

AN INNOVATIVE HYBRID MODEL FOR EFFECTIVE DDoS ATTACK DETECTION IN SOFTWARE DEFINED NETWORKS

Quang Truong Can, Tien Dat Nguyen, Minh Bao Pham, Thanh Tung Nguyen, Thi Hanh AnTran, Thi Thai Mai Dinh

Faculty of Telecommunication System, UET, VNU, 203-G2, 144 Xuan Thuy Street, Hanoi, Viet Nam

ABSTRACT

Software-Defined Networking (SDN) is a sophisticated network architecture that offers enhanced flexibility and streamlined management through a centralized controller. While these advantages allow SDNs to adapt to growing network demands, they also introduce potential security risks. Specifically, the centralized nature of SDN makes it vulnerable to network attacks, such as Distributed Denial of Service (DDoS) attacks, which can overwhelm network resources and cause widespread congestion. In this study, we propose a DDoS detection model that combines entropy-based features with Support Vector Machine (SVM) machine learning to create a hybrid approach. This model capitalizes on the strengths of both methods to improve detection accuracy. Our results, based on simulations and practical SDN implementation, show that our approach effectively and rapidly detects DDoS attacks with high precision. This paper addresses the challenge of enhancing the efficiency and accuracy of DDoS attack detection by providing a comprehensive dataset collected from both simulated and practical environments, thereby improving the detection system's performance in real-time situations.

KEYWORDS

SDN, DDoS attacks, network security, machine learning, statistical analysis method, entropy, dynamic entropy.

1. INTRODUCTION

In recent years, the global surge in network-connected devices has reached unprecedented levels, leading to an exponential increase in network traffic. It is projected that the number of these devices will reach 29.3 billion by 2023 [1], making it essential to enhance network performance continually. To address this challenge, Software-Defined Networking (SDN) has emerged as a significant advancement in the field. The introduction of the SDN architecture in 2011 marked a major development in computer networking, offering programmable features that significantly improve network monitoring and performance. Compared to traditional networks, SDN provides greater centralization and flexibility in system management. By separating packet forwarding in the data plane from routing functions in the control plane, as illustrated in Figure 1, SDN overcomes the limitations of conventional networks' static architecture, enabling more efficient packet routing.

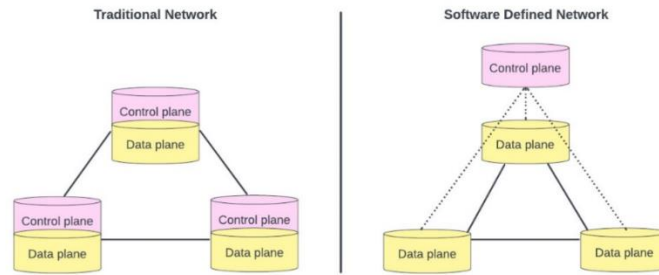


Figure 1. Traditional and SDN network architecture

The SDN architecture consists of three distinct layers: the application layer, the control layer, and the infrastructure layer [2]. The application layer plays a critical role in managing network applications, allowing administrators to effectively monitor and control the network by interacting with the control layer through the Northbound API. Operating at a higher level, the control layer is responsible for directing packets to the forwarding devices in the infrastructure layer for further processing. However, this architecture also introduces challenges and vulnerabilities, particularly concerning cyber-attacks. The most common and dangerous attack on this network architecture is a Distributed Denial of Service (DDoS) attack. In a DDoS attack, hackers flood the network with an overwhelming volume of virtual packets, leading to bandwidth overload and resource exhaustion, which can render the controller unable to manage user services. Given these risks, it is crucial to proactively detect and prevent DDoS attacks to ensure network security. Several approaches have been proposed to address this issue [3]. Most existing methods rely on public or outdated datasets and lack real-time detection capabilities. Additionally, they are primarily implemented in simulations rather than real-world environments. This paper provides a comprehensive review of research combining entropy and machine learning techniques, with a focus on the application of Support Vector Machine (SVM) mechanisms trained on our dataset. The objective is to introduce a high-performance hybrid model for effectively detecting DDoS attacks. Notably, most studies in this area are restricted to simulation environments, making real-world deployment a significant challenge. After proposing a hybrid model with superior performance metrics compared to existing methods, we implement it in a practical SDN environment to test and compare its results with previous approaches. This contribution improves the practicality and reliability of our findings, bringing them closer to real-world application.

2. RELATED WORKS

In recent years, various methods have been proposed to detect DDoS attacks in SDN. In papers [4] and [5], the authors identified a new type of DDoS attack in SDN that is difficult to detect using conventional machine learning techniques. They introduced a real-time DDoS detection system for SDN environments using Principal Component Analysis (PCA) to analyze network traffic data. This technique, applied to each subnet after dividing the network, successfully detected DDoS attacks targeting controllers or switches with a 95.24% success rate. However, this method's fixed threshold reduces its adaptability.

Several studies have addressed the challenge of detecting DDoS attacks using statistical methods, such as entropy-based detection. In [6], the authors used a real-time approach and sFlow-RT technology to calculate the entropy of network traffic within an SDN architecture. Another study [7] utilized a Modified Adaptive Threshold Algorithm (MATA) based on a traffic baseline, achieving a false alarm rate of 0.7% and an accuracy of 99%. However, this threshold is

susceptible to fluctuations when large volumes of legitimate traffic reach the controller, causing the moving baseline to produce inaccurate results for the detection system. Other papers, such as [8] and [9], also used entropy-based detection, achieving impressive accuracy rates of 99.73% and 95%, respectively. While these methods are easy to calculate and quickly detect DDoS attacks, they share similar limitations, such as susceptibility to spoofing and false positives when network behaviour changes [10-12]

In addition, several researchers have explored machine learning or deep learning methods, particularly the SVM model. In [13], an SVM-based algorithm classified six-tuple characteristic values to distinguish between normal and abnormal traffic, achieving a 95.24% average detection accuracy and a 1.26% false alarm rate. Another study [14] used feature entropy to train a nonlinear one-class SVM, achieving approximately 95% accuracy. Similarly, works in [15] and [16] using SVM methods achieved detection accuracies of 85% and 95.98%, respectively. A hybrid method combining Information Entropy and Deep Learning [17] achieved an accuracy detection rate of 98.98%. Other examples include [18, 19, and 20], where methods such as Extreme Learning Machines, Deep Neural Network (DNN), and SVM combined with Self Organizing Map (SOM) were proposed for real-time DDoS detection, with accuracy rates ranging from 96-98%.

In [21], an effective real-time DDoS detection method using a Deep Neural Network (DNN) within an SDN framework was introduced, boasting a 97.59% accuracy. This model used a four-layer DNN to process specifically chosen features, with key steps including data normalization and conversion of non-numeric values to numeric, enhancing the model's ability to differentiate between normal and DDoS traffic. Besides, paper [26] also made a comparison between different machine models. They applied ICIDS2017 and CICDDoS2019 datasets support vector machines (SVMs), K-nearest neighbours (KNNs), Decision Trees (DTs), Multiple Layer Perceptron (MLP), and Convolutional Neural Networks (CNNs), and compares their performance. Finally, the results showed that SVM achieved a good accuracy compared to the remains, it's considered to have the ability to correctly and simply detect DDoS in SDN architecture.

The researchers in [22] conducted a critical analysis of 12 recent DDoS detection methods using benchmark data, summarizing detection methods for application-layer DDoS attacks from January 2014 to November 2021. The authors in [23] proposed a feature selection-whale optimization algorithm-deep neural network (FSWOA-DNN) to mitigate DDoS attacks, achieving a 95.35% accuracy. Notably, to enhance the security of the proposed model, normal data was secured using homomorphic encryption and stored securely in the cloud.

Despite the significant progress in DDoS detection methods, many studies remain limited to simulated environments and lack the accuracy required for effective DDoS attack detection in practical settings. According to paper [27], although there was much research related to detect DDoS attacks in SDN using ML/DL, many problems still exist. Particularly, most of the papers use an offline dataset for verification and no deployment of automated real-time defense models, which neglects the crucial need to evaluate the performance of these models in real-time situations where DDoS attacks actually occur. To address these limitations, we propose a new and innovative hybrid approach that leverages the benefits of a dynamic entropy threshold and an advanced SVM model. This approach was developed after thoroughly reviewing the Existing literature on static entropy [6, 12], dynamic entropy [24], and SVM [14-16]. Our hybrid approach offers a more comprehensive and robust solution for detecting DDoS attacks in practical environments where the limitations of previous studies are evident. By combining the strengths of dynamic entropy thresholds and advanced SVM modeling, our approach aims to provide a more accurate and reliable method for detecting DDoS attacks at the practical and real-time environment.

The rest of this article is structured to present the proposed method and its evaluation. Section 3 outlines the hybrid approach, which combines a dynamic entropy threshold with an enhanced SVM model for DDoS detection, emphasizing its key features and advantages over traditional methods. Section 4 details the results from both simulations and practical environments, including performance metrics. Section 5 evaluates the system's accuracy, reliability, and efficiency. Section 6 compares the method's performance with existing approaches, providing an in-depth analysis of the findings. Finally, Section 7 summarizes the key conclusions and future directions, discussing the research's implications and suggesting areas for further study. Overall, this article provides a comprehensive evaluation of the hybrid method for detecting DDoS attacks, contributing significantly to network security.

3. PROPOSED DDoS DETECTION SYSTEM

This section introduces our novel methods for detecting DDoS attacks, which combine a dynamic threshold method based on entropy values and an enhanced SVM model inspired by [13]. The flowchart of the model is depicted in Figure 2. In the proposed hybrid model, the entropy module plays a crucial role in identifying potential network anomalies by measuring the variation in information within network traffic. Due to entropy's high sensitivity to shifts in network traffic, its effectiveness for anomaly detection can be hindered by frequent unexpected drops. These drops often occur when a large number of packets are transmitted to the same target host, potentially leading to false positives for DDoS attacks. To mitigate this risk, the model incorporates a warning condition that prompts further validation to ensure accurate detection.

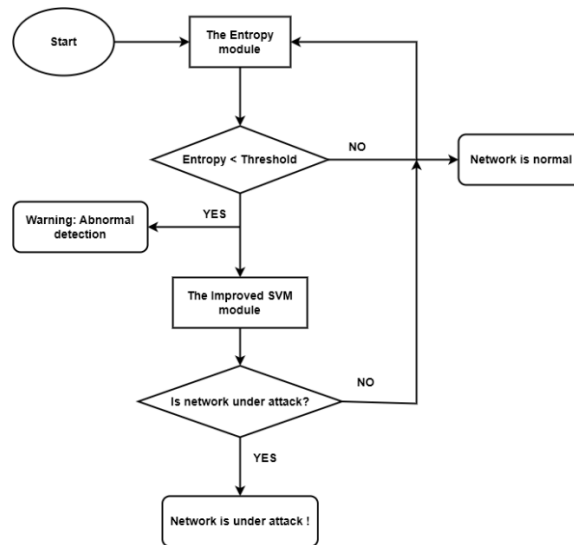


Figure 2. Hybrid Model

The enhanced SVM module then provides reliable and precise validation of the detected anomaly. Trained to distinguish between normal and malicious network traffic, the SVM model enables us to make informed decisions about the nature of the identified network anomaly. By integrating both the entropy module and the enhanced SVM module, our hybrid model offers a robust solution for detecting DDoS attacks while minimizing false alarms.

As shown in the flowchart in Figure 2, the proposed hybrid method for DDoS attack detection comprises two distinct stages, each addressing specific challenges and contributing to a more comprehensive solution:

First Stage, Entropy Module: In this stage, the entropy module continuously monitors network traffic and analyses information variation in the data. By examining data patterns, this module detects anomalies and triggers a warning message, acting as an early warning system for administrators and users. This early detection capability allows for proactive measures to be taken before the attack escalates, thereby enhancing network security.

Second Stage, Enhanced SVM Module: Once the entropy module detects a potential anomaly, the enhanced Support Vector Machine (SVM) model is activated to validate the anomaly. Trained on large datasets, the SVM model proficiently categorizes network traffic into two classes: normal and malicious. If the anomaly is confirmed as an attack, an alert is generated to notify about the ongoing threat. Conversely, if the traffic is deemed normal, the system reassures with a message stating "Network is normal." This second stage of validation ensures the hybrid approach's exceptional accuracy and reliability in detecting DDoS attacks in real-world environments.

The following subsections will present the proposed method in detail, providing a thorough explanation of its key components. The content is organized as follows: starting with the definition and formula used to calculate threshold values for the entropy module, followed by an outline of the process for training the SVM to enhance its detection capabilities, and concluding with a discussion on the concept of DDoS attacks and the various techniques employed for their detection.

3.1. Entropy Module

This section explains the formula and methodology used by the system to calculate the entropy value. The central controller continuously monitors the status of network traffic by collecting incoming packet header information, including the target IP addresses. The entropy value is calculated based on this data. A high entropy value suggests that the network traffic is evenly distributed across different destination addresses. Conversely, an unexpected surge in the number of packets directed to a single destination address can reduce the entropy value, signalling a potential network anomaly. To maintain normal network operations, it is essential to keep the entropy value within a specific threshold range.

Let's consider \mathbf{W} as a window containing N IP addresses, where n (with $n < N$) represents the number of distinct destination IP addresses in the incoming packet headers within this window. We can define the window \mathbf{W} as:

$$\mathbf{W} = [x_1, x_2, x_3, \dots, x_n] \quad (1)$$

In this context, $x_1, x_2, x_3, \dots, x_n$ represent the distinct destination IP addresses observed in the window \mathbf{W} . The entropy calculation will be based on the distribution of these distinct addresses, which helps in detecting any anomalies in network traffic.

The entropy value H is calculated using the following formula:

$$H = -\sum_{i=1}^N p_i \log(p_i) \quad (2)$$

where p_i represents the probability of an IP address in the window \mathbf{W} , calculated by:

$$p_i = \frac{x_i}{N} \quad (3)$$

Here, x_i is the count of IP address x_i in the window \mathbf{W} .

To determine the entropy thresholds, the confidence interval is used as follows:

$$ConfidenceInterval = \bar{X} \pm Z \cdot \frac{\sigma}{\sqrt{N}} \quad (4)$$

In this equation, \bar{X} is the sample mean, Z is the confidence coefficient, and σ is the sample standard deviation. A 95% confidence level is selected, so $Z = 1.9599$.

For a fixed entropy threshold, the value Δ is defined as:

$$\Delta = H_{n_{min}} - H_{a_{max}}$$

where $H_{n_{min}}$ is the normal average traffic entropy minus the confidence interval, and $H_{a_{max}}$ is the average entropy value during an attack plus the confidence interval. The fixed threshold is then determined as $H_{a_{max}} - \Delta$. If the entropy value falls below this static threshold, it indicates a potential ongoing attack.

Traditionally, a fixed threshold has been used to detect DDoS attacks, but this method lacks flexibility. The static threshold is set based on a fixed percentage of attack traffic relative to total traffic (e.g., 25%, 50%, or 75%), which doesn't adapt well to changing traffic patterns or new types of attacks.

To address this limitation, a dynamic threshold approach has been introduced. Instead of relying on a fixed threshold, the dynamic threshold continuously updates based on changes in the entropy value of incoming traffic. This allows for a more adaptive and responsive detection of network anomalies, including DDoS attacks.

The entropy value H_i of a window W is calculated using equation (2). The average entropy value

\bar{H}_t and the standard deviation σ_t are then computed as follows:

$$\bar{H}_t = \frac{1}{t} \sum_{i=1}^t H_i \quad (5)$$

$$\sigma_t = \frac{1}{t} \sum_{i=1}^t (H_i - \bar{H}_t)^2 \quad (6)$$

Using these parameters, the dynamic threshold value $T_{dynamic}$ is defined by the following formula:

$$T_{dynamic} = \bar{H}_t + C_d \cdot \sigma_t \quad (7)$$

In equation (7), \bar{H}_t and σ_t represent the average entropy value and standard deviation at time t , respectively. According to the normal distribution, 95% of entropy values fall within the range $\bar{H}_t \pm 2\sigma_t$. Values smaller than $\bar{H}_t \pm 2\sigma_t$ are less impactful on the overall results, making them a reliable basis for selecting C_d for this system. Based on experimental results, C_d is set to -2 , which corresponds to a 95% confidence interval in equation (7).

This completes the theoretical foundation for the proposed entropy-based method. These principles will be the basis for developing the SVM model, which will be detailed in the next section.

3.2. Support Vector Machine Module

In this subsection, we detail the Support Vector Machine (SVM) algorithm, a widely used model for classification tasks. The SVM algorithm is effective at transforming a dataset that is not linearly separable into a higher-dimensional space, allowing for linear separation of data points. This characteristic makes SVM particularly suitable for binary classification tasks, such as distinguishing between normal and attack states in this project. Characteristic values are parameters that represent the system's status and can vary significantly between normal and attack events. For detecting DDoS attacks, five key characteristics, referred to as a five-tuple, are used:

Speed of Source IP (SSIP): This measures the total number of incoming IP sources within a specified time period.

$$SSIP = \frac{Sum_{IPsrc}}{T} \quad (8)$$

In this formula, Sum_{IPsrc} represents the total number of IP sources recorded from the flowentries, and T is the time interval (e.g., 3 seconds). During a DDoS attack, the SSIP value increases due to a surge in packets from numerous IP addresses within a short timeframe.

Standard Deviation of Flow Packet (SDFP): This represents the standard deviation of the packet counts over a given period.

$$SDFP = \sqrt{\frac{1}{M} \sum_{i=1}^M (n_{packet_i} - Mean_{packets})^2} \quad (9)$$

Where

$$Mean_{packets} = \frac{1}{M} \sum_{j=1}^M number_packet_i$$

In this formula, M is the total number of new flow entries during time period T , $Mean_{packets}$ is the average number of incoming packets, and $number_{packet_i}$ is the packet count in the i^{th} flow entry. The SDFP decreases during a DDoS attack because the attacker floods the network with packets from different IP addresses, leading to an overloaded flow table in the switch and a reduction in SDFP.

Standard Deviation of Flow Bytes (SDFB): This measures the variation in the number of bytes in packets over a given time period.

$$SDFB = \sqrt{\frac{1}{M} \sum_{i=1}^M (n_{byte_i} - Mean_{bytes})^2} \quad (10)$$

Where

$$Mean_{bytes} = \frac{1}{N} \sum_{j=1}^N number_bytes_i$$

In this context, $Mean_{bytes}$ represents the average number of bytes during the time interval T , and $number_bytes_i$ is the byte count for each flow entry. During a DDoS attack, this value decreases significantly because attackers often reduce payload size to maximize the number of packets sent. Consequently, each packet contains only the header and trailer, leading to a sharp drop in packet size.

Speed of Flow Entries (SFE): This metric indicates the total number of flow entries added to the switch within a specific time period.

$$SFE = \frac{M}{T} \quad (11)$$

As discussed with the Standard Deviation of Flow Packet (SDFP), during an attack, the number of flow entries increases as the attacker sends numerous packets from various IP addresses. This surge in flow entries raises suspicion of an ongoing attack.

Number of Interactive Flow Entries Ratio (NIFE): This represents the ratio of interactive flows to the total number of flow entries.

$$NIFE = \frac{2*Pair_sum}{M} \quad (12)$$

In equation (12), *Pair_sum* denotes the number of interactive flow entries. Two flows are considered interactive if they meet the following conditions:

$$\begin{aligned} Src_{IP_i} &= Dst_{IP_j} \\ Src_{port_i} &= Dst_{port_j} \\ Src_{IP_j} &= Dst_{IP_i} \\ Dst_{port_j} &= Src_{port_i} \end{aligned}$$

Interactive flow entries occur when two nodes communicate with each other, requiring at least one interactive flow entry between them. During an attack, typically only one direction is targeted by fake IP sources towards the victim, leading to a lack of interactive flow entries. As a result, this value decreases during such events.

This section has introduced five key characteristics—based on source IP address, destination IP address, source port number, destination port number, and flow entries—that are used for detecting DDoS attacks. These characteristics form the basis of the SVM model proposed in [14]. To enhance this model, we have introduced a sixth feature: the entropy value discussed in subsection 3.1. The entropy value proves to be a highly sensitive metric, crucial for early detection of DDoS attacks. Its responsiveness to changes in network information enables rapid identification of abnormal behavior. During a DDoS attack, the entropy value sharply declines even with a small number of packets targeting the same destination within a short period. This heightened sensitivity highlights the significant role of the entropy value in our enhanced SVM model.

By integrating the entropy module with our SVM model, we have developed a hybrid model designed for optimal DDoS attack detection. The entropy module acts as an early detection mechanism, quickly identifying potential anomalies, while the enhanced SVM model provides the necessary validation to ensure accuracy. This combination offers a robust and effective solution, enabling our hybrid model to excel in detecting DDoS attacks.

4. IMPLEMENTATION AND RESULT

4.1. Preparation and Implementation

The simulation was conducted on an ASUS F570zd running Ubuntu 20.04. To create a simulated network environment, Mininet was selected as the network simulator, and a POX controller was employed. POX, a Python-based SDN controller, is more advanced than its predecessor, NOX. To evaluate the performance of our proposed DDoS attack detection model, we run our model script along with Mininet, which served as a virtual network testbed, capturing real-time traffic data that was then fed to our model for analysis. The simulated topology consisted of 64 hosts and 9 Open Switches, including 1 core Open Switch.

For practical testing, we used an Aruba Switch 2930F with Open Flow protocol, controlled by POX. The real network topology included 1 controller, 2 switches, and 8 hosts, as illustrated in Figure 3. Host h6 (IP address 10.10.0.6) was designated as the attacker, while host h4 (IP address 10.10.0.4) was the attack target. Hosts h3 (IP address 10.10.0.3) and h5 (IP address 10.10.0.5) generated normal traffic samples.

To facilitate communication between hosts via POX, the l2_learning module was utilized and modified. This module analyses incoming packets to extract IP addresses and adds flow entries to the flow table of an Open Flow switch. Scapy, a DDoS tool, was used to generate and flood TCP/UDP/ICMP packets with spoofed source IP addresses to simulate both normal and attack traffic. Normal traffic was generated at a rate of 0.1 seconds per packet, while attack traffic volumes were set at 25%, 50%, and 75% of the total traffic, with rates of 0.3, 0.1, and 0.033 seconds per packet, respectively.

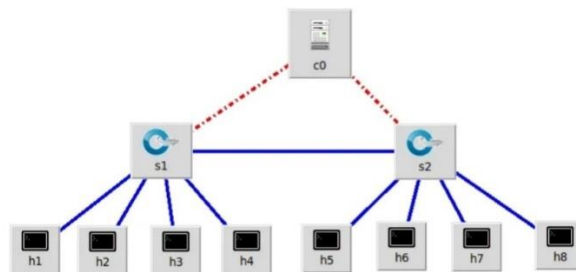


Figure 3. Topology in practical model

To implement the entropy method, we followed the processes and formulas outlined in Subsection 3.1 and integrated them into the l2_learning module.

The simulation and practical proceeded as follows:

Topology Setup: The Mininet network simulator (for simulation) and POX controller were used to establish the network topology. Normal traffic was generated from hosts h3 and h5, sending packets randomly to other hosts at a rate of 10 packets per second.

Detection Program: The detection program was initiated to generate normal traffic followed by attack traffic using Scapy. This allowed us to compare the characteristics of normal and attack traffic and analyse any differences.

SVM Model Prediction: The SVM models were used to predict the network status. The predicted results were compared with the actual network state to assess the accuracy of the SVM models in detecting DDoS attacks.

The new dataset collected was specifically tailored to fit the topology used in this study, ensuring that the simulation and practical results accurately reflect the performance of the SVM models. The scenarios described above were also run using the hybrid model.

4.2. Result

The results of the dynamic entropy method are illustrated in Figures 4 to 6. These figures depict key parameters of the method: entropy, average entropy, standard deviation, and dynamic threshold. The observations cover three scenarios with varying percentages of attack traffic: 25%, 50%, and 75%, as shown in Figures 4, 5, and 6, respectively. In these figures, simulation results are indicated by solid lines (Entropy-Sim, Ave-Sim, Std-Sim, and Threshold-Sim), while experimental results are shown by dashed lines (Entropy-Exp, Ave-Exp, Std-Exp, and Threshold-Exp).

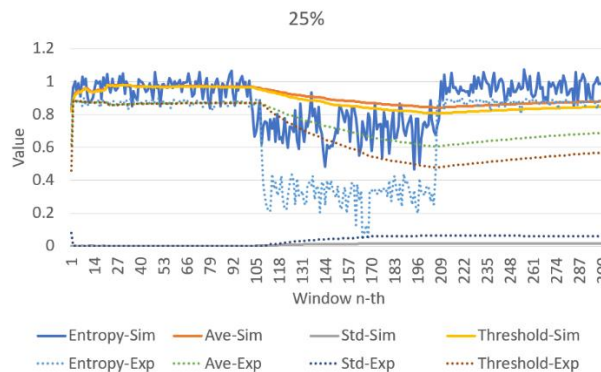


Figure 4. Dynamic threshold method with 25% attack

In Figure 4, for the normal traffic, the rate of packet generation is 0.1s/packet and for the attack traffic, that is 0.3s/packet, making the attack traffic account for 25% of the total traffic. In the first 100 windows, which is normal traffic generation stage, the entropy value fluctuates above and below the threshold and the mean. However, when attack traffic to host 10.0.0.1 on the 102nd window is conducted, the entropy value immediately drops below the threshold and the threshold value also decreases gradually. From the 102nd to 205th windows, the entropy remains below the threshold, indicating that an attack has taken place. From the 207th to 303rd windows, the attack is stopped, and normal traffic is restored. During this process, the entropy value increases above the threshold, indicating that the network has returned to its normal state. The experimental results also show a similar change in values, although the decrease in values is deeper due to the small size, which results in less randomness, of the actual network model.

In Figure 5 and Figure 6, when the attack traffic accounts for 50% and 75% of the total network traffic, the decrease in entropy, average, and threshold values is slightly deeper. In short, these changes are still obvious to indicate that the attack has been conducted from the 102nd to 205th windows. The values of these parameters vary overtime corresponding to the different scenarios. It is evident that the results are consistent with the formulas presented in the previous section.

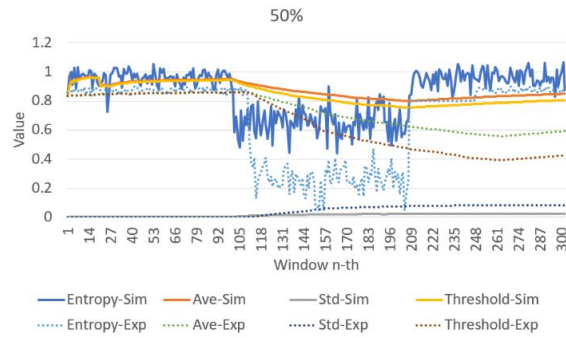


Figure 5. Dynamic threshold method with 50% attack

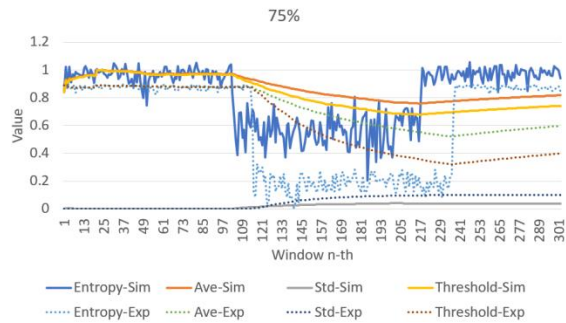


Figure 6. Dynamic threshold method with 75% attack

Figure. 7 to 12 shows the value variation of these features. The running scenario is similar to the dynamic entropy methods shown above. We can clearly see the change in values with and without an attack. In this case, attack traffic accounts for 50% of the total network traffic and SVM is used to perform attack traffic generation.

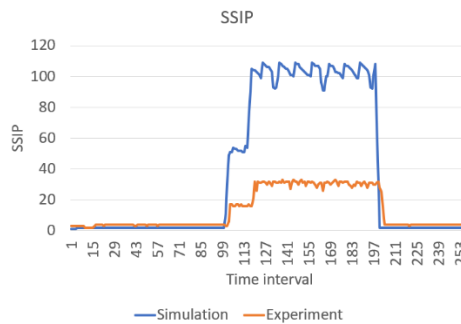


Figure 7. SSIP value

Figure 7 illustrates that the SSIP value shows a substantial increase during an attack, which occurs between the 99th and 197th seconds. Under normal conditions, approximately 5 to 6 new packets are generated every 3 seconds. During an attack, however, the packet volume sharply rises to 115 packets per 3 seconds in the experimental results and 30 packets per 3 seconds in the simulation results. Despite the differences between the simulation and real-world models, this value clearly distinguishes between attack and non-attack scenarios.

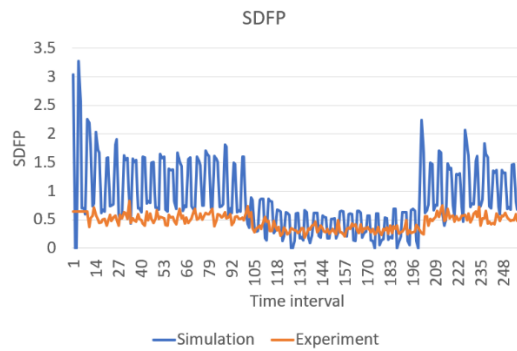


Figure 8. SDFP value

The SDFP value is depicted in Figure. 8. This figure clearly illustrates the difference in the number of packets generated during normal and attack scenarios. In the experimental results, the standard deviation of the number of packets generated is around 1.4 in the normal scenario and drops to around 0.4 when an attack occurs. The values in the simulation follow a similar pattern but are lower in magnitude.

Figure 9 displays the standard deviation of flow bytes (SDFB). The experiment reveals that under normal conditions, the standard deviation averages around 45. However, during an attack, this value decreases to approximately 10. The simulation results align with these observations, showing similar behavioural patterns. This drop in standard deviation during an attack highlights a significant deviation from normal network behaviour, serving as a key indicator of unusual activity in the network.

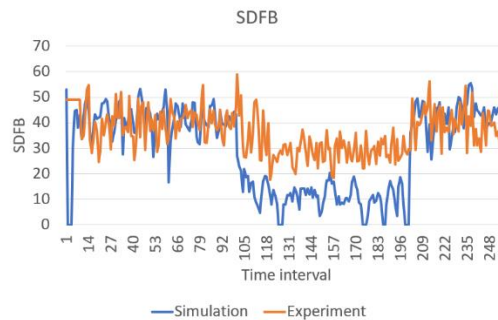


Figure 9. SDFB value

Figure 10 illustrates the NIFE value, which measures the interaction between streams. Under normal conditions, the average NIFE value is approximately 0.9 in simulations and 0.7 in experiments. However, during an attack, the NIFE value drops sharply to nearly 0 in both simulation and experimental settings. This drastic decrease indicates a significant reduction in two-way interaction between the hosts, reflecting the victim's inability to respond promptly due to the ongoing attack.

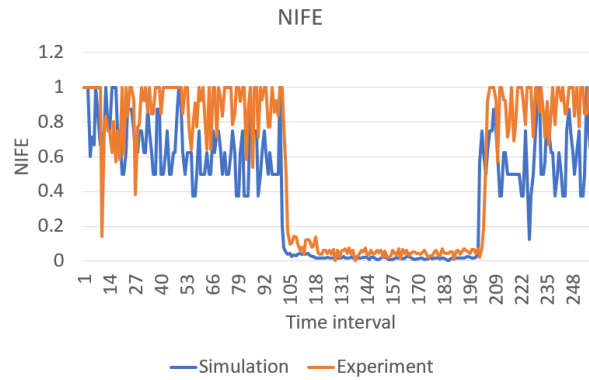


Figure 10. NIFE value

The SFE value is illustrated in Figure 11, which represents the rate at which flow entries are created within each 3-second interval. During a DDoS attack, attackers frequently use numerous unique IP addresses, causing the controller to add new entries to the flow table. Consequently, a higher number of new IP addresses leads to an increased number of flow entries. This value parallels the SSIP value and effectively captures the impact of a DDoS attack on the network's flow entries.

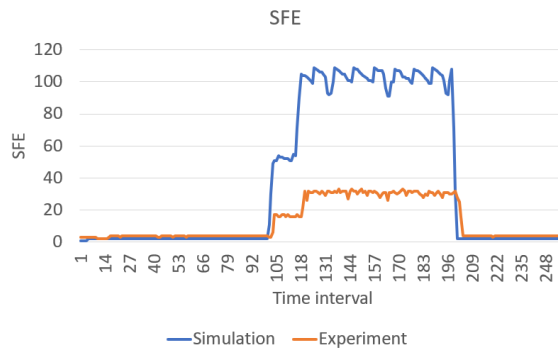


Figure 11. SFE value

The entropy value in Figure. 12 as the sixth feature of the SVM is similarly modulated as in the entropy-based statistical models, which is notably decreased during the attack stage.

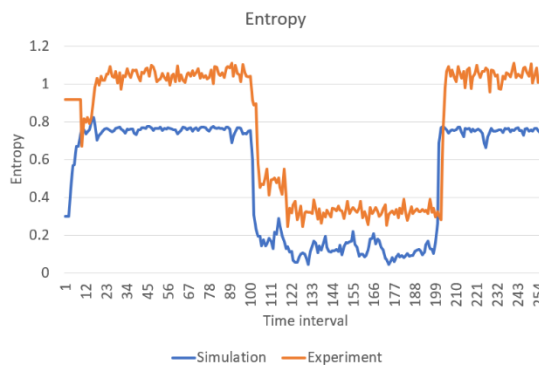


Figure 12. Entropy value

5. EVALUATED METRICS

All performance metrics are defined by [25]. Statistical results in Table I show that accuracy, precision, and recall improve with the complexity of the methods.

The fixed threshold entropy method demonstrates relatively low accuracy, achieving 94.59% in simulation and 92.98% in the experiment. This is primarily due to the static threshold's inability to adapt to dynamic changes in the network's state. While the method is simple and provides a quick detection time of approximately 5.4 seconds, its accuracy is limited because entropy's sensitivity to information does not allow for higher accuracy levels.

In contrast, the dynamic threshold entropy method performs significantly better in simulation, with an accuracy exceeding 97%. However, in the experiment, accuracy is just over 90%, largely due to the method's simplicity, which results in a lack of randomness and difficulty in precisely determining the network's state. Despite this, the dynamic threshold method offers ease of implementation, straightforward operations, and a swift detection time of about 3.4 seconds, capitalizing on entropy's characteristics. It also provides high flexibility for adapting to various network scenarios.

Table 1. PERFORMANCE METRICS FOR THE METHODS

Methods (Simulation/ Experiment)	TP (%)	FP (%)	TN (%)	FN (%)	Accuracy (%)	Precision (%)	Recall (%)	Response Time (s)	Warning (%)
Fixed threshold entropy	93.3	6.7	96	4	94.6	93.3	96	5.4	-
	91.7	8.3	94.5	5.5	93	91.7	92.84	5.6	-
Dynamic threshold entropy	96.4	3.6	98.1	1.2	98	96.4	97.56	3.5	-
	92.2	7.8	94.5	5.1	90	92.2	93.99	3.3	-
SVM	97.5	2.5	98.3	1.7	97.9	97.5	98.3	7.7	-
	96.6	3.4	97.8	2.2	97.2	96.6	97.78	7.6	-
EnhancedSVM	99.2	0.9	98.6	1.4	98.9	99.2	99.15	7.1	-
	99	1	99.3	0.7	99.2	99	99.01	7.3	-
Proposed Hybrid	99.7	0.3	99.1	0.9	99.4	99.1	99.7	3.8	3.8
	99.8	0.2	99.5	0.5	99.8	99.5	99.65	3.9	4.2

The SVM models outperform both entropy-based statistical methods, with accuracies exceeding 97%, and the enhanced SVM method achieving up to 99%. This demonstrates the superiority of machine learning approaches. However, the average response time for these methods is 7.2 seconds, owing to the 3-second information capture period needed to gather sufficient data for calculations. Consequently, SVMs require more than twice this period to detect an attack.

The hybrid model marks a significant advancement in intrusion detection. It features a rapid response time of 4 seconds and achieves up to 99% accuracy, highlighting its effectiveness. The alert-to-outcome ratio in the simulation and experiment is 3.8% and 4.2%, respectively, indicating the false decision rate of the entropy module. Nevertheless, the strength of the SVM module enhances the hybrid model by reconfirming the entropy module’s decisions, improving the likelihood of a correct outcome. The entropy module’s sensitivity to anomalous information variance, combined with the SVM module’s accuracy, reinforces the overall effectiveness of the hybrid model in detecting DDoS attacks.

6. COMPARING THE PROPOSED METHOD WITH OTHER WORK

Figure 13 presents the statistical results for accuracy, precision, and recall of the various methods. Figure 14 illustrates the response time for each method to successfully detect DDoS attacks. The static threshold method exhibited lower accuracy, with an average of approximately 93% and a detection time of 5.5 seconds. In contrast, the dynamic threshold method showed marginal improvements, achieving about 4% higher accuracy and a response time that was 2 seconds faster compared to the static threshold method.

Entropy-based methods demonstrated less accuracy than SVM-based methods, with SVM achieving an average accuracy of 98%. However, SVM had a slower response time of around 7 seconds, compared to the 3.4 seconds and 5 seconds of the two entropy-based methods. The high accuracy of SVM is attributed to the significant differences in feature values between normal and attack scenarios, making it easier for SVM to delineate the two data sets. This performance analysis underscores the SVM model’s high precision in classification. Adding entropy as a sixth feature to the enhanced SVM model further enhanced accuracy in challenging classification cases. For instance, as shown in Figure 8, while the number of packets or bytes during an attack might be similar to normal traffic, the number of destination IP addresses may be abnormal. In such cases, the entropy value decreases quickly, aiding in earlier attack detection.

The hybrid model outperformed all other methods, achieving up to 4% higher accuracy than the static threshold entropy method and a faster response time of around 3 seconds. The integration of entropy with SVM in the hybrid model combines the strengths of both methods, resulting in exceptional performance and outstanding results.



Figure 13. Performance metrics

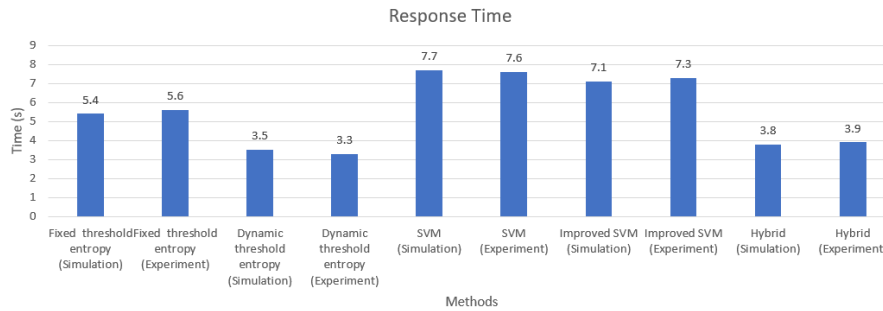


Figure 14. Response time

7. CONCLUSION AND FUTURE WORK

The primary objective of this research was to address the issue of DDoS attacks in software-defined networks. We conducted a comprehensive investigation evaluating several models: the static entropy method, the proposed dynamic entropy threshold solution, the enhanced SVM classification model, and a hybrid model combining dynamic entropy with the enhanced SVM model. Both simulation and experimental environments were used for this evaluation.

Our findings reveal that the proposed models are highly effective in accurately detecting DDoS attacks, with the hybrid model achieving accuracy levels of up to 99% while maintaining fast response times. These results are significant for the cyber security field, providing a hardware-compatible solution that integrates seamlessly into software-defined networks.

Future research could explore integrating additional machine learning and deep learning methods to improve the classification of diverse attack types, such as slow DDoS and Man-in-the-Middle attacks. Investigating mitigation strategies like load balancing and monitoring port traffic to identify attackers based on specific thresholds would also be beneficial. A deeper understanding of various DDoS attack types, combined with the exploration of different algorithms and SDN architectures, will help identify optimal solutions and advance the field of network security.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

ACKNOWLEDGEMENTS

Quang Truong Can was funded by the Master, PhD Scholarship Programme of Vin group Innovation Foundation (VINIF), code VINIF.2024.ThS.13.

REFERENCES

- [1] Cisco, "Cisco Annual Internet Report (2018–2023) White Paper," 10-Mar-2020. [Online]. Available: <https://www.cisco.com>.
- [2] SDX Central, "Understanding the SDN Architecture." [Online]. Available: <https://www.sdxcentral.com/resources/sdn/insidesdn-architecture/>. [Accessed Dec. 2022].
- [3] Ali, T. E., Chong, Y., & Manickam, S. (2022). Machine Learning Techniques to Detect a DDoS Attack in SDN: A Systematic Review. *Applied Sciences*, 13(5), 3183. <https://doi.org/10.3390/app13053183>

- [4] Y. Cui, L. Yan, S. Li, H. Xing, W. Pan, J. Zhu, and X. Zheng, "SDAnti-DDoS: Fast and Efficient DDoS Defence in Software-Defined Networks," *Journal of Network and Computer Applications*, vol. 65, pp. 65-79, 2016.
- [5] S. Salaria, S. Arora, N. Goyal, P. Goyal, and S. Sharma, "Implementation and Analysis of an Improved PCA Technique for DDoS Detection," in *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, pp. 280-285, 2020.
- [6] Abdul Adhim, Satoshi Okada, and Takuho Mitsunaga, "SDN-Based Detection Method Against DoS/DDoS Attacks in an IoT Environment," in *Symposium on Cryptography and Information Security*, Osaka, Japan & Online, Jan. 18–21, 2022.
- [7] Haymarn Oo, Nan, Risdianto, Aris Cahyadi, Teck, Chaw Ling, and Maw, Aung Htein. "Flooding Attack Detection and Mitigation in SDN with Modified Adaptive Threshold Algorithm." *International Journal of Computer Networks & Communications*, vol. 12, no. 3, May 2020, pp. 75–95. Academy and Industry Research Collaboration Center (AIRCC). DOI: 10.5121/ijcnc.2020.12305.
- [8] Ranyelson N. Carvalho, Jacir L. Bordim, and Eduardo A. P. Alchier, "Entropy-Based DoS Attack Identification in SDN," in *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*.
- [9] Ming Xuanyuan, Visham Ramsurrun, and Amar Seeam, "Detection and Mitigation of DDoS Attacks Using Conditional Entropy in Software-Defined Networking," in *2019 11th International Conference on Advanced Computing (ICoAC)*.
- [10] i. Özçelik and R.R. Brooks, "Deceiving Entropy-Based DoS Detection," *Computers & Security*, vol. 48, pp. 234-245, 2014. doi:10.1016/j.cose.2014.10.013.
- [11] J. David and C. Thomas, "Efficient DDoS Flood Attack Detection Using Dynamic Thresholding on Flow-Based Network Traffic," *Computers & Security*, vol. 82, pp. 284-295, 2019. doi:10.1016/j.cose.2019.01.002.
- [12] Vitali D, Villani A, Spognardi A, Battistoni R, Mancini LV, "DDoS Detection with Information Theory Metrics and Netflows – A Real Case," in *International Conference on Security and Cryptography*, 2012, pp. 172-181. doi:10.5220/0004064501720181.
- [13] Jin Ye, Xiangyang Cheng, Jian Zhu, Luting Feng, and Ling Song, "A DDoS Attack Detection Method Based on SVM in Software Defined Network," *Security and Communication Networks*, vol. 2018, Article ID 9804061, 2018.
- [14] Dong Li, Chang Yu, Qizhao Zhou, and Junqing Yu, "Using SVM to Detect DDoS Attack in SDN Network," in *IOP Conference Series: Materials Science and Engineering*, 2018.
- [15] Pynbianglut Hadem, Dilip Kumar Saikia, Soumen Moulik, "An SDN-Based Intrusion Detection System Using SVM with Selective Logging for IP Traceback," *Computer Networks*, vol. 191, 108015, 2021.
- [16] K. Muthamil Sudar, M. Beulah, P. Deepalakshmi, P. Nagaraj, and P. Chinnasamy, "Detection of Distributed Denial of Service Attacks in SDN Using Machine Learning Techniques," in *2021 International Conference on Computer Communication and Informatics (ICCCI -2021)*, Jan. 27–29, 2021, Coimbatore, India.
- [17] Lu Wang and Ying Liu, "A DDoS Attack Detection Method Based on Information Entropy and Deep Learning in SDN," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC 2020)*.
- [18] C. Gong, D. Yu, L. Zhao, X. Li, and X. Li, "An Intelligent Trust Model for Hybrid DDoS Detection in Software Defined Networks," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 16, e5264, 2019. doi:10.1002/cpe.5264.
- [19] Auther Makuvaza, Dharm Singh Jat, and Attlee M. Gamundani, "Deep Neural Network (DNN) Solution for Real-Time Detection of Distributed Denial of Service (DDoS) Attacks in Software Defined Networks (SDNs)," *SN Computer Science*, vol. 2, Article 107, 2021. doi:10.1007/s42979-021-00467-1.
- [20] T. V. Phan, N. K. Bao, and M. Park, "A Novel Hybrid Flow-Based Handler with DDoS Attacks in Software-Defined Networking," in *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCCom/IoP/SmartWorld)*, doi:10.1109/uiccate-scalcom-cbdc-com-iopsmartworld.2016.0069.

- [21] Deep Neural Network (DNN) Solution for Real-Time Detection of Distributed Denial of Service (DDoS) Attacks in Software Defined Networks (SDNs),” SN Computer Science, vol. 2, pp. 1-10.
- [22] Detection of DDoS Attack Using IDS Mechanism: A Review,” in 2022 1st International Conference on Informatics (ICI), pp. 36-46, IEEE
- [23] .a.Agarwal, M. Khari, and R. Singh, “Detection of DDoS Attack Using Deep Learning Model in Cloud Storage Application,” Wireless Personal Communications. doi:10.1007/s11277-021-08271-z.
- [24] N. Do Van, L. D. Huy, C. Q. Truong, B. T. Ninh, and D. T. Thai Mai, “Applying Dynamic Threshold in SDN to Detect DDoS Attacks,” in 2022 International Conference on Advanced Technologies for Communications (ATC), Ha Noi, Vietnam, pp. 344-349, 2022. doi:10.1109/ATC55345.2022.9943031.
- [25] Rainio, O., Teuho, J. & Klén, R. Evaluation metrics and statistical tests for machine learning. Sci Rep 14, 6086 (2024). <https://doi.org/10.1038/s41598-024-56706-x>[30] Ali, Tariq Emad, Yung-Wey Chong, and Selvakumar Manickam. 2023. "Comparison of ML/DL Approaches for Detecting DDoS Attacks in SDN" Applied Sciences 13, no. 5: 3033. <https://doi.org/10.3390/app13053033>
- [26] Ali, Tariq Emad, Yung-Wey Chong, and Selvakumar Manickam. 2023. "Comparison of ML/DL Approaches for Detecting DDoS Attacks in SDN" Applied Sciences 13, no. 5: 3033. <https://doi.org/10.3390/app13053033>
- [27] Ali, Tariq Emad, Yung-Wey Chong, and Selvakumar Manickam. 2023. "Machine Learning Techniques to Detect a DDoS Attack in SDN: A Systematic Review" Applied Sciences 13, no. 5: 3183. <https://doi.org/10.3390/app13053183>

AUTHORS

Can Quang Truong graduated from the University of Engineering and Technology, Vietnam National University in 2023. He is a member of Telecommunication Systems Laboratory in University of Engineering and Technology. His research interests include Software Defined Network, Cyber Security and Network Automation.



Nguyen Tien Dat graduated from the University of Engineering and Technology, Vietnam National University in 2023. He is a member of Telecommunication Systems Laboratory in University of Engineering and Technology. His research interests include Software Defined Network, Cyber Security and Network Monitoring.



Pham Minh Bao graduated from the University of Engineering and Technology, Vietnam National University in 2023. He is a member of Telecommunication Systems Laboratory in University of Engineering and Technology. His research interests include Software Defined Network, Cyber Security and Cloud Computing.



Nguyen Thanh Tung graduated from the University of Engineering and Technology, Vietnam National University in 2023. He is a member of Telecommunication Systems Laboratory in University of Engineering and Technology. His research interests include Software Defined Network, Cyber Security and System Designing.



Tran Thi Hanh An graduated from the University of Engineering and Technology, Vietnam National University in 2024. She is a member of Telecommunication Systems Laboratory in University of Engineering and Technology. Her research interests include Software Defined Network ,Network Infrastructure and Network Security



Assoc. Prof. Dinh Thi Thai Mai received the Engineer of Electronics and Telecommunications from the Post and Telecommunications Institute of Technology in 2006, the M.Sc. degree from the University of Paris Sud 11, France, in 2008, and the Ph.D. degree from the VNU University of Engineering and Technology, Hanoi, Vietnam, in 2017. She is the Head of the Department of Telecommunications Systems, Faculty of Electronics and Telecommunications, VNU University of Engineering and Technology. Currently, her research interests include 5G/6G mobile networks, wireless communications, localization techniques and Security in SDN.

