






Evaluating Robustness and Reliability of a C-NIDS for IoT Networks in Virtualized Environments

Agnaldo Batista , Carlos Pedroso , Samuel Brísio , Guilherme Rodrigues , and Aldri Santos 

Abstract—Protecting thousands of smart devices across multiple IoT domains requires robust and collaborative security solutions, such as Collaborative Network Intrusion Detection Systems (C-NIDS). In this context, collaboration among NIDS improves intrusion detection by sharing information about local threats. However, most current solutions neglect the reliability and robustness assessment of these systems, especially when we consider C-NIDS. Thus, the lack of standardized and robust assessments to measure the reliability and accuracy of C-NIDS in attacks detection results in significant uncertainties regarding their ability to correctly identify threats. In this paper, we present an evaluation of a collaborative NIDS that integrates standalone NIDS to share information about detected and mitigated threats, improving overall intrusion detection. The evaluation took place in a virtualized IoT environment and results showed that collaborative NIDS detected DDoS attacks with an accuracy of at least 99.94%, a minimum recall of 99.94%, a precision of 100%, and F1 score of 1 in all evaluated scenarios. Those results highlight the development and adoption of uniform evaluation criteria to ensure the effectiveness and reliability of C-NIDS.

Link to graphical and video abstracts, and to code:
<https://latam.ieee9.org/index.php/transactions/article/view/9287>

Index Terms—Collaborative NIDS, Evaluation, Virtualization, Robustness, Reliability and IoT

I. INTRODUCTION

INTERNET OF THINGS (IoT) technologies have rapidly proliferated, integrating into diverse domains such as energy, healthcare, transportation, and urban infrastructure [1]. However, this proliferation raises significant concerns about the security of these connected devices, which are increasingly vulnerable to attacks [2]. Traditionally, Network Intrusion Detection Systems (NIDS) operate standalone to independently detect and mitigate a range of threats [3]. On the other hand, IoT devices' complexity grows, and integration across different domains challenges these isolated systems. As a result, there is an urgent need for more integrated and robust security approaches to safeguard these networks effectively.

Collaborative NIDS (C-NIDS) emerges as a promising solution by allowing different NIDS to share information and experiences, such as alarms and attack signatures [4]. Collaboration among NIDS makes it possible to build collective knowledge about the threats faced, increasing the accuracy of

intrusion detection [3], [5]. Nonetheless, most current solutions ignore the need to assess the reliability and robustness of these systems, especially for C-NIDS. Although these systems offer significant advantages in improving threat detection through collaborative information sharing among NIDS, the lack of standardized evaluation criteria presents challenges [6]. Without evaluations to assess reliability, accuracy, and resilience, there are significant uncertainties regarding the ability of C-NIDS to identify and mitigate threats consistently and effectively across multiple IoT domains. Therefore, it is essential to address these openings to ensure that C-NIDS can operate with the reliability and robustness required to protect IoT infrastructures against sophisticated threats [7].

Thus, there is a gap in the evaluation of collaborative NIDS, even given their importance for protecting networks such as IoT [3]. Further, those evaluations often ignore the potential benefits of virtualization technologies, which allows a more complete evaluation environment for the performance of the NIDSs. Addressing these challenges, integrating virtualization techniques may significantly improve the evaluation frameworks for NIDS. Virtualization creates isolated, scalable, reproducible test environments where researchers may evaluate NIDS more rigorously [8]. This approach helps to assess NIDS performance by distinct configurations and attack scenarios, thus providing a more detailed and reliable evaluation. Addressing these gaps enables NIDS to foster the accuracy and reliability and facilitate the development of more resilient and adaptable security solutions for integrated IoT environments.

This work comprehensively evaluates the reliability and robustness of a collaborative NIDS (C-NIDS) that leverages autonomous systems to share information about detected and mitigated threats, enhancing intrusion detection. We evaluate the previously proposed and implemented C-NIDS [9] in a virtualized IoT environment, focusing on its ability to mitigate and block attacks effectively. The C-NIDS operates in a distributed and collaborative manner, enabling direct communication among NIDS to reduce the impact of distributed denial-of-service (DDoS) attacks targeting the IoT network. Our evaluation in a virtualized IoT environment demonstrates the C-NIDS's robustness and reliability in protecting the IoT network. The results show that the C-NIDS detected two DDoS attacks with at least 99.94% accuracy, a minimum recall of 99.94%, 100% precision, and an F1 score of 1 across all scenarios. These findings highlight the need for uniform evaluation criteria to ensure the effectiveness and reliability of C-NIDS.

The remaining of paper is organized as follows. Section III details the proposed C-NIDS and the virtualized environment.

The associate editor coordinating the review of this manuscript and approving it for publication was Oscar Mauricio Caicedo (*Corresponding author: Agnaldo de Souza Batista*).

Agnaldo Batista, and C. Pedroso are with Federal University of Parana, Curitiba, Brazil (e-mails: asbatista@inf.ufpr.br, and capjunior@inf.ufpr.br).

S. Brísio, G. Rodrigues, and A. Santos are with Federal university of Minas Gerais, Belo Horizonte, Brazil (e-mails: samuelbrisiko@dcc.ufmg.br, and aldris@dcc.ufmg.br).

Section IV and Section V describe and discuss system performance evaluation, respectively, and Section VI presents conclusions and future works.

II. RELATED WORK

The NIDSs literature has explored several methods to achieve efficient and accurate detection, aiming at system robustness and reliability. Nevertheless, most studies have focused on individual NIDS and have concentrated mainly on NIDS performance [6], [10] or classifier analysis [11], often neglecting a complete reliability and robustness assessment. Table I summarizes and highlights the works limitations. For instance, in [12], researchers presented the trust probability metric to evaluate the effectiveness of collaborative IDSs in Wireless Sensor Networks despite calculating only one metric limited the system's reliability assessment. In [11], an intelligent IDS for Edge of Things networks was developed, focusing on feature selection and machine learning despite its neglected robustness aspects. In [13], an empirical comparative analysis of IDSs was conducted, but the study faced challenges due to individualized assessments and variations in datasets, compromising the consistency and accuracy of comparisons. In [6], the authors proposed a deep learning-based IDS for IoT devices, focusing only on performance while ignoring security metrics. In [10], the authors evaluated the performance of centralized and distributed collaborative IDS models in Multi-Access Edge Computing environments. However, they ignore assessing robustness and reliability metrics. In contrast to existing works, our proposal combines comprehensive metrics, traffic analysis, collaboration, and testing in virtualized environments with realistic scenarios. These points address critical gaps in the literature, enabling a thorough evaluation of reliability and robustness while offering greater practical applicability in combating DDoS attacks in IoT networks.

TABLE I
COMPARISON OF PROPOSED APPROACH WITH EXISTING SOLUTIONS

| Criteria | [12] | [11] | [13] | [6] | [10] | [14] |
|-------------------------|------|------|------|-----|------|------|
| Comprehensive metrics | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ |
| Traffic analysis | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ |
| Collaboration | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ |
| Robustness evaluation | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Virtualized environment | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Realistic scenarios | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |

III. THE COLLABORATIVE NIDS AND TESTBED

This section presents an overview of a collaborative NIDS (C-NIDS) proposed previously by us in [15] to protect IoT networks by ensuring seamless integration. We describe C-NIDS main characteristics and summarize system operation in the face of an attack. Next, we briefly detail the virtual environment – *testbed*¹ – employed to carry out the evaluation.

The C-NIDS encompasses standalone signature-based NIDS connected through the Internet or another network access control, as depicted in Fig. 1. Each NIDS protects a given IoT

network environment named Net_1 , Net_2 , and Net_3 by monitoring the data traffic traversing the gateway. Those NIDS establish connections to communicate and exchange attack-related information, such as attack time, signature, and applied rule. Moreover, they hold rules previously configured to meet the network domain security requirements. In addition, communication among NIDS occurs only when necessary, without impacting communication among IoT devices and SMDs.

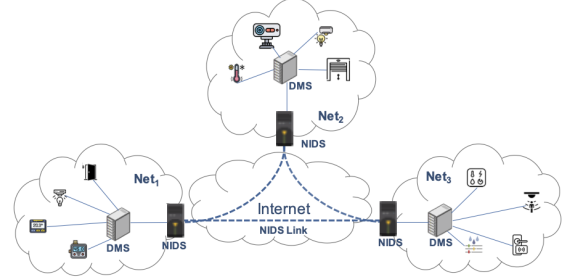


Fig. 1. Integration of IoT networks through a collaborative NIDS.

We take into account a scenario composed by an Island organization model, where a set of IoT devices operates in *islands* to fulfill their application domain operation requirements. Further, these islands periodically communicate with each other to exchange any information. Particularly, as shown in Fig. 1, each island, besides a set of IoT devices and one gateway, comprises one Data Monitoring Server (DMS) and one NIDS. This DMS receives the data sensed from IoT devices and communicates them with each other to keep a complete view of the surveyed environments, for instance, like a smart home environment. The NIDS acts on monitoring the traffic traversing the gateway, and whenever it is under attack, it works to mitigate the detected threat through the available configured rules. In case of successfully, the NIDS exchanges messages through a secure and direct channel to share attack-related information with another NIDS, thus improving overall network security.

All NIDSs belonging to C-NIDS perform based on rules configured according to both the IoT application domain and data traffic patterns from the island to be protected. For example, in *smart home* environment, a NIDS could protect it from unwanted actions like opening the front door and interrupting the heating system. Whereas a *smart industry* environment would demand protection against attempts to interrupt an assembly line to cut the energy supply, among other threats. Therefore, each NIDS requires suitable rules database to deal with possible threats on the environment under protection. Taking into account such individual knowledge, adding new NIDSs to C-NIDS means increasing the base of rules available for the system as a whole. Since the increase in the number of rules makes it possible for C-NIDS to face new threats, its performance benefits from system scalability.

The C-NIDS operation takes place as shown in Fig. 2, where run time instants T_n , for instance, indicate a timely ordered operation. The system starts functioning with the rules configuration of each NIDS (*Step 1*), which runs on the gateway. All the data from one island toward another passes through the gateway, which forwards it to the gateway of that

¹<https://github.com/mentoredproject/WP3-IEEE-LAT-2025>

island. Thus, the island's NIDS monitors the traffic, analyzes the data patterns related to island devices' behavior, and stores the identified patterns as events (*Step 2*). In this way, each NIDS can employ these events to help in the detection of malicious activities (i.e., attack) and mitigation process.

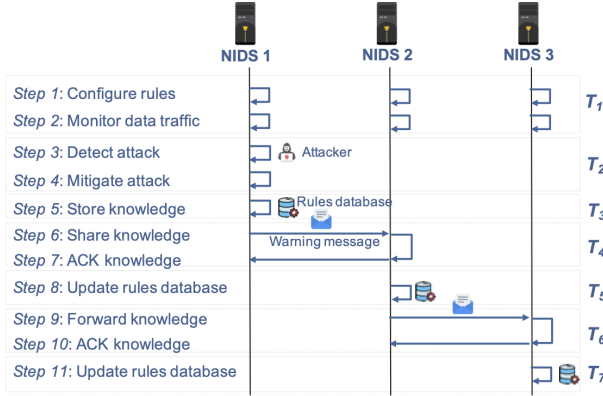


Fig. 2. Sharing attack-related knowledge from standalone NIDS operations.

Whenever a NIDS detects an attack (*Step 3*), it looks up in its rules database for a rule to mitigate this malicious action (*Step 4*) and stores the acquired knowledge in a log database (*Step 5*). In the case of a successful operation to mitigate the threat, as this NIDS belongs to a C-NIDS, the islands' interactions through gateways also allow it to share all the information (knowledge) about attacks detected and mitigated by other NIDS through warning messages (*Step 6*). These NIDSs receive those messages (*Step 7*) and store the shared knowledge in their log database so that they can face new threats (*Step 8*). In this way, C-NIDS contributes to enhancing attack detection and mitigation in various IoT application domains.

A. Testbed

The testbed allowed us to emulate a network environment, as depicted in Fig. 4, to evaluate the C-NIDS and analyze its results. For building this virtual environment, as shown in Fig. 3, we have applied a set of tools for mimicking a realist network environment. We built isolated environments by containers applying Docker and Kubernetes, when Docker version 24.0.2 managed the applications inside the containers.

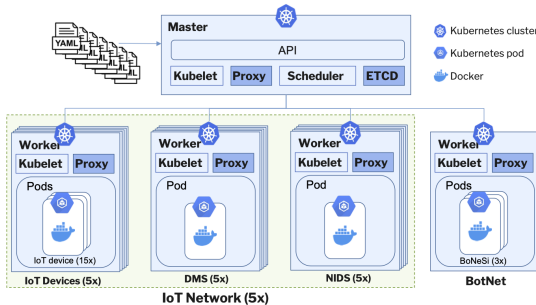


Fig. 3. Testbed structure organization in a virtualized environment.

The execution and management of several containers occurred by Kubernetes version 1.25.3. Further, for interaction

with Kubernetes, we employed Kubectl version 1.27.2, which enabled us to create and configure clusters, pods, and services. Therefore, we could install, inspect, and manage cluster resources and visualize log files. Moreover, we have applied KIND (Kubernetes in Docker) version 0.17.0 [16] to run local Kubernetes clusters by Docker container nodes.

For supporting containers connectivity and security, we make usage of the plug-in Calico Container Network Interface (CNI) version 3.25 [17] through the command line interface Calicoctl version 3.25.0. CNI enabled us to assign a routable IP address to each pod and establish a Layer 3 network, thus providing connectivity for containers and pods. Further, it allowed us to manage security policies, and configure network infrastructure and devices.

IV. TESTING AND EVALUATION

This section presents the methodology employed to evaluate and analyze the performance of each NIDS to face DDoS attacks. We have adopted an approach that allows for a comprehensive analysis of the overall effectiveness of C-NIDS against DDoS attacks. We also present the assessment metrics that take into account NIDS behavior over the traffic traversing the gateway.

A. Methodology for C-NIDS Evaluation

We developed a four-step methodology based on [13] in order to test and evaluate C-NIDS against DDoS attacks emulated by a botnet. Below, we detail each step.

a) Network Deployment: The network's proper operation in the virtual environment is the basis for C-NIDS functioning. Thus, we set up the network environment configuration depicted in Fig. 4 based on the parameters detailed in Section III-A. We emulated five IoT networks with similar infrastructure, each one mimicking a smart home environment. However, the IoT devices collect different data on each network, according to Intel lab data dataset [18]. These devices continuously transmitted the collected data to DMS through the UDP protocol at a frequency of 1 msg/s in order to provide inner network data traffic. DMS exchange messages over the TCP protocol at a frequency of 1 msg/3 s to ensure data traffic among networks.

b) NIDS Configuration: The configuration of each NIDS (IDS Snort [19]) followed the standard setup instructions from the developers. However, we customized the IDS Snort to enable NIDS to share attack-related information. The evaluating process comprised configuring each NIDS with previously defined rules to face the DDoS attacks depicted in Fig. 5. NIDS from Net₁ operated with the following UDP rule:

```
alert udp $EXTERNAL_NET any -> $HOME_NET 50001 (msg:
"Possible UDP DDOS Flood Detection"; detection_
filter:track by_dst,count 1000, seconds 5;
sid:1000021;)
```

NIDS from Net₂ operate with the following ICMP rule:

```
alert icmp $EXTERNAL_NET any -> $HOME_NET 50001
(msg:"Possible ICMP DDOS Flood Detection";
detection_ filter:track by_dst, count 1000, seconds
5; sid:1000022;)
```

We specified both rules with a threshold of 1,000 packets. Thus, these NIDS detect and start blocking the attacks when the packets traversing the gateway exceed such reference level.

Conversely, the NIDS from Net_3 , Net_4 , and Net_5 operated without any rules.

c) *Botnet Deployment*: We have implemented a botnet to mimic a network with devices infected by malware, which a single attacking party controls. They emulated DDoS attacks targeting jeopardize DMSs operation. Fig. 4 depicts the botnet performing attacks on the DMSs of all five networks.

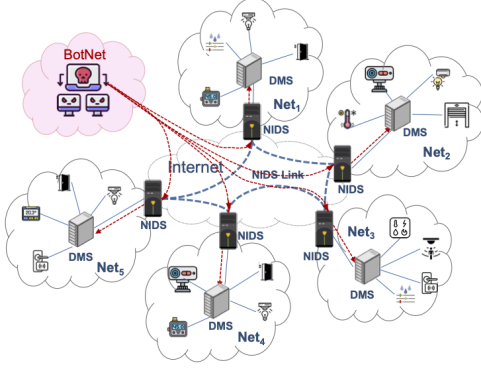


Fig. 4. Operation of a collaborative NIDS against DDoS attacks from a botnet.

We employed the BoNeSi tool version 0.3.1 [20] to simulate botnet traffics in the testbed environment and to help us to investigate the effect of DDoS attacks on C-NIDS. For instance, BoNeSi enables generating ICMP, UDP, and TCP (HTTP) flooding attacks from a defined botnet size with different IP addresses. In the evaluation, we set up a botnet with 3,000 bots, which performed attacks of approximately 35,000 packets per second. As shown in Fig. 3, three instances of the BoNeSi application, each one mounted in one pod and all of them inside of only one worker, compose the botnet. Our scenario in the testbed encompasses two DDoS attacks, one based on the UDP protocol and the other on the ICMP protocol, due to the communication between IoT devices and DMS takes place over the UDP protocol, and the communication among DMS over the TCP protocol, respectively. Those attacks followed the chronology presented in Fig. 5 to mimic a real scenario where DDoS attacks last a maximum of 1.5 minutes [21].

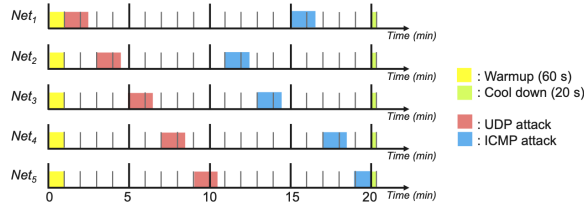


Fig. 5. Chronology of DDoS attacks targeting IoT network data servers.

d) *Signature-based Detection*: Signature-based NIDS acted to detect intrusions during attacks, thus helping us to verify overall C-NIDS operation effectiveness. This process encompassed specifying attack parameters like protocol, duration, traffic volume, and rules, among others. Further, whenever a NIDS under attack successfully mitigated the detected threat, it collaborated with others, sharing attack-related information.

B. Evaluation Metrics

We applied the standard evaluation metrics Accuracy (ACC), Precision (PRC), Recall (REC), and F1-scores (F1) to NIDS evaluation [22]. *Accuracy* (ACC) measures how accurate the NIDS is in detecting normal or malicious traffic behavior, and is defined as

$$ACC = \frac{TP + TN}{TP + TN + FP + FN},$$

Precision (PRC) means how often the NIDS predicts correctly the positive values and is given by

$$PRC = \frac{TP}{TP + FP},$$

Recall (REC) indicates how correctly the NIDS identify the attacks and is defined as

$$REC = \frac{TP}{TP + FN},$$

F1-measure (F1) is a measure of predictive performance and determines NIDS general precision and is given by

$$F1 = 2 \times \frac{PRC \times REC}{PRC + REC},$$

were true positive (TP) is the attack traffic correctly classified, true negative (TN) is the normal traffic correctly classified, false positive (FP) is the normal traffic incorrectly classified, and false negative (FN) is the attack traffic incorrectly classified. Furthermore, we computed packet losses in intra-network and inter-network data traffic to analyze the performance of C-NIDS during emulated attacks.

V. RESULTS AND ANALYSIS

We conduct the C-NIDS evaluation into the attack windows in each network, as seen in Fig. 5. In the first window, the data monitoring server (DMS) of each network was subject to a DDoS attack done over the UDP protocol. In the second window, the attack took place over the ICMP protocol. We took the first 60 seconds of each emulation a warm-up interval and a 20-second cool-down time after each emulation ends. We initially evaluated the collaborative scenario with three IoT networks; then, we increased to five to investigate the influence of network scalability on C-NIDS performance. As expected, the communication between IoT devices and DMS presented similar throughput since the topology of IoT networks remained the same. Conversely, the throughput of DMS communication experienced an increase due to the need for each DMS to exchange information with all other DMSs, i.e., four instead of two from the initial scenario. Consequently, there was an increase in packet losses, although generally, networks operate at low losses. However, C-NIDS performed similarly in both scenarios, corroborating the benefits of a collaborative NIDS to overall network security.

Internal network communication occurred regularly during both attack windows. As shown in Table II, every DMS received almost all the data sent by IoT devices. Such results evidence the effectiveness of the NIDS in protecting internal network operations against external malicious data. Moreover, the traffic traversing the gateway also suffered some losses, since it comprises exchanged data among DMS over the TCP protocol and the data traffic from the botnet to DMS

on each network. For instance, NIDS commonly blocked less than 4.2% of TCP packets during DDoS attacks. Given that we emulated DDoS attacks, the applied rules to the NIDSs operation specified a threshold of 1,000 packets as the maximum limit to start blocking the malicious traffic. Therefore, NIDS blocked at least 99.93% of the botnet traffic, so the DMS of Net₅ received 996 malicious packets. In this sense, the low losses on TCP packets highlight the reliability of NIDS behavior over network communication.

TABLE II

ANALYSIS OF INTRA AND INTER-NETWORK DATA TRAFFIC

| Attack window | Network | Data traffic (# packets) | | | | | | Total packets |
|---------------|------------------|--------------------------|-----------|--------------|--------------|-----------|--------------|---------------|
| | | IoT → DMS | DMS ↔ DMS | DMS → Botnet | Botnet → DMS | DMS → DMS | Botnet → DMS | |
| | | Sent | Rcv | Blk | Rcv | Blk | | |
| 1st | Net ₁ | 1221 | 1219 | 197 | 5 | 871 | 1788360 | 1789435 |
| | Net ₂ | 1199 | 1199 | 171 | 4 | 930 | 1704102 | 1705207 |
| | Net ₃ | 1215 | 1215 | 137 | 0 | 937 | 1721651 | 1722725 |
| | Net ₄ | 1214 | 1208 | 188 | 2 | 939 | 1719044 | 1720179 |
| | Net ₅ | 1200 | 1196 | 139 | 0 | 996 | 1635939 | 1637078 |
| 2nd | Net ₁ | 1220 | 1192 | 180 | 0 | 1022 | 2727569 | 2728799 |
| | Net ₂ | 1221 | 1217 | 180 | 2 | 1095 | 2718552 | 2719833 |
| | Net ₃ | 1222 | 1213 | 103 | 4 | 963 | 2715155 | 2716234 |
| | Net ₄ | 1223 | 1202 | 182 | 0 | 1018 | 2766193 | 2767414 |
| | Net ₅ | 825 | 822 | 71 | 3 | 996 | 1843659 | 1844732 |

Rcv: Received, Blk: Blocked

For calculating the applied metrics to evaluate C-NIDS operation in detecting and blocking malicious data, we computed from data traffic the values of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), as presented in Table III. It is worth noting that the duration of the attacks affects the number of TCP packets that go through the gateways. In our experiment, we took into account the attack duration detailed in Fig. 5 and the frequency of exchanging DMS messages. Thus, the total of TCP packets traversing the gateways ranged from 137 to 197 in the first attack window (UDP protocol), which lasted two minutes, and from 71 to 182 in the second window (ICMP protocol), which lasted two or three minutes according to the targeted network.

TABLE III

ANALYSIS OF STANDALONE NIDS BEHAVIOR AGAINST DDoS ATTACKS

| Protocol | Network | # packets | TP | TN | FP | FN |
|----------|------------------|-----------|---------|-----|----|------|
| UDP | Net ₁ | 1789435 | 1788362 | 197 | 5 | 871 |
| | Net ₂ | 1705207 | 1704102 | 171 | 4 | 930 |
| | Net ₃ | 1722725 | 1721651 | 137 | 0 | 937 |
| | Net ₄ | 1720179 | 1719050 | 188 | 2 | 939 |
| | Net ₅ | 1637078 | 1635943 | 139 | 0 | 996 |
| ICMP | Net ₁ | 2728799 | 2727597 | 180 | 0 | 1022 |
| | Net ₂ | 2719833 | 2718556 | 180 | 2 | 1095 |
| | Net ₃ | 2716234 | 2715164 | 103 | 4 | 963 |
| | Net ₄ | 2767414 | 2766214 | 182 | 0 | 1018 |
| | Net ₅ | 1844732 | 1843662 | 71 | 3 | 996 |

Given that TN and FP presented shallow values when compared mainly with TP, as shown in Table III, they slightly affect NIDS *Accuracy* (ACC) and *Recall* (REC). Besides, such a condition stands out during the second attack window (ICMP protocol), when the number of packets traversing the gateways of Net₁, Net₂, Net₃, and Net₄ increases by more

than 50% when compared to that in the first attack window. Consequently, the TP, TN, FP, and FN values lead both metrics to attain similar values for each network and attack, as shown in Table IV. Taking into account DMS communication, the volume of traffic data from the botnet traversing the gateway is much bigger than the TCP traffic data. However, all NIDS achieved an elevated accuracy in identifying and blocking the malicious data, thus attaining 99.96% in some cases, thus preserving DMS operation. Further, the attack through the ICMP protocol jeopardizes DMS communication slightly more than the attack over the UDP protocol. However, NIDS accuracy in detecting and blocking ICMP attacks is higher than in the face of UDP attacks. NIDS attained REC values up to 99.96%, pointing out that these devices correctly classified almost all traffic traversing the gateway and, consequently, blocked the malicious traffic from the botnet.

TABLE IV

PERFORMANCE OF STANDALONE NIDS AGAINST DDoS ATTACKS

| Attack protocol | Network | ACC (%) | REC (%) | PRC (%) | F1 |
|-----------------|------------------|---------|---------|---------|----|
| UDP | Net ₁ | 99.95 | 99.95 | 100 | 1 |
| | Net ₂ | 99.95 | 99.95 | 100 | 1 |
| | Net ₃ | 99.95 | 99.95 | 100 | 1 |
| | Net ₄ | 99.95 | 99.95 | 100 | 1 |
| | Net ₅ | 99.94 | 99.94 | 100 | 1 |
| ICMP | Net ₁ | 99.96 | 99.96 | 100 | 1 |
| | Net ₂ | 99.96 | 99.96 | 100 | 1 |
| | Net ₃ | 99.96 | 99.96 | 100 | 1 |
| | Net ₄ | 99.96 | 99.96 | 100 | 1 |
| | Net ₅ | 99.95 | 99.95 | 100 | 1 |

All NIDS achieved high values for *Recall* (REC) and *Precision* (PRC), as shown in Table IV. Whereas REC achieved values close to 100%, PRC attained better results, meaning that NIDS successfully predicted the positive values during the attacks in all networks. Such a relation of high REC and high PRC evidences that the NIDSs belonging to C-NIDS correctly classified most of the positive samples. We also computed F1 from PRC and REC to verify the effectiveness of every NIDS prediction. The results indicated that standalone NIDS attained the maximum F1 value, 1, underlining their elevated *Precision* and *Recall* on data traffic classification, thus blocking malicious activities against the network. Further, the alignment of these metrics indicates a balanced performance, where false positives and false negatives are minimized, thus enhancing the overall reliability of NIDSs.

For evaluating C-NIDS operation, we started by configuring every NIDS, but disregarding any rules to face DDoS attacks. Such a strategy aimed to enable us to analyze the impact of the attacks on the data traffic traversing the gateways without NIDS protection. As shown in Fig. 6(a), 6(c), and 6(e), DDoS attacks compromise data communication between IoT devices and DMS and among the DMSs. For instance, all DMS lost around 87.5% of data sent by IoT devices (UDP protocol) due to the high volume of attack data. Conversely, since the TCP traffic among DMS naturally organizes itself in the face of DDoS attacks, it presented low losses. Given that the DDoS attacks target DMS, both attacks, regardless of the applied protocol, jeopardized the data traffic from IoT devices to DMS

similarly.

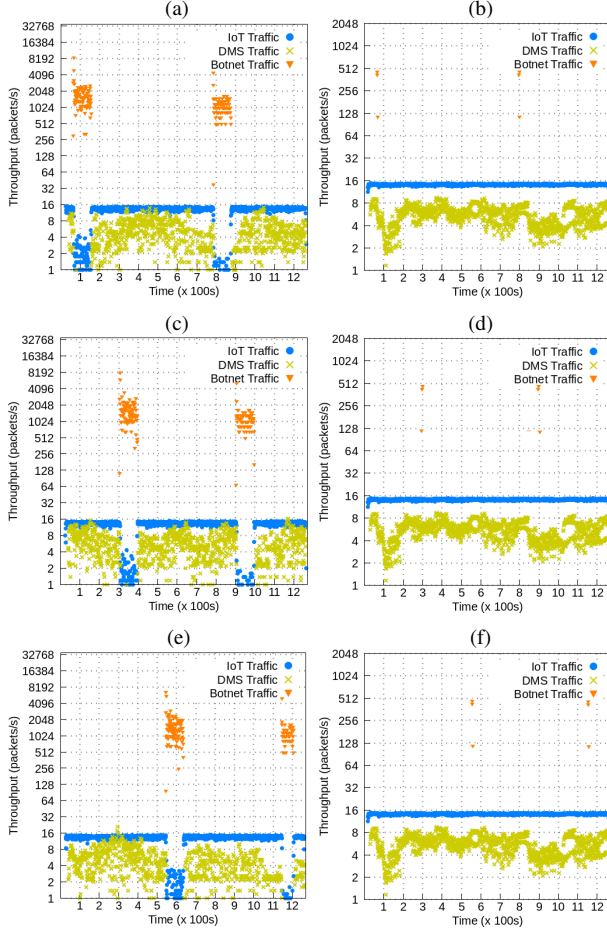


Fig. 6. Throughput of Net₁ (a, b), Net₃ (c, d) and Net₅ (e, f) when NIDS operate without (a, c, e) and with (b, d, f) preconfigured rules.

To evaluate C-NIDS performance in the face of DDoS attacks, we configured NIDS of Net₁ and Net₂ with suitable rules to meet the chronology of attacks presented in Fig. 5 and according to the details in Section IV-Ab. Then, we conducted 35 emulations and computed the packet losses to each network during the attacks. Table V presents the averaged values for packet loss according to the attack window. It is worth noting that the C-NIDS operation reduced the loss of UDP packets during the attack windows to a maximum of 28.43 packets, although all networks operated during the second attack window with UDP packet losses. Despite NIDS of Net₃, Net₄, and Net₅ operating without specific rules to face both attacks, they benefited from collaborating with other NIDS to protect its network. Therefore, it allowed IoT devices to exchange messages with DMS regularly. Commonly, DDoS attacks aim to overwhelm a target device or network, thus preventing users from accessing the provided services. In our experiment, the attacks over the ICMP protocol lead to low packet loss of TCP packets in Net₁ and Net₄, mainly the DMS from Net₅, which lost a maximum of 2.63 from 74 packets, 3.5%. Fig. 6(b), 6(d), and 6(f) depict standalone NIDS performance under the attacks, which evidences C-NIDS overall resilience. Although NIDS from Net₃ and Net₅ operate without preconfigured rules to face DDoS attacks,

they function similarly to other NIDS from C-NIDS. Such a performance points out the relevance of collaboration in C-NIDS, given that a NIDS receives rules from others and becomes resilient to new attacks.

TABLE V
PACKET LOSSES IN INTRA AND INTER-NETWORK
COMMUNICATION DURING DDoS ATTACKS

| Attack Window Network | 1st attack (UDP) | | | | | 2nd attack (ICMP) | | | | |
|--------------------------|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|------------------|------------------|
| | Net ₁ | Net ₂ | Net ₃ | Net ₄ | Net ₅ | Net ₁ | Net ₂ | Net ₃ | Net ₄ | Net ₅ |
| # UDP packets | 2.23 | 0.3 | 0.06 | 6.1 | 3.86 | 28.43 | 3.36 | 8.6 | 24.53 | 3.1 |
| # TCP packets | 5.13 | 3.13 | 0 | 1.4 | 0 | 0.3 | 2.5 | 4 | 0 | 2.63 |

Despite the promising results, such evaluation also underscores a key area for improvement. NIDS proper operation relies on preconfigured rules to minimize packet losses during attacks. For instance, DMS lost around 87.5% of data sent by IoT devices in all networks due to the high volume attack data. This scenario calls for adaptive and dynamic rule-generation mechanisms in order to allow NIDS to respond swiftly to new and evolving attack patterns. Conversely, the NIDS operation without preconfigured rules depends on collaborative efforts to maintain their integrity, which suggests that intelligence shared between them can significantly strengthen each system's defense mechanisms. Further, the comparative analysis with non-collaborative NIDS underlines the importance of a more comprehensive evaluation framework to reinforce the benefits of C-NIDS to networks seamless integration.

A. Challenges on Virtual Operation

The testbed demanded particular configurations to ensure the proper functioning of the emulations. Thus, we configured Docker and Kubernetes tools to emulate IoT devices, gateways, and data servers. Further, these tools imposed adjustments in inter-container communication, including plugins such as Calico, to ensure correct and functional address configurations. Additionally, to make the DDoS attack emulation, we configured instances of the BoNeSi, allowing the configuration of enough botnets to evaluate the action of the attack. Nevertheless, when configuring the tool, a single BoNeSi instance was enough to congest the network, so more instances were needed to emulate the attacks and deliver the expected impact. Lastly, while OVS virtual switches offer advanced features for interconnecting multiple networks and configuring routing, their implementation in a virtual environment was non-trivial because all virtualization involved precise adjustments since emulating the operation and robustness of physical switches in OVS demands in-depth knowledge of virtual networks and communication protocols of IoT devices. In addition, configuring jointly OVS, gateway, and NIDS emerged as a challenge so that they could run together.

Enhancing the virtualization of IoT testbeds depends on the continuous adaptation and customization of available tools, as well as an innovative approach to overcome constraints in the representation of IoT devices and emulation of attack models. The evolution of these solutions is essential for creating models capable of accurately assessing the robustness and reliability of C-NIDS. As these technologies advance, building testbeds increasingly similar to reality will be possible, foster-

ing the evolution of more secure and robust IoT applications.

VI. CONCLUSION

This work presented a comprehensive evaluation of the robustness and reliability of a C-NIDS in a virtualized IoT environment. C-NIDS allows the integration of autonomous systems by sharing information about detected and mitigated threats to improve overall security. It reduces the damage caused by DoS attacks targeting an IoT. The results demonstrate C-NIDS reliability and robustness in protecting IoT networks against such attacks. Moreover, the evaluation in a virtualized environment proved closer to a real one, thus evidencing the importance of testing solutions in scenarios mimicking actual operating conditions. As future work, we intend to expand the evaluation to measure the C-NIDS reliability against other attack models and compare it with other solutions. We also plan to evaluate and integrate new protocols in diverse communication scenarios among IoT devices.

ACKNOWLEDGEMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, FAPESP - grant #2018/23098-0, CNPq - grants #313641/2020-0 and #2307752/2023-2.

REFERENCES

- [1] A. Pastorek and A. Tundis, "Navigating the landscape of IoT security and associated risks in critical infrastructures," in *19th International Conference on Availability, Reliability and Security, ARES '24*, (New York, NY, USA), ACM, 2024. doi: 10.1145/3664476.3669979.
- [2] Y. R. Siwakoti, M. Bhurtel, D. B. Rawat, A. Oest, and R. Johnson, "Advances in IOT security: Vulnerabilities, enabled Criminal Services, attacks and countermeasures," *IEEE Internet of Things Journal*, vol. 10, pp. 11224–11239, mar 2023. doi: 10.1109/IJOT.2023.3252594.
- [3] A. Heidari and M. A. Jabraeil Jamali, "Internet of Things intrusion detection systems: a comprehensive review and future directions," *Cluster Computing*, vol. 26, no. 6, pp. 3753–3780, 2023. doi: 10.1007/s10586-022-03776-z.
- [4] L. K. Vashishtha, A. P. Singh, and K. Chatterjee, "HIDM: A Hybrid Intrusion Detection Model for Cloud Based Systems," *Wireless Personal Communications*, vol. 128, no. 4, pp. 2637–2666, 2023. doi: 10.1007/s11277-022-10063-y.
- [5] L. Zhu, B. Zhao, W. Li, Y. Wang, and Y. An, "TICPS: A trustworthy collaborative intrusion detection framework for industrial cyber-physical systems," *Ad Hoc Networks*, vol. 160, p. 103517, 2024. doi: 10.1016/j.adhoc.2024.103517.
- [6] K. Sood, D. D. N. Nguyen, M. R. Nosouhi, N. Kumar, F. Jiang, M. Chowdhury, and R. Doss, "Performance Evaluation of a Novel Intrusion Detection System in Next Generation Networks," *IEEE Transactions on Network and Service Management*, vol. 20, no. 3, pp. 3831–3847, 2023. doi: 10.1109/TNSM.2023.3242270.
- [7] N. Moustafa, N. Koroniotis, M. Keshk, A. Y. Zomaya, and Z. Tari, "Explainable Intrusion Detection for Cyber Defences in the Internet of Things: Opportunities and Solutions," *IEEE Communications Surveys & Tutorials*, vol. 25, pp. 1775–1807, may 2023. doi: 10.1109/COMST.2023.3280465.
- [8] J. Gomez, E. F. Kfoury, J. Crichigno, and G. Srivastava, "A survey on network simulators, emulators, and testbeds used for research and education," *Computer Networks*, vol. 237, p. 110054, 2023. doi: 10.1016/j.comnet.2023.110054.
- [9] C. Pedroso and A. Santos, "Dissemination control in dynamic data clustering for dense IIoT against false data injection attack," *International Journal of Network Management*, vol. 32, no. 5, p. e2201, 2022. doi: 10.1002/nem.2201.
- [10] R. Sharma, C. A. Chan, and C. Leckie, "Evaluation of Centralised vs Distributed Collaborative Intrusion Detection Systems in Multi-Access Edge Computing," in *2020 IFIP Networking Conference (Networking)*, pp. 343–351, IEEE, 2020. doi: 10.1007/978-3-030-69893-5_15.
- [11] V. Kumar, V. Kumar, N. Singh, and R. Kumar, "Enhancing Intrusion Detection System Performance to Detect Attacks on Edge of Things," *SN Computer Science*, vol. 4, no. 6, p. 802, 2023. doi: 10.1007/s42979-023-02242-w.
- [12] A. Ramos, M. Lazar, R. Holanda Filho, and J. J. Rodrigues, "A security metric for the evaluation of collaborative intrusion detection systems in wireless sensor networks," in *2017 IEEE international conference on communications (ICC)*, pp. 1–6, IEEE, 2017. doi: 10.1109/ICC.2017.7997192.
- [13] J. Hesford, D. Cheng, A. Wan, L. Huynh, S. Kim, H. Kim, and J. B. Hong, "Expectations Versus Reality: Evaluating Intrusion Detection Systems in Practice," *arXiv preprint arXiv:2403.17458*, 2024. doi: 10.48550/arXiv.2403.17458.
- [14] B. K. Pandey, V. Saxena, A. Barve, A. K. Bhagat, R. Devi, and R. Gupta, "Intrusion Detection for Secure Social Internet of Things Based on Collaborative Edge Computing: A Generative Adversarial Network-Based Approach," *Soft Computing*, pp. 1–11, 2023. doi: 10.1109/TCSS.2021.3063538.
- [15] C. Pedroso, A. Batista, S. Brisio, G. Rodrigues, and A. Santos, "A Direct Collaborative Network Intrusion Detection System for IoT Networks Integration," in *Anais do XXXII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pp. 477–490, SBC, 2024. doi: 10.5753/sbr.2024.1354.
- [16] T. K. Authors, "KIND." <https://kind.sigs.k8s.io/>, May 2023.
- [17] I. Tigera, "About Calico." <https://docs.tigera.io/calico/latest/about/>, May 2023.
- [18] S. Madden, "Intel Lab Data." <https://db.csail.mit.edu/labdata/labdata.html>, June 2004.
- [19] Cisco, "What is Snort?." <https://www.snort.org/>, May 2023.
- [20] M. Goldstein, "BoNeSi - The DDos Botnet Simulator." <https://github.com/Markus-Go/bonesi>, May 2023.
- [21] Netscout, "Global Attack Duration Breakdown (2H 2023)." <https://www.netscout.com/threatreport/global-highlights/>, Jan. 2024.
- [22] D. Powers, "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011. doi: 10.9735/2229-3981.



Agnaldo de Souza Batista is a Ph.D. candidate at Federal University of Paraná (UFPR), Brazil. His main research interests are resilient communication, robust systems, data security, wireless networking, the Internet of Things (IoT), e-health, and management of critical events. He is a member of the Wireless and Advanced Networks lab (NR2) team, and of the Brazilian Computer Society (SBC).



Carlos Pedroso is a Ph.D. candidate at Federal University of Paraná (UFPR), Brazil. Master in Informatics from UFPR (2019). Has experience in Computer Science, with emphasis on Hardware, Computer networks, Wireless Sensor Networks and Internet of Things, acting mainly on the following topics: IoT and Security. Member of the Brazilian Computer Society (SBC) and IEEE Communication Society Communication (ComSoc).



Samuel Brísio de Jesus is undergraduate student in Computer Science at Federal University of Minas Gerais (UFMG), Brazil. His main research interests are resilient communication, and the Internet of Things (IoT).



Guilherme dos Santos Rodrigues is undergraduate student in Control Systems Engineering at Federal University of Minas Gerais (UFMG), Brazil. His main research interests are resilient communication, and the Internet of Things (IoT).



Aldri Santos is a full professor of the Department of Computer Science at the Federal University of Minas Gerais (UFMG). Aldri is working in the following research areas: network management, fault tolerance, security, data dissemination, wireless ad hoc networks and sensor networks. He is leader of the research group (Wireless and Advanced Networks). Aldri has also acted as reviewer for publications as IEEE ComMag, IEEE ComNet, ComCom, IEEE Communications Surveys and Tutorials, IEEE eTNSM, JNSM, Ad hoc Networks. Aldri has served as member of the technical committee of security information and IEEE Communication Society Communication (ComSoc).