [18]. Feature selection and principal component analysis (PCA) are performed once the IDS receives the network flows.

The IDS uses a trained DL model to classify each network flow as attack or legitimate. Then the IPS uses the IDS output to react against DDoS attacks. Particularly, the IPS decides the action to apply for each network connection. The actions are sent to the Flow Rule Manager, which translates them to flow rules that are installed in the data-plane devices (switches). The Flow Rule Manager continually communicates with the Routing module to avoid installing conflicted rules.

A. DDoS ATTACK DETECTION

Some CICFlowMeter's alternatives to monitor network traffic are Sflow[5] and controller-based monitoring (e.g., Open Network Operating System ONOS statistics[6]). One advantage of CICFlowMeter over other solutions is that it offers numerous significant features, which increase the IDS' performance. Furthermore, this solution maintains network monitoring on the data plane, and the SDN controller is not overwhelmed with this task.

Using a DL model, the IDS identifies active network connections as malicious or legitimate. The IPS periodically consults this information that serves to define the appropriate mitigation actions.

B. DDoS ATTACK MITIGATION

In the proposed mitigation strategy, attack connections are intended to be blocked permanently. In contrast, legitimate connections are contemplated not to be altered, blocked temporarily (if they are affected by the IDS' false positives), or recovered (if they have been previously blocked). To this effect, the workflow of the mitigation strategy is depicted in Figure 2.

Four actions are considered in Figure 2: (i) No Action, (ii) Block Permanently a connection, (iii) Block Temporarily a connection, and (iv) Recover a connection. Furthermore, two lists are used to trace the connection's states: PermanentlyBlocked[] which contains the connections blocked with a timeout of $\tau \gg 0$, and TemporarilyBlocked[] which

[4]https://github.com/CanadianInstituteForCybersecurity/CICFlowMeter
[5]https://sflow.org/
[6]https://api.onosproject.org/2.3.0/apidocs/org/onosproject/net/statistic/

TABLE 1. SDN-SlowRate-DDoS dataset vs existing datasets. HR DDoS = high-rate DDoS, SR DDoS = slow-rate DDoS.

Dataset	HR DDoS	SR DDoS	SDN	Real/Simulated	Resources
KDD99[3]	✓			Real	pcaps
NSL-KDD [13]	✓			Real	pcaps
ICMPv6-based dataset [14]	✓			Simulated	Flow-based statistics
LATAM-DDoS-IoT [15]	✓			Real	pcaps
CF20-based dataset [16]	✓			Emulated	pcaps, CICFlowMeter statistics
CICDDoS2019 [17]	✓			Real	pcaps, CICFlowMeter statistics
CICDoS2017 [18]		✓		Real	pcaps, CICFlowMeter statistics
SDN-SlowRate-DDoS (this work)		✓	✓	Real	pcaps, controller-based statistics

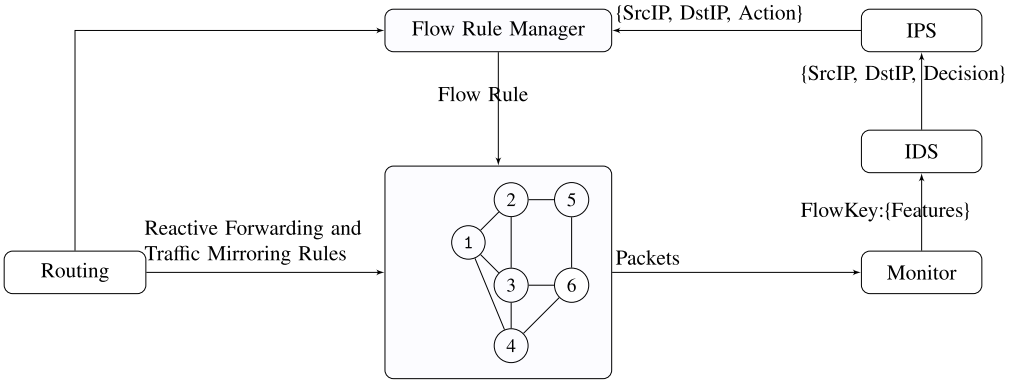

FIGURE 1. Proposed framework for autonomous defense against DDoS attacks. The design is modular, so each component can be improved or replaced.

contains the connections blocked for a limited number of time steps, tbsteps. These lists are initiated when the IPS is executed.

The mitigation strategy works as follows. For a given bidirectional connection i, j that has been permanently blocked in a previous step, no action is taken since the IPS had defined a critical connection. Subsequently, if the connection has been temporarily blocked in a previous step, it is recovered once the temporary connection timeout tbsteps is reached.

If the connection i, j has not been previously blocked (temporarily or permanently), the IPS considers the information received from the IDS to decide among three options: Perform no action, block the connection temporarily, or block the connection permanently. In this sense, mstate is defined for i, j to define how critical the attack is for that connection as

$$\text{mstate}_{i,j} = \frac{1}{1 + e^{-(g_{i,j} - \eta)}}, \tag{1}$$

where

$$g_{i,j} = \sum f_{i,j}, \tag{2}$$

and η helps set the threshold for a critical connection. $f_{i,j}$ in (2) represents the vector of detection ($f_{i,j} = 1$) and non-detection ($f_{i,j} = 0$) in the period between $t - t + \Delta t$.

If the connection has mstate<= α, it is considered non-critical and no action is executed. On the contrary, if the connection has mstate> α, it is temporarily or permanently blocked. Roulette wheel selection is applied using mstate to decide how to block the connection. Connections that are more critical have high values for mstate (which means that there are many detection events for i, j in the period $t - t + \Delta t$), and therefore have a higher chance of being permanently blocked.

Finally, Figure 3 shows the main process of the proposed mitigation method. The IDS is continuously consulted about the network connections. For all active connections, the mitigation strategy shown in Figure 2 is applied. Additionally, a permanently blocked connection is recovered each $\tau \gg 0$. As the mitigation strategy is intended to work autonomously, the condition of recovering any connection after $t/\tau >= 1$ helps increase the availability for legitimate connections affected by an IDS with a high false positive rate (FPR). Naturally, in this operation persistent attackers can be released. However, they will be detected by the IDS and blocked again by the mitigation strategy presented in Figure 2.

The IPS algorithm contains different parameters that can be estimated using experiments. In this work, after evaluating multiple alternatives, the parameters were set as $\tau = 10$ minutes, tbsteps= 20, $\eta = 3$, and $\alpha = 0.2$.

IV. EXPERIMENTS AND RESULTS

A. EXPERIMENTAL SETUP

Figure 4 describes the experimental setup used to test the proposed framework on real SN4I equipment.

The configured data center contains two spine switches and four leaf switches. Two HP Aruba 3810M JL071A

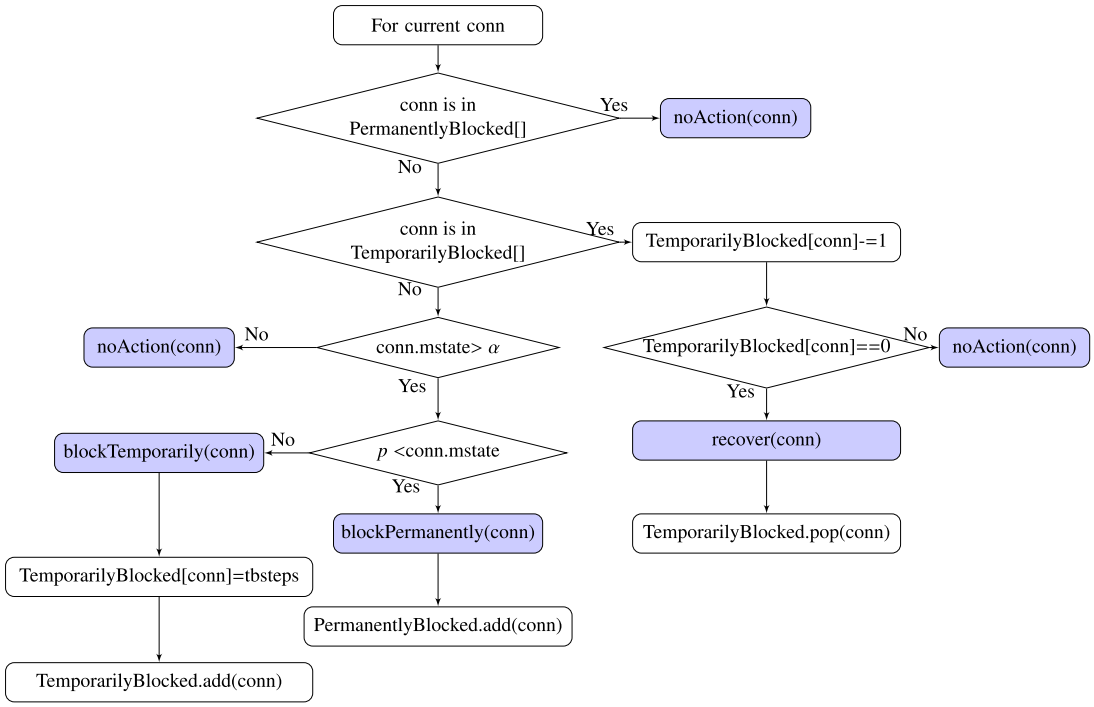

FIGURE 2. Workflow of the mitigation strategy. The shadowed blocks contain the four actions considered. PermanentlyBlocked[] and TemporarilyBlocked[] are initiated when the IPS is executed.

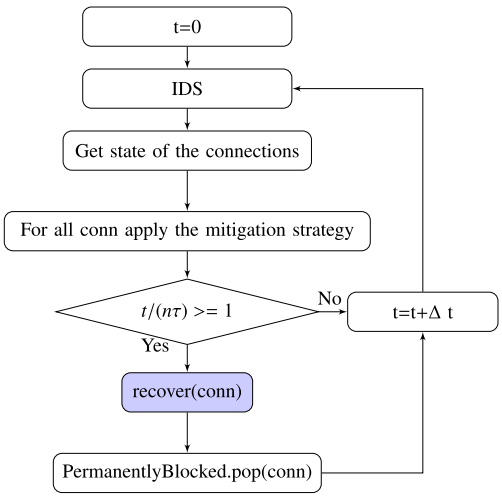

FIGURE 3. Main process for controlling the mitigation strategy.

switches work as spine switches: SW Leioa and SW I2tLab2. Leaf switches VS1, VS2, VS3, and VS4 are virtual switches configured from the physical switch SW I2tLab1 (NEC IP8800/S3640-48T2XW). An additional switch VS5 is needed for traffic monitoring, which is a virtual switch configured from SW I2TLab1.

Each leaf switch is connected to both spine switches. Furthermore, traffic mirroring is performed only by leaf switches. A microserver is connected to the VS5 mirroring switch to capture and process network traffic.

ONOS[7] controller is used to manage the SDN-based data center. The IDS and IPS servers are allocated on a different computer. The I2tLab router commands the management networks 10.98.2.0/24 and 10.98.1.0/24.

The testbed is configured with a set of microservers and minicomputers that run different services and clients. The configured network is 10.0.0.0/24 and contains 20 devices. Table 2 summarizes the equipment used in the experimental setup. Furthermore, Figure 5 shows a photo of the testbed of the SN4I infrastructure located at the Faculty of Engineering in Bilbao.

TABLE 2. Equipment used in the testbed setup.

Component	Value or tool
Host containing ONOS	Ubuntu 20.04.5 LTS, Memory: 16GB, Processor: Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz, CPUs: 8
Host containing IDS and IPS	Ubuntu 20.04.1 LTS, Memory: 8GB, Processor: Intel(R) Core(TM) i7-4700MQ CPU @ 2.40GHz, CPUs: 1
Spine switches	2 switches HP Aruba 3810M JL071A.
Leaf switches	1 switch NEC IP8800/S3640-48T2XW (4 virtualized leaf switches+ 1 mirroring switch).
Hosts in the data plane	3 microservers SuperMicro 101F-8 and 10 minicomputers Orange PI 3 LTS.
Flow Collector	1 microserver SuperMicro 101F-8.

B. LEGITIMATE AND ATTACK TRAFFIC

Table 3 shows the servers and clients configured in the data center to produce legitimate and attack traffic. Three Apache

[7]https://opennetworking.org/onos/

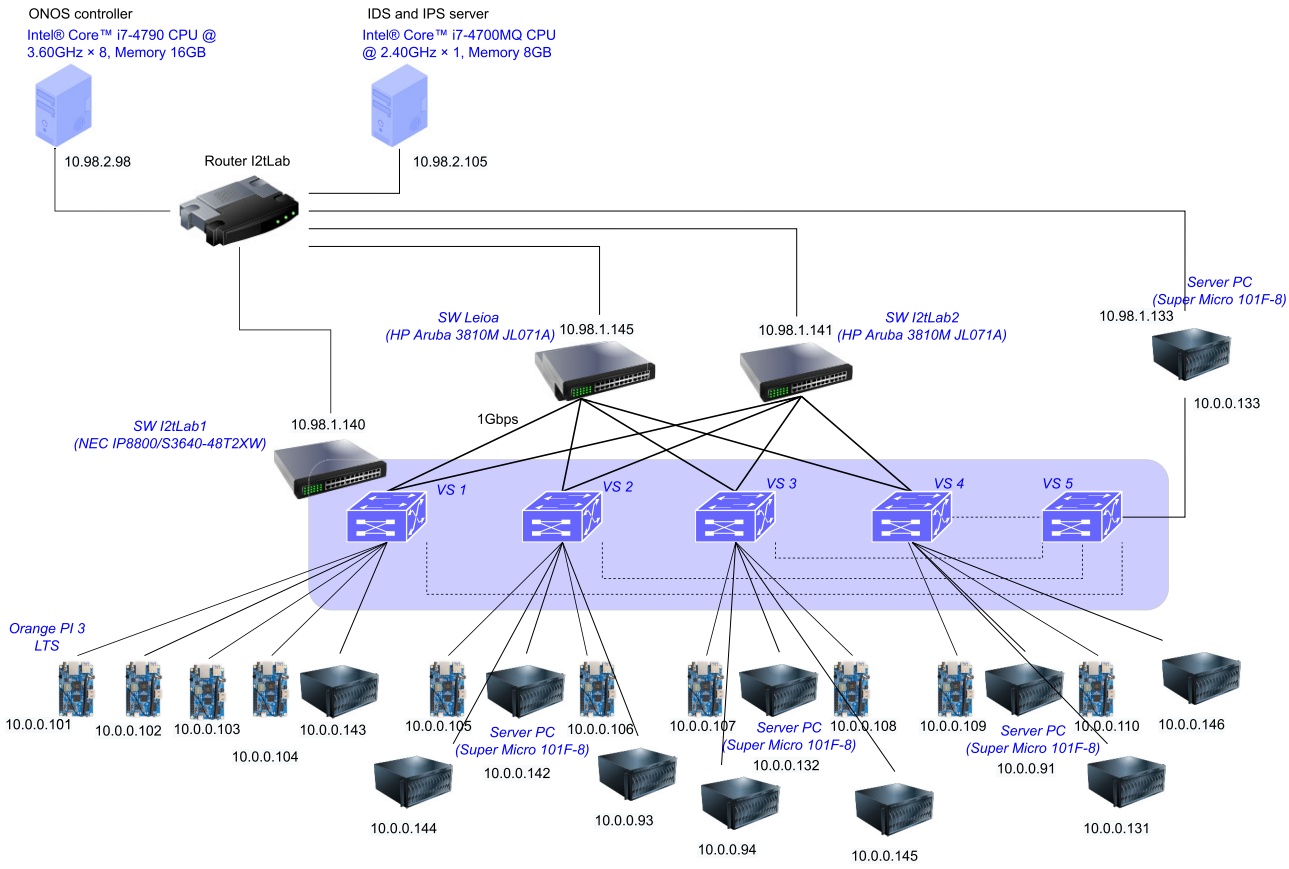

FIGURE 4. SDN-based experimental testbed configured using real equipment from SN4I. Data center topology with two spine switches (SW Leioa and SW I2tLab2) and four leaf switches (VS1-VS4) + one mirroring switch (VS5).

(a) (b)

FIGURE 5. Photo of the experimental setup. (a) Two spine switches (2 switches HP Aruba 3810M JL071A), (b) Leaf switches (1 switch NEC IP8800/S3640-48T2XW with four virtualized switches + one mirroring switch) and 10 minicomputers.

servers (.142, .132, and .91), which act as victim servers, are configured. The attackers are the minicomputers connected to the VS1 (.101, .102, .103, .104). Slowhttptest[8] is used to attack Apache servers.

To generate legitimate traffic, various servers are deployed on the network. Two video streaming services are configured (on minicomputers .107 and .109) using VLC.[9] Clients to these services are configured on the minicomputers .105 and .106. Similarly, two file transfer protocol (FTP) servers and several TCP and UDP servers based on Iperf [10] and clients are deployed for these servers, as shown in Table 3.

[8]https://www.kali.org/tools/slowhttptest/

[9]https://www.videolan.org/vlc/

[10]https://iperf.fr/

TABLE 3. Configuration of servers and clients to produce traffic. Two video streaming servers (VLC), two FTP servers, three Apache servers, and 9 tcp/udp servers (Iperf) are configured.

Client(s)	Server: Service
10.0.0.101/102/103/104	10.0.0.142: Apache Server
10.0.0.144	10.0.0.94: Iperf TCP
10.0.0.93	10.0.0.94: Iperf UDP
10.0.0.105	10.0.0.107: VLC Server
10.0.0.101/102/103/104	10.0.0.132: Apache Server
10.0.0.101/102/103/104	10.0.0.132: Iperf UDP
10.0.0.143	10.0.0.108: FTP Server
10.0.0.143	10.0.0.145: Iperf TCP
10.0.0.144	10.0.0.145: Iperf UDP
10.0.0.106	10.0.0.109: VLC Server
10.0.0.143	10.0.0.131: Iperf TCP
10.0.0.143	10.0.0.131: Iperf UDP
10.0.0.101/102/103/104	10.0.0.91: Apache Server
10.0.0.144	10.0.0.146: Iperf TCP
10.0.0.93	10.0.0.146: Iperf UDP
10.0.0.93	10.0.0.110: FTP Server

C. SDN-SlowRate-DDoS DATASET

Using the data center testbed, a set of experiments was performed to capture a dataset. Table 4 shows the set of experiments included in the SDN-SlowRate-DDoS dataset. In total, 23 experiments were executed, varying different parameters such as the number of attackers (A), number of victims (V), duration of the experiment (T), and attack rate (r). For all experiments, the number of attack connections C was set to 20000 connections. The slow http read attack was selected because it is one of the most harmful slow-rate DDoS attacks [25]. For all experiments with $T = 4000$, legitimate traffic starts at $t = 0$ s and ends at $t = 4000$ s, while attacks are executed from $t = 1000$ to $t = 3200$ s. In the case of experiments with $T = 1200$ s, legitimate traffic starts and ends at $t = 0$ and $t = 1200$ s, respectively, while attack traffic is limited to $t = 300$ to $t = 900$ s. Note that for $A = 0$ and $V = 0$, only legitimate traffic is generated in the network.

The SDN-SlowRate-DDoS dataset offers two components: (i) pcap files captured at the mirroring switch (see VS5 in Figure 4), and (ii) comma separated value (CSV) files containing ONOS statistics of the network flows. As the pcap files are captured at the mirroring switch, they contain all traffic from the network. Researchers interested in evaluating machine learning (ML) or DL models to detect slow-rate DDoS attacks can use applications such as CICFlowMeter, Tshark,[11] NetMate,[12] or ARGUS [26] to capture network flows from these pcap files and train and evaluate their models.

On the contrary, researchers interested in evaluating ML or DL models using centralized monitoring systems based on controller statistics can use the features presented in the CSV files. Table 5 shows the features of the network flows included in the CSV files. The sampling time for these statistics is 1 second.

[11]https://www.wireshark.org/docs/man-pages/tshark.html
[12]https://sourceforge.net/projects/netmate-meter/

TABLE 4. SDN-SlowRate-DDoS dataset with data center topology. For one attacker, .101 attacks .132. For four attackers, .101, .102, .103, and .104 attack .132. For four attackers and two victims, .101 and .102 attack .132, and .103 and .104 attack .91. For three attackers and three victims, .101 attacks .132, .102 attacks .91, and .103 attacks .142. Pcap files contain all network traffic while CSV files contain controller-based flow statistics of the network.

#	Setup	Resources	Size (GB)
1	A=0, V=0, T=4000	CSV, pcap	54.2 GB
2	A=1, V=1, T=4000, r=[30]	CSV	120 MB
3	A=4, V=1, T=4000, r=[30, 20, 30, 25]	CSV	150 MB
4	A=4, V=2, T=4000, r=[20, 30, 30, 25]	CSV	154 MB
5	A=3, V=3, T=4000, r=[30, 30, 30]	CSV	122 MB
6	A=1, V=1, T=4000, r=[100]	CSV	122 MB
7	A=1, V=1, T=4000, r=[25]	CSV	129 MB
8	A=4, V=1, T=4000, r=[10, 15, 10, 15]	CSV	146 MB
9	A=4, V=2, T=4000, r=[10, 15, 10, 15]	CSV	152 MB
10	A=1, V=1, T=4000, r=[50]	CSV	89.4 MB
11	A=0, V=0, T=1200	CSV, pcap	52.8 GB
12	A=1, V=1, T=1200, r=[35]	CSV, pcap	25.1 GB
13	A=4, V=1, T=1200, r=[25, 30, 30, 25]	CSV, pcap	26 GB
14	A=4, V=2, T=1200, r=[20, 20, 20, 20]	CSV, pcap	30 GB
15	A=3, V=3, T=1200, r=[25, 20, 25]	CSV, pcap	28 GB
16	A=1, V=1, T=1200, r=[100]	CSV, pcap	26.1 GB
17	A=4, V=1, T=1200, r=[10, 15, 15, 15]	CSV, pcap	16.2 GB
18	A=4, V=2, T=1200, r=[10, 15, 10, 15]	CSV, pcap	9.1 GB
19	A=1, V=1, T=1200, r=[55]	CSV, pcap	29.5 GB
20	A=2, V=2, T=1200, r=[25, 25]	CSV, pcap	25.4 GB
21	A=4, V=1, T=1200, r=[25, 30, 30, 25]	CSV, pcap	21.7 GB
22	A=4, V=2, T=1200, r=[20, 20, 20, 20]	CSV, pcap	33.2 GB
23	A=4, V=1, T=1200, r=[20, 15, 15, 15]	CSV, pcap	13.3 GB

TABLE 5. Features of the controlled-based statistics of the SDN-SlowRate-DDoS dataset.

#	Feature	Description
1	DeviceId	SWs' IDs. Spine 1: of:012cecebb81c0100, Spine 2: of:012cecebb81c3200, Leaf SWs: of:0000000000000001-of:0000000000000004.
2	FlowId	17-digit number that identifies a flow.
3	IpSrc	Source IP (IPv4)
4	IpDst	Destination IP (IPv4)
5	MacSrc	Source MAC address
6	MacDst	Destination MAC address
7	PortSrc	Source Port
8	PortDst	Destination Port
9	Protocol	Protocol used (e.g. 6: TCP)
10	Bytes	Number of bytes processed by a flow.
11	Packets	Number of packets processed by a flow.
12	Life	Time of life of a network flow in seconds.
13	TimeStamp	Flow's sampling time in nanoseconds.

According to Table 5, each sample contains 13 columns. Nine of these features identify a specific network flow: DeviceID (identification of the SDN switch), FlowId, IpSrc, IpDst, MacSrc, MacDst, PortSrc, PortDst, and Protocol. Furthermore, three relevant flow characteristics are captured, including the number of bytes and packets processed by the flow and the life time of the flow. An additional feature named TimeStamp is provided, which represents the time the flow was sampled. Raw statistics presented in Table 5 can be used to calculate other relevant flow statistics, such as median bytes per flow, median packets per flow, median duration of flow, etc, as presented in [27]. These SDN controller-based

TABLE 6. IDS' performance. Acc. and FPR stand for accuracy and false positive rate, respectively.

Attks.	Victims							
	V= 0		V = 1		V = 2		V = 3	
	Acc.	FPR	Acc	FPR	Acc.	FPR	Acc.	FPR
A=0	81.76	18.24	-	-	-	-	-	-
A=1	-	-	88.19	13.08	-	-	-	-
A=2	-	-	-	-	86.84	12.91	-	-
A=3	-	-	-	-	-	-	88.32	13.66
A=4	-	-	89.23	11.47	90.87	10.75	-	-

statistics are valuable for designing, testing, and evaluating DDoS detection systems using ML or DL.

Finally, note that CSV files are provided for all experiments in Table 4. However, not all experiments contain the pcap files, mainly because our memory limitation during the creation of the dataset. The total size of the dataset is 388 GB.

D. IDS' ASSESSMENT

In this study, the long short-term memory (LSTM) model presented in [10], which was trained using data captured from a simulated datacenter, is used to classify attack and legitimate traffic in the SN4I-based testbed. The data center network topology used to train and evaluate the LTSM model in [10] is similar to the physical network topology used in this study. Therefore, as this model demonstrated high performance and robustness to detect slow-rate DDoS attacks in simulated conditions (it showed an average detection rate of 98% in [10]), in this work the LSTM model's performance was considered to be validated in real conditions with physical equipment. Furthermore, it is worth noticing that the tool (Slowhttptest) used to generate attack traffic in [10] is also the one employed in this study. Nevertheless, in this work legitimate traffic is generated differently from that in [10]. Particularly, traffic of FTP and video streaming is new for the LSTM model.

Table 6 shows the mean values of accuracy and FPR for various experiments that combine different numbers of attackers A and victims V. As the LSTM model was trained with legitimate traffic different from that used for its evaluation, a high level of FPRs is observed. In particular, it was observed that FTP and video streaming traffic caused most false positive events. However, these results are relevant given that the evaluation is performed in a production environment. Previous studies, including our study in [9], demonstrated that ML or DL models can achieve high performance, but most of them were evaluated using datasets or simulated environments.

E. IPS's ASSESSMENT

Table 7 shows the experiments carried out to evaluate the performance of the IPS. The number of attackers A, the number of victims V, the attack rate r, and the duration of the experiment T are varied to obtain a set of 13 experiments. For all experiments, the number of attack connections is set to $C = 20000$.

TABLE 7. Experiments for the evaluation of IPS. For all experiments, C = 20,000 was used.

Exp.	Description
1	$A = 1, V = 1, T = 4000, r$=[35]
2	$A = 1, V = 1, T = 4000, r$=[30]
3	$A = 1, V = 1, T = 1200, r$=[200]
4	$A = 1, V = 1, T = 1200, r$=[40]
5	$A = 4, V = 1, T = 4000, r$=[20, 20, 20, 20]
6	$A = 4, V = 1, T = 1200, r$=[20, 20, 20, 20]
7	$A = 4, V = 1, T = 1200, r$=[25, 20, 20, 15]
8	$A = 3, V = 3, T = 1200, r$=[20, 25, 20]
9	$A = 3, V = 3, T = 1200, r$=[15, 15, 15]
10	$A = 3, V = 3, T = 1200, r$=[30, 35, 30]
11	$A = 4, V = 2, T = 4000, r$=[20, 20, 20, 20]
12	$A = 4, V = 2, T = 1200, r$=[25, 25, 25, 25]
13	$A = 4, V = 2, T = 1200, r$=[20, 20, 20, 20]

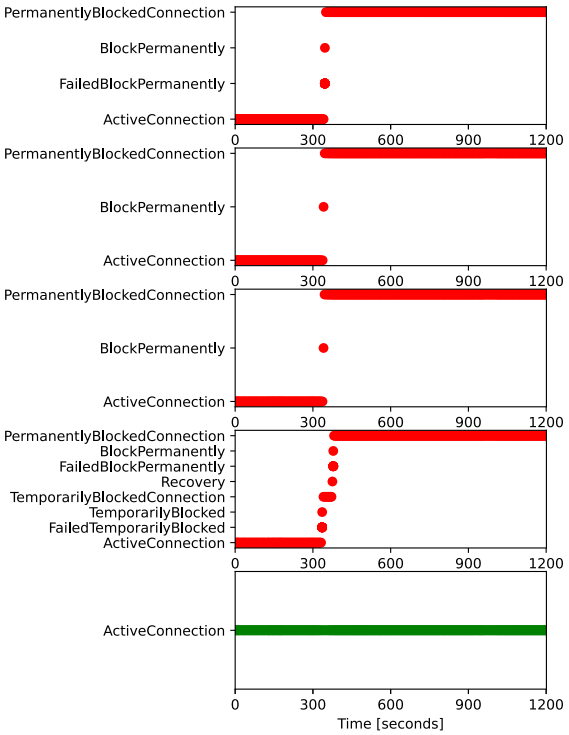

FIGURE 6. Example of the IPS' actions for A=4 and V=1 (experiment #6 from Table 7). Attacks start at t = 300 s and end at t = 900 s. From top to bottom, actions for the four attack connections (red) and for one legitimate connection (green) are shown. All attackers are eventually blocked.

Figure 6 shows an example of the behavior of the IPS for an experiment that involves four attackers ($A = 4$) and one victim ($A = 1$). From top to bottom, the four first images show the IPS' actions for the malicious connections, while the last image shows the IPS' actions for one legitimate connection. It is observed that the first three attackers are correctly blocked. In the case of the fourth attacker, it was initially temporarily blocked and recovered. However, as it still continues to attack the server, it is detected by the IDS and the IPS blocks the attacker permanently.

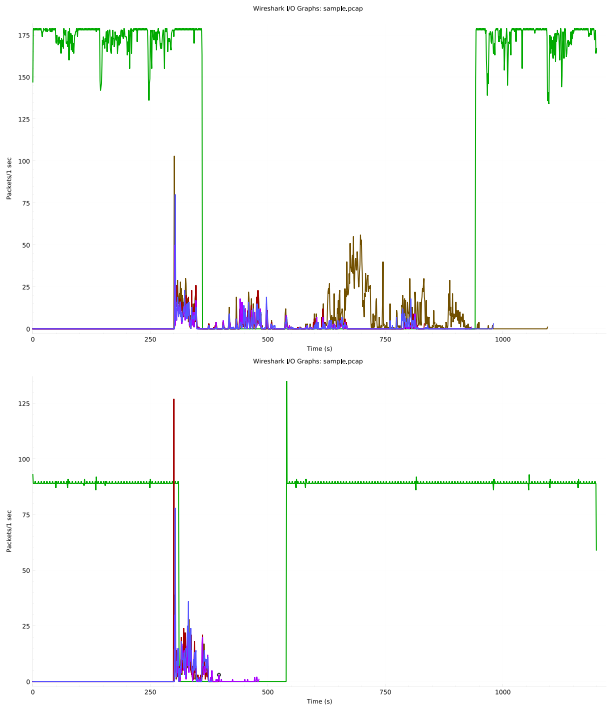

FIGURE 7. Traffic behaviour without IPS (top) and with IPS (bottom) for A=4 and V=1 (experiment # 6 from Table 7). The behaviour for one legitimate connection is shown in green, whereas the four attack connections are shown in colors other than green. When the IPS is activated, all attack traffic is blocked and the legitimate connection is minimally affected.

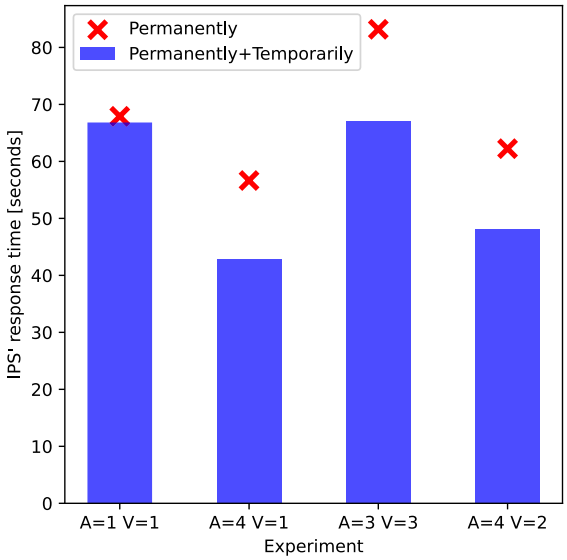

FIGURE 8. IPS' mean response time for four scenarios. The mean of the IPS' response time for all experiments is 53.18 s.

Figure 7 shows the traffic (packets/second) for the sample experiment described above. The traffic behavior of one legitimate connection is shown in green, whereas the traffic behaviors of the four attack connections are shown in other colors, rather than green. The top image of Figure 7 presents the traffic for the selected connections (four attack connections and one legitimate connection) when the IPS is not activated, while the bottom image depicts the traffic when the IPS is activated. The attacks start at $t = 300$ s and end at $t = 900$ s. When IPS is deactivated, attack connections are observed to use most of the network resources, leading to a denial of service to legitimate users. However, when IPS is activated, the attack traffic is blocked and legitimate connections are minimally affected by malicious packets injected by attackers before they are detected and blocked.

According to the IPS design, a connection can be temporarily or permanently blocked depending on how critical the attack is. An effectiveness of 100% of the IPS means that all attack connections are immediately and permanently blocked. Nevertheless, the closed loop of traffic monitoring, attack detection and mitigation requires a minimal execution time. Furthermore, even if the attack connections are temporarily blocked, it certainly helps the system's performance.

In this regard, two metrics are presented in Figure 8 (for all malicious connections and for all experiments presented in Table 7): (i) the mean response time for attacks that are permanently blocked, and (ii) the mean response time for attackers that are temporarily or permanently blocked. It is observed that the former is always lower than the latter. This condition indicates that most of the time the IPS first considers malicious connections as noncritical. However, as they continue to attack the servers, they are identified as critical connections and thus permanently blocked by the IPS.

There is no a clear trend of the response time when the number of attackers A and victims V is varied. Furthermore, the mean response time, considering all experiments, for permanently blocking attack connections is 65.95 s, while the mean response time for either temporarily or permanently blocking attack connections is 53.18 s.

Figure 9 depicts the efficiency of the IPS to mitigate slow-rate DDoS attacks. IPS demonstrates 100% of effectiveness in mitigating malicious connections (permanently + temporarily blocking attack connections) for $A = 1$ and $V = 1$, $A = 4$ and $V = 1$, $A = 3$ and $V = 3$. For $A = 4$ and $V = 2$, the IPS's mitigation performance is reduced to 91.66%. Furthermore, the IPS is able to permanently mitigate more than the 75% of the connections for all conditions of A and V.

Finally, Figure 10 shows the mean number of legitimate connections blocked by the IPS. An ideal IPS does not block legitimate connections. However, given that IDSs tend to produce false positives, there is a probability that the IPS blocks legitimate connections.

According to Figure 10, the IPS affects on average less than one legitimate connection for all experiments. However, blocking even a single connection is critical in production environments. In this sense, the IPS included *Recovery* actions when a connection is temporarily or permanently blocked, as shown in Figures 2 and 3. Therefore, the

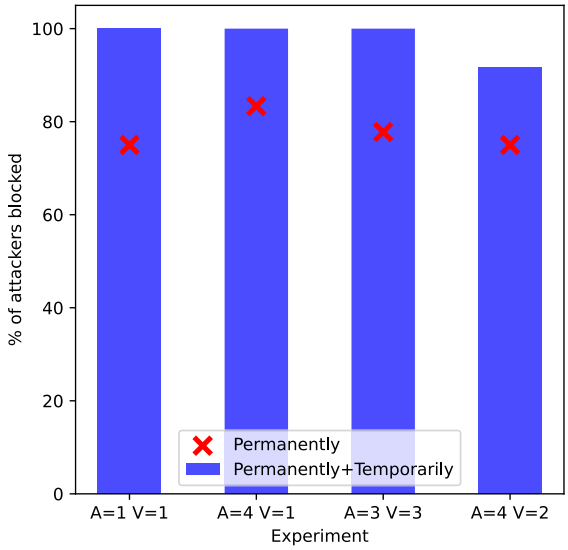

FIGURE 9. Efficiency of the IPS' response to mitigating slow-rate DDoS attacks.

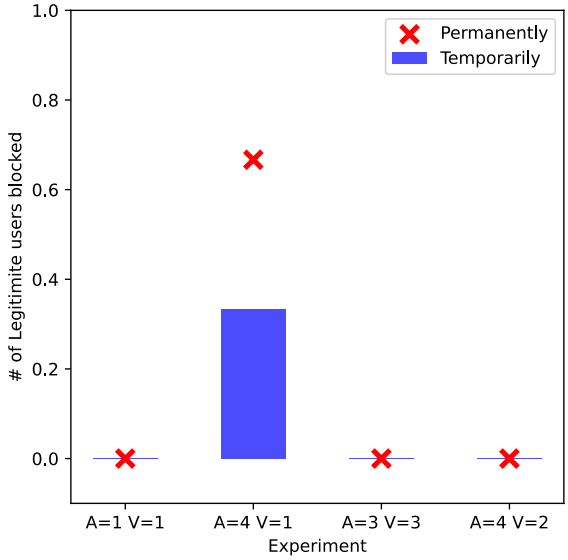

FIGURE 10. Mean value of the number of permanent and temporarily blocked legitimate connections.

IPS releases the negligible number of legitimate blocked connections, and they are minimally interrupted.

V. DISCUSSION

A. FRAMEWORK PERFORMANCE

The LSTM model presented a high performance to detect slow-rate DDoS attacks in a real scenario, even if the data used to train this model are based on simulation [10]. In particular, the legitimate simulated traffic used to train the DL model differed significantly from that of the physical testbed used in this study to test the model. This condition resulted in a mean value of the FPR of 13.23%. One solution to increase the performance of the LSTM model is to retrain it with the traffic collected from the physical testbed. This approach is considered for future work.

Furthermore, even if the DL model did not perfectly separate the attack from legitimate connections, the IPS was totally capable of performing the correct mitigation actions. It timely blocked slow-rate DDoS attacks in mean time <53.18 s. Similar work presented in [28] and [29] demonstrated an average mitigation time of less than 30 s for low-rate DDoS attacks. However, they used simulated testbeds, small topology, and their tests were limited to $A = 1$ and $V = 1$. Consequently, the migration of their solutions to networks with physical equipment will certainly increase their IPSs' response times.

The IPS' mean response time presented in this work (53.18 s) using physical equipment is more than two times the response time obtained in simulations in our previous study in [10], where the IPS' average response time was lower than 20 s for $A = 4$ and $V = 1$. This increase in response time was expected since, unlike the simulated testbed presented in [10], in this work the IDS, IPS, and ONOS controller are placed in physically separated computers interconnected by a router, as presented in Figure 4, which delays the execution of mitigation actions. Nevertheless, this response time is small enough to successfully mitigate the slow-rate DDoS attacks. On top of that, this difference in response times of simulated versus real conditions highlights the importance of validating the design of a security system in production networks in order to ensure that it is effective and scalable.

Finally, the solution proposed in this work did not affect legitimate connections, since the mean number of legitimate connections blocked is less than one. This result is relevant since regardless of whether the IDS presented a high FPR, the IPS was effective in blocking only critical connections.

B. SLOW-RATE DDoS DATASET

The design of an effective security solution is highly dependent on the dataset used to test them. The dataset provided contains data generated with real equipment from the SN4I infrastructure.

Unlike previous security datasets, the two resources provided in our dataset allow interested researchers to design and test three types of IDS: (i) network packet statistics-based IDSs, (ii) flow statistics-based (e.g. CICFlowMeter statistics) IDSs, and (iii) controller statistics-based IDSs.

VI. CONCLUSION

This work presented the evaluation of an automated security framework using real equipment. The results of the detection and mitigation of DDoS attacks showed the high effectiveness of the solution. Additionally, the SDN-SlowRate-DDoS dataset was introduced that collects real traffic with slow-rate DDoS attack events. This dataset will help researchers design and test realistic IDSs.

Furthermore, the proposed framework uses a monitoring switch that captures all traffic from the network, which could

limit the scalability of the solution. Future work considers improving the monitor module by using P4-enabled switches that monitor, filter, and process packets before they are sent to the Monitor module, which will increase the scalability of the framework.

Finally, the results of the proposed framework evaluated in a real scenario indicate that using intelligent mechanisms, it is possible to effectively automate the mitigation of complex DDoS attacks. This work used DL to detect attacks. Using intelligent and adaptive mitigation mechanisms will improve the proposed framework. Therefore, in future work, reinforcement learning-based solutions for DDoS mitigation will be tested using the physical testbed.

ACKNOWLEDGMENT

The authors would like to thank the University of the Basque Country (UPV/EHU) for allowing them the use of the SN4I facilities for performing the experimental tests.

REFERENCES

[1] W. Li, W. Meng, and L. F. Kwok, "Surveying trust-based collaborative intrusion detection: State-of-the-art, challenges and future directions," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 280–305, 1st Quart., 2022.

[2] N. M. Yungaicela-Naula, C. Vargas-Rosales, J. A. Pérez-Díaz, and M. Zareei, "Towards security automation in software defined networks," *Comput. Commun.*, vol. 183, pp. 64–82, Feb. 2022.

[3] P. Goransson, C. Black, and T. Culver, *Software Defined Networks: A Comprehensive Approach*. San Mateo, CA, USA: Morgan Kaufmann, 2016.

[4] M. A. Razib, D. Javeed, M. T. Khan, R. Alkanhel, and M. S. A. Muthanna, "Cyber threats detection in smart environments using SDN-enabled DNN-LSTM hybrid framework," *IEEE Access*, vol. 10, pp. 53015–53026, 2022.

[5] T. G. Nguyen, T. V. Phan, D. T. Hoang, T. N. Nguyen, and C. So-In, "Federated deep reinforcement learning for traffic monitoring in SDN-based IoT networks," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 4, pp. 1048–1065, Dec. 2021.

[6] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proença, "Adversarial deep learning approach detection and defense against DDoS attacks in SDN environments," *Future Gener. Comput. Syst.*, vol. 125, pp. 156–167, Dec. 2021.

[7] Ismail, M. I. Mohmand, H. Hussain, A. A. Khan, U. Ullah, M. Zakarya, A. Ahmed, M. Raza, I. U. Rahman, and M. Haleem, "A machine learning-based classification and prediction technique for DDoS attacks," *IEEE Access*, vol. 10, pp. 21443–21454, 2022.

[8] A. L. Yaser, H. M. Mousa, and M. Hussein, "Improved DDoS detection utilizing deep neural networks and feedforward neural networks as autoencoder," *Future Internet*, vol. 14, no. 8, p. 240, Aug. 2022.

[9] N. M. Yungaicela-Naula, C. Vargas-Rosales, and J. A. Perez-Diaz, "SDN-based architecture for transport and application layer DDoS attack detection by using machine and deep learning," *IEEE Access*, vol. 9, pp. 108495–108512, 2021.

[10] N. M. Yungaicela-Naula, C. Vargas-Rosales, J. A. Pérez-Díaz, and D. F. Carrera, "A flexible SDN-based framework for slow-rate DDoS attack mitigation by using deep reinforcement learning," *J. Netw. Comput. Appl.*, vol. 205, Sep. 2022, Art. no. 103444.

[11] E. J. Taquet, J. Astorga, J. J. U. Galan, M. Huarte, D. G. Conejo, and L. N. L. De La Calle Marcaide, "Towards a 5G compliant and flexible connected manufacturing facility," *Dyna*, vol. 93, no. 1, pp. 656–662, Nov. 2018.

[12] N. M. Yungaicela-Naula, C. Vargas-Rosales, J. A. Pérez-Díaz, E. Jacob, and C. Martinez-Cagnazzo, "SDN-SlowRate-DDoS dataset," 2023, doi: 10.21227/amrt-8y98.

[13] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.

[14] O. E. Elejla, M. Anbar, B. Belaton, and S. Hamouda, "Labeled flow-based dataset of ICMPv6-based DDoS attacks," *Neural Comput. Appl.*, vol. 31, no. 8, pp. 3629–3646, Aug. 2019.

[15] J. G. Almaraz-Rivera, J. A. Perez-Diaz, J. A. Cantoral-Ceballos, J. F. Botero, and L. A. Trejo, "Toward the protection of IoT networks: Introducing the LATAM-DDoS-IoT dataset," *IEEE Access*, vol. 10, pp. 106909–106920, 2022.

[16] S. Alam, Y. Alam, S. Cui, C. Akujuobi, and M. Chouikha, "Toward developing a realistic DDoS dataset for anomaly-based intrusion detection," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2021, pp. 1–6.

[17] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2019, pp. 1–8.

[18] H. H. Jazi, H. Gonzalez, N. Stakhanova, and A. A. Ghorbani, "Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling," *Comput. Netw.*, vol. 121, pp. 25–36, Jul. 2017.

[19] A. Maheshwari, B. Mehraj, M. S. Khan, and M. S. Idrisi, "An optimized weighted voting based ensemble model for DDoS attack detection and mitigation in SDN environment," *Microprocessors Microsystems*, vol. 89, Mar. 2022, Art. no. 104412.

[20] B. Keerthana, M. Balachandra, H. Hebbar, and B. Muniyal, "Creation of SDIoT testbed for DDoS attack using mininet: Experimental study," in *Pervasive Computing and Social Networking*. Singapore: Springer, 2023, pp. 759–772.

[21] S. Wang, K. Gomez, K. Sithamparanathan, M. R. Asghar, G. Russello, and P. Zanna, "Mitigating DDoS attacks in SDN-based IoT networks leveraging secure control and data plane algorithm," *Appl. Sci.*, vol. 11, no. 3, p. 929, Jan. 2021.

[22] S. Wang, J. F. Balarezo, K. G. Chavez, A. Al-Hourani, S. Kandeepan, M. R. Asghar, and G. Russello, "Detecting flooding DDoS attacks in software defined networks using supervised learning techniques," *Eng. Sci. Technol., Int. J.*, vol. 35, Nov. 2022, Art. no. 101176.

[23] M. Berman, J. S. Chase, L. Landweber, A. Nakao, M. Ott, D. Raychaudhuri, R. Ricci, and I. Seskar, "GENI: A federated testbed for innovative network experiments," *Comput. Netw.*, vol. 61, pp. 5–23, Mar. 2014.

[24] R. L. Neupane, T. Neely, P. Calyam, N. Chettri, M. Vassell, and R. Durairajan, "Intelligent defense using pretense against targeted attacks in cloud platforms," *Future Gener. Comput. Syst.*, vol. 93, pp. 609–626, Apr. 2019.

[25] S. Tayama and H. Tanaka, "Analysis of slow read DoS attack and communication environment," in *Mobile and Wireless Technologies 2017*. Singapore: Springer, 2018, pp. 350–359.

[26] P. Kumar, H. Bagga, B. S. Netam, and V. Uduthalapally, "SAD-IoT: Security analysis of DDoS attacks in IoT networks," *Wireless Pers. Commun.*, vol. 122, no. 1, pp. 87–108, Jan. 2022.

[27] R. Braga, E. Mota, and A. Passito, "Lightweight DDoS flooding attack detection using NOX/OpenFlow," in *Proc. IEEE Local Comput. Netw. Conf.*, Oct. 2010, pp. 408–415.

[28] D. Tang, Y. Yan, S. Zhang, J. Chen, and Z. Qin, "Performance and features: Mitigating the low-rate TCP-targeted DoS attack via SDN," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 428–444, Jan. 2022.

[29] D. Tang, X. Wang, Y. Yan, D. Zhang, and H. Zhao, "ADMS: An online attack detection and mitigation system for LDoS attacks via SDN," *Comput. Commun.*, vol. 181, pp. 454–471, Jan. 2022.

NOE M. YUNGAICELA-NAULA (Student Member, IEEE) received the B.Sc. degree in electronic and telecommunication engineering from Universidad de Cuenca, Cuenca, Ecuador, in 2015, and the M.Sc. degree in intelligent systems from Tecnológico de Monterrey, in 2018, where he is currently pursuing the Ph.D. degree. From November 2017 to March 2018, he was a Visiting Scholar with Concordia University, Montreal, QC, Canada. His current research interests include the use of techniques of artificial intelligence to automate different tasks on new generation networks, such as traffic modeling, intrusion detection systems, and network resource optimization.

CESAR VARGAS-ROSALES (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering and in communications and signal processing from Louisiana State University. He is the coauthor of the book *Position Location Techniques and Applications* (Academic Press/Elsevier). His research interests include personal communications, 5G/6G, cognitive radio, MIMO systems, intrusion/anomaly detection in networks, localization, interference, network and channel coding, and optimum receiver design. He is a member of the Mexican National Researchers System (SNI), the Mexican Academy of Science (AMC), and the Academy of Engineering of Mexico. He is an Associate Editor of IEEE Access and *International Journal of Distributed Sensor Networks*. He is a Distinguished Lecturer of the IEEE Communications Society (2021–2022), the IEEE Communications Society Monterrey Chapter Chair, and the Faculty Advisor of the IEEE-HKN Lambda-Rho Chapter with Tecnologico de Monterrey. He was also the Technical Program Chair of the IEEE Wireless Communications and Networking Conference (IEEE WCNC).

JESUS ARTURO PEREZ-DIAZ (Member, IEEE) received the B.Sc. degree in computer science from the Autonomous University of Aguascalientes, in 1995, and the Ph.D. degree in new advances in computer science systems from Universidad de Oviedo, in 2000. He became a Full Associate Professor with University de Oviedo, from 2000 to 2002. Currently, he is a Researcher and a Professor with Tecnologico de Monterrey—Campus Guadalajara, Mexico, and a member of the Mexican National Researchers System. His research interests include cyber-security in SDN and multifactor authentication, where he has supervised several master's and Ph.D. theses and published several articles in international journals. He was recognized by the COIMBRA Group as one of the Best Young Latin-American Researcher, in 2006, and received a research stay with Louvain Le Nouveau University, Belgium. He has been awarded by the CIGRE and Intel for the development of innovative systems. He received the Best Student Award for his B.Sc. degree from the Autonomous University of Aguascalientes.

EDUARDO JACOB (Senior Member, IEEE) received the B.Sc. degree in industrial engineering and the M.Sc. degree in industrial engineering and industrial communications and electronics, in 1987 and 1991, respectively, and the Ph.D. degree in communications engineering from the University of the Basque Country (UPV/EHU), in 2001. During two years, he worked with a public telecommunications research and development enterprise (currently Tecnalia). He spent several years as the IT director in the private sector. Since 1994, he has been full-time with UPV/EHU, where he was elected as the Head of the Department of Communications Engineering, from 2012 to 2016. He is currently a Full Professor and leads the I2T (Engineering and Research on Telematics) Research Laboratory. He has also directed several Ph.D. theses and managed several research projects at the local, national, and European levels. His research interests include applying software-defined networks to industrial communications, cybersecurity in distributed systems, software-defined wireless sensor networks, and in-network processing.

CARLOS MARTINEZ-CAGNAZZO received the B.Sc. degree in electrical engineering from Universidad de la Republica, Montevideo, Uruguay, in 1998. He has worked in the internet industry in different roles, including large network design and network security. He is currently the Chief Technical Officer with Latin American Network Address Registry. His research interests include internet measurements, inter-domain routing, BGP security, IPv6, and DNS. He has taught courses on computer networking and cybersecurity and partnered in different research projects trying to bridge the gap between academia and the industry.