Migration Letters

Volume: 20, No: S13(2023), pp. 407-419

ISSN: 1741-8984 (Print) ISSN: 1741-8992 (Online)
www.migrationletters.com

Detection and Mitigation of DDOS Attack in SDN Environment Using Hybrid CNN-LSTM

Dhanya M Rajan[1*] and Dr.D. John Aravindhar[2]

Abstract

DDoS attacks, powered by botnets to flood network resources, pose a significant threat to traditional network setups. Software-Defined Networking (SDN) boosts network adaptability and programmability by separating the control and data planes. However, the centralized control in SDN can be a vulnerability, allowing attackers to exploit security flaws and launch DDoS attacks. These attacks overwhelm network controllers and switches, consuming bandwidth and server resources, and disrupting regular user access. In response to the threat, we've implemented an online SDN defence system designed to detect and counter such attacks. This system includes modules for both spotting anomalies and handling them. The anomaly detection model combines Convolutional Neural Network and Long Short-Term Memory (CNN-LSTM) techniques to effectively spot irregular traffic patterns. For mitigation, the model identifies abnormal traffic by implementing flow rule orders from the controller and traces back to the attacker via IP tracing. To measure our approach's effectiveness, we used various evaluation metrics like Accuracy, F-measure, Precision, Recall, ROC Curve, and Precision-Recall Curve. Our methodology displayed impressive results, achieving a 99.83% accuracy in multiclass classification and 99.17% accuracy in binary classification. In comparison with existing DDoS detection systems, our AI-driven mitigation techniques demonstrated their superiority. Overall, our research aims to streamline the detection and mitigation of DDoS attacks.

Keywords: *Convolutional Neural Network, Long Short-Term Memory, Software-Defined Networking, Distributed Denial of Service, Botnets.*

Introduction

The ever-changing network attack landscape poses escalating cyber threats as the number of connected devices surpassed 26 billion in 2019. This escalation is fueled by the deep reliance of government, military, and commercial entities on the Internet. A significant threat emerges from an orchestrated and widespread assault known as a Distributed Denial of Service (DDoS) attack. In this form of attack, the perpetrator employs multiple compromised computers, acting as proxies, to simultaneously initiate a barrage of Denial-of-Service (DoS) attacks on the intended target. Consequently, the targeted system becomes inundated, exhausting its resources or leading to a total failure, making it unable to fulfil the standard requirements of its users. Since its inception in 1999, DDoS attacks have evolved into one of the most prevalent and threatening online risks. According to a Radware survey, these attacks currently top the list of concerns for network security among internet-related enterprises.[1]

[1*] Research Scholar, Department of CSE, Hindustan Institute of Technology and Science, Padur, Chennai. dhanyamrajan83@gmail.com

[2] Professor & HEAD- MIDARC Department of CSE, Hindustan Institute of Technology and Science, Padur, Chennai. jaravindhar@hindustanuniv.ac.in

Software Defined Networking (SDN) employs a revolutionary network structure to attain centralized control over hardware. This method divides the data-forwarding duties of network devices from the decision-making control function. In SDN's control plane, the control logic, potentially consisting of one or more controllers, assumes complete responsibility. In SDN, controllers oversee all forwarding devices and determine routing, offering a level of sharing, adaptability, and intricate control over switches not typically found in standard IP networks. SDN employs a central controller to manage the network state, granting it a comprehensive global view. Communication between the controller and switches primarily utilizes the open-source OpenFlow protocol, enabling direct modification of flow rules within any switch as needed. This streamlined network architecture increases the flexibility of network control and makes SDN extensively applicable to wireless LANs, cloud computing, and the networks of cloud data centers. [2]

SDN's centralized structure exposes it to DDoS attacks, commonly executed in traditional networks by botnets controlled by attackers. These attacks flood targeted servers with an overwhelming volume of application requests from numerous controlled endpoints, causing a depletion of the servers' bandwidth and processing capacity. In the area of SDN, researchers have identified new strategies for DDoS attacks. These include assaults on controllers through packet flooding, targeting southbound channels with methods like CrossPath, and overwhelming switches with techniques such as flow table overflow etc. The performance of SDN will be significantly impacted by both conventional DDoS assaults and innovative DDoS attacks targeted towards SDN architecture. For DDoS attacks in SDNs, numerous researchers have put out a variety of anomaly detection techniques in recent years. [3]

Utilizing centralized Intrusion Detection Systems (IDS) sets apart these techniques from conventional network anomaly detection methods, as it diminishes the cost associated with adding extra detection devices while enhancing overall detection effectiveness. This prevents interruptions to regular application services on targeted servers. Additionally, new DDoS attack strategies for SDN have been identified by researchers. These include assaults on controllers through packet floods, switches via CrossPath attacks, and attacks against southbound channels. Traditional DDoS assaults as well as new DDoS attacks that target SDN design will negatively impact SDN performance. [4]

Attack detection, load balancing, traffic filtering, and traffic analysis are the four stages that are frequently used to categorize traditional DDoS detecting and mitigation techniques. (1) DDoS traffic must be distinguished from regular traffic during attack detection. The majority of common detection techniques rely on machine learning and message statistics, the accuracy of which must be guaranteed in real-time. (2) Load balancing offers a momentary resistance to abrupt abnormal traffic within the load balancing module's tolerance range, relieving the afflicted target's storage and compute pressure by rerouting or traffic migration. (3) In order to increase the identification accuracy and guarantee that normal traffic may be transmitted normally by network devices, traffic filtering eliminates DDoS attack traffic by identifying abnormal traffic characteristics. (4) By examining the attack traffic data gathered, traffic analysis seeks to pinpoint the attacker's intended behaviour and track the assault's origin. [5]

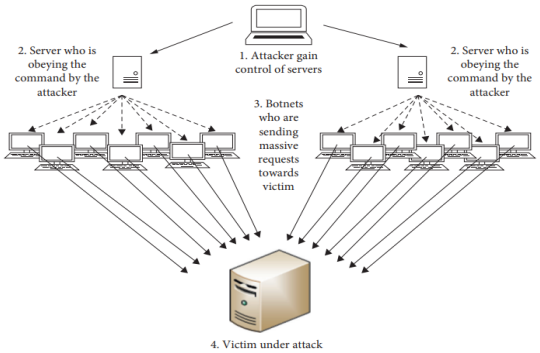

Figure 1: DDoS attack scenario.

Illustrated in Figure 1, the objective of the DDoS attack is to inundate the server with a substantial volume of requests, with the intention of overwhelming it. This will cause the impacted server to become unavailable, thereby preventing authorized users from accessing its services. Zombies, sometimes known as botnets, are ad hoc devices that may be remotely managed by a hacker using software. Massive queries are made to the server via botnets. Many gadgets, including IoT devices, can be utilized as botnets, and the attacker typically has complete control over them. [6]

Agent handler architecture is a type of DDoS assault. In this paradigm, the master interacts with the other DDoS attack system components. The handlers are used to communicate with the agents via software applications known as masters on the Internet. The DDoS assault is carried out using agent software that is present on botnets, or so-called compromised machines. In order to locate the active agents and plan an attack, the attacker can connect with a large number of handlers. Presently, DDoS assault anomaly detection techniques are categorized into three main groups: information statistics-based methods, ML-based methods, and deep learning-based methods.[7]

Related Works

Brindha et al. [8] presented an innovative four-step intrusion detection system tailored for IoT. In the first stage, data normalization is conducted as a component of the preprocessing phase. Later on, the system fetches "statistical features of a higher order" and employs an improved Recursive Feature Elimination (RFE) method, integrating enhanced technical indicator-based features of a second order (such as ATR, CMF, CTI, and upgraded EMA). These features include kurtosis, variance, and skewness. Aladdine et al. introduced a novel architecture comprising two components: one for detecting and mitigating DoS/DDoS attacks. This architecture stands out for its ability to accurately identify the specific attack and packet types used, enabling precise, fine-grained detection within the detection component.

Jin Wang et al. [10] have established and implemented a defence system in the domain of Software-Defined Networking (SDN) aimed at identifying and mitigating online attacks. This system comprises two primary components: an anomaly detection module and a mitigation module. The lightweight hybrid deep learning approach utilized by the anomaly detection module is the Convolutional Neural Network and Extreme Learning Machine (CNN-ELM). This approach aims to identify irregularities in network traffic. In a connected initiative, Deepak and his team [11] have presented a fresh approach to identify DDoS assaults in 5G networks. This approach involves two key phases: extracting features and detecting attacks. Their detection process utilizes a blend of classifiers using Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM). To enhance detection accuracy, the RNN's weight is refined using the Opposition Learning-based Seagull Optimization Algorithm (OLSOA) model.

Due to the superior performance of deep learning neural networks in contrast to other machine learning methods and intrusion detection techniques, Nisanthan et al.'s [12] research focused on their utilization for categorizing and mitigating DDoS attacks specifically within the HTTP layer of a server. On 177 samples, the suggested classifier by Omerah et al. [13] was evaluated. To simulate the DDoS attack and precisely identify the various DDoS attack types in the network, tools like Mininet and Wireshark were used.

Badre et al.'s [14] research defines DDoS attacks as the actions of malicious users who initiate assaults on the 5G Core Network. Based on the ML module created, they demonstrated that the prototype is capable of identifying these attacks, automatically producing a sinkhole-type slice using a minimal amount of physical resources, and isolating the malicious users within this slice to counteract the attackers' behaviour. Sowmya et al. [15] developed SmartDefense, a distributed platform designed to detect and counter DDoS attacks by identifying and addressing those originating from and in close proximity to their source. The platform utilized edge computing technologies.

6 base classifiers (2 SVMs, 2 Random forests, and 2 Gradient Boosted Machines) were used in the ensemble proposed by Aastha et al. [16] and were distinguished by hyperparameter values. A unique hybrid metaheuristic optimization technique determines the best set of weights (BHO). As per the algorithms proposed by Theyazhn et al. [17], the RF method exhibited a 98% accuracy in binary classification, and the SVM model attained a 97.54% accuracy in multiclassification. Additionally, the analysis included assessing prediction discrepancies between input data and values predicted by different machine and deep learning methodologies. This evaluation involved statistical analyses like Mean Square Error (MSE), Pearson correlation coefficient (R), and Root Mean Square Error (RMSE).

To identify DoS attacks within IoT networks, Kuburat et al. [18] introduced an Intrusion Detection System (IDS) employing a refined long short-term memory deep learning technique. They evaluated the model's performance using benchmark datasets, CICIDS-2017 and NSL-KDS.

In the context of safeguarding networks from DDoS attacks, Nagesh et al. [19] recommended the utilization of Grey Wolf Optimization (GWO) methods in conjunction with a Long Short-Term Memory Recurrent Neural Network (LSTM-RNN). Furthermore, Sungwoong et al. [20] suggested a collaborative framework for detecting DDoS attacks from the source side, employing LSTM (Long Short-Term Memory). This framework provides a comprehensive perspective on identifying attacks within a globally interconnected network.

Sinthuja et al. [21] proposed a novel approach employing the BFOFA-LSTM hybrid technique to optimize LSTM parameters. Contrasting with fully connected GANs, Aranya et al. [22] advocate an LSTM encoder-decoder GAN that significantly improves the capture of power system temporal dynamics. This enhancement yields heightened accuracy and faster responsiveness for both objectives.

Bibhuti et al[23] suggested a detection framework for attacks, employing a two-stage ensemble of classifiers that includes the Gated Recurrent Unit (GRU), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), and Optimized Deep Belief Network (DBN). Meanwhile, Surya Pavan and team [24] presented an intelligent approach for detecting DDoS attacks, known as the DI-ADS, specifically designed for fog-based IoT applications. The effectiveness of this method relies mainly on the integration of a Deep Learning Model (DLM) into the system, which is utilized to detect DDoS attacks on the network. Meenakshi et al. [25] suggested utilizing four frequently utilized digital libraries (IEEE, ACM, ScienceDirect, Springer) in addition to a scholarly search engine (Google Scholar) to explore the most recent literature.
The following is a summary of the contributions made by the proposed work.

- ❖ This article distinguishes itself through its introduction of a security system utilizing SDN, designed to rapidly identify and stop DDoS attacks in real-time across different online platforms. The CNN-LSTM intrusion detection method enhances accuracy significantly by incorporating the IP traceback mechanism within the SDN framework.
- ❖ An inventory of atypical traffic is compiled to effectively pinpoint the origins of attacks. The conservation of memory space occurs through the selective logging of unusual packets flagged by the IDS.
- ❖ The blacklist aims to streamline IP traceback by leveraging data from the packet/timestamp flow for greater efficiency.
- ❖ Anomalies flows are totally deleted from the source by issuing to effectively mitigate DDoS attacks.

The rest of this document is structured in the following manner: Section 2 offers a concise overview of related works, while Section 3 delineates the architecture and implementation specifics of our proposed system. Section 4 delves into the discussion of experimental results, and Section 5 presents conclusions along with an outline of future work.

Proposed Method

Both geographical and temporal data can be expressed using the CNN-LSTM approach. An incursion assault uses a variety of attack techniques, depending on the target or point of attack because it happens in real-time. A CNN is used for feature extraction, utilizing the convolution kernel technique known for its success in image processing to acquire high-level features. For processing long-term sequence data and improving detection accuracy, LSTM also uses gate functions to control the remembering and forgetting of prior data. The CNN-LSTM algorithm paradigm is thus appropriate for processing intrusion detection in this investigation. We chose CNN and LSTM for the identification of DDOS assaults based on the findings from prior studies and similar works. The establishment of a blacklist of aberrant traffic allows for the successful tracing of attack sources in this manner. Memory usage is minimized by storing solely the abnormal packets recognized by the IDS.

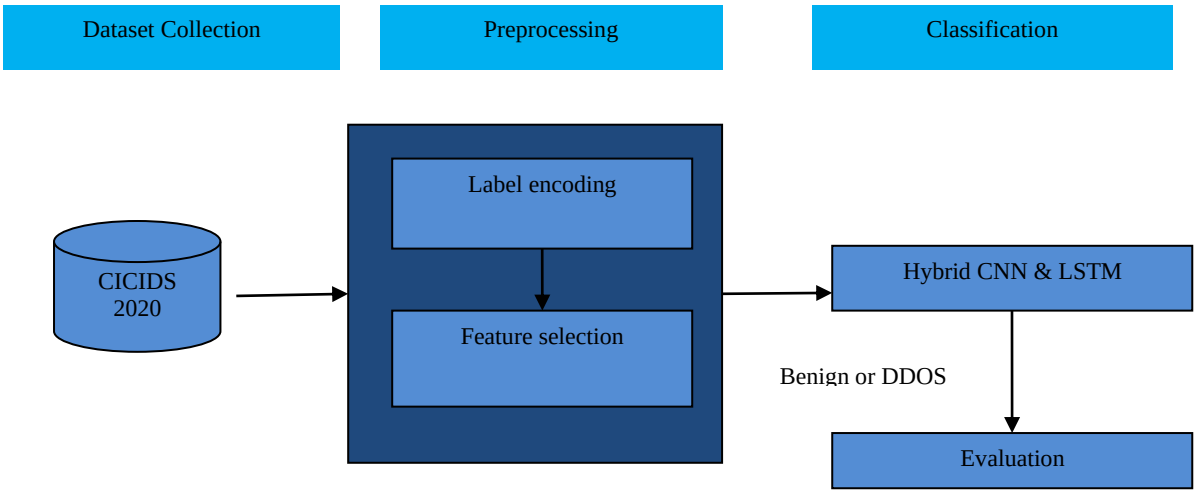

Figure 2: Block diagram for the proposed hybrid CNN-LSTM

Data Set Preprocessing

Data preprocessing involves preparing raw data for subsequent analysis or use. Real-world data is frequently irregular, fragmentary, and full of mistakes. Preprocessing data is a technique used to address these issues.

Feature Selection

The accuracy of a model's detection in machine learning greatly depends on the quality of its classification features. Effective feature extraction demands a blend of complex mathematics and intuitive assessments. In contrast, deep learning enables the automatic extraction of features across multiple layers, combining these traits to generate outputs at various levels.

Models for intrusion detection based on deep learning might result in significant computational expenses and could overfit during training, complicating the detection of multiple attacks. This is due to the excessive use of (unnecessary) features in these models. To effectively capture the traits of an attack, fewer features are required, but models that result from this tend to produce a heightened number of false alarms and reduced accuracy, potentially causing frequent false positives. As a result, choosing the right features is crucial because doing so will directly impact how well a model is trained.

DDOS Attack Detection Based on Proposed Hybrid CNN-LSTM

In order to effectively identify data, CNN primarily extracts object characteristics using many stacked convolutional and pooling layers. Nevertheless, CNNs employ fully connected BP neural networks as perceptrons, utilizing gradient descent in training to minimize global error. Consequently, CNN perceptrons are deemed inadequate classifiers due to extended training periods and limited network generalization. On the other hand, LSTMs operate like single hidden layer feedforward neural networks. Their hidden layer's weights and biases are initially randomized, removing the necessity for extra tuning to reach the best solutions during training. Consequently, LSTMs offer the benefits of rapid training, robust generalization, and high classification accuracy.

Because LSTM operates as a neural network with limited depth, its capacity to acquire features is constrained, requiring abundant training data for attaining high accuracy. Convolutional Neural Networks (CNN) and single-hidden layer feed-forward neural networks each exhibit unique strengths. Thus, combining CNN and LSTM produces a hybrid CNN-LSTM network, capitalizing on the advantages of both and addressing their individual shortcomings, as detailed in this section. The configuration of our CNN-LSTM model is outlined in Figure 3, illustrating a dual-phase network consisting of feature extraction and classification. The feature extraction phase involves convolutional layers and max pooling layers, while the classification phase utilizes a single hidden-layer feed-forward neural network (LSTM). Moreover, a comprehensive breakdown of relevant parameters—including filter quantity, feature mapping size, filter kernel, and sliding window step size—is provided.

As an illustration, during the initial phase, the convolutional layer employs the Relu activation function and integrates 64 filters with a 3x3 kernel, utilizing a sliding window step size of 1. Moving onto the third stage involves integrating 128 filters using a 3x3 kernel in the convolutional layer, employing a sliding window step size of 1 and utilizing Relu as the activation function. second phase includes implementing a stride size of 2 along with a 2x2 maximum pooling kernel. the fourth phase includes a 2x2 maximum pooling operation as part of the feature mapping process. To adapt the LSTM model for DDoS attack classification among regular flows, the fifth stage transforms the feature mapping from the preceding stage into a one-dimensional vector. We then go into depth about the construction of this portion of the hybrid model.

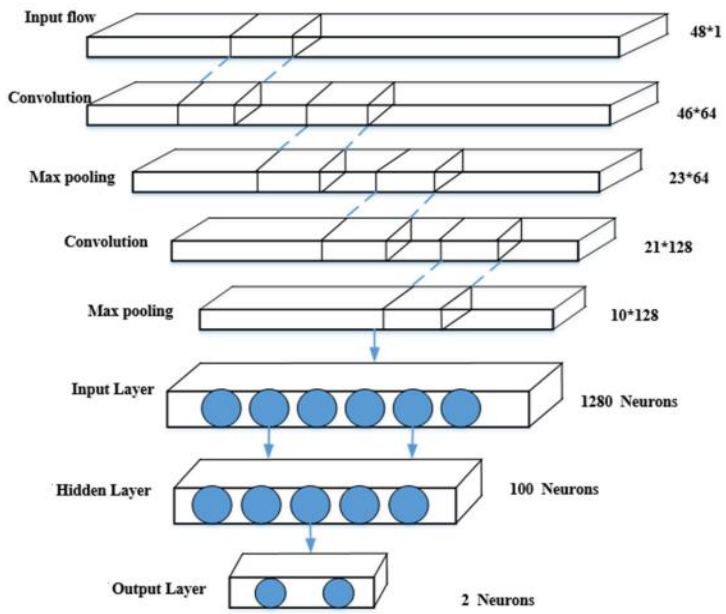

Figure 3: Deep learning hybrid model CNN-LSTM.

Each convolutional layer utilizes several convolution kernels, which convolve with input feature vectors, resulting in the creation of corresponding feature maps. However, because each convolution kernel has the ability to share weights, the complexity of the model can be effectively reduced while memory usage is also decreased. The process of convolution can be represented as:

$$Z^{(l,m)} = f(w^{l,m}) \times P^{(l-1)} + b^{(l,m)} \quad (1)$$

Pooling layer It effectively addresses network overfitting by compressing features in the convolutional layer, reducing their dimension while preserving local invariance and cutting down on network computation. This method is referred to as the subsampling layer.

The maximum pooling function is chosen in this study, and its formula is given as

$$P^{l(i,j)} = max_{(j-l)V+l \le t \le jV}\{a^{l(i,j)}\} \quad (2)$$

Completely interconnected layer Its primary function is feature classification, which can reduce the number of dimensions in a feature vector from two to one by nonlinearly mixing the data from the convolutional and pooling layers. The mathematical formulation of it is:

$$y = f(\sum_{i \in G_i} x^{(l-l,i)} W^{(l,i)} + b^{(l,j)}\} \quad (3)$$

Numerous activation functions are commonly employed, with the Relu function being utilized in this study as the activation function. Recognized for its simplicity and fast computation, it effectively minimizes network time complexity, leading to accelerated convergence speed.

Mitigation Method Based on the Proposed IP Traceback

We utilize the network structure shown in Figure 5 to conduct simulation experiments showcasing the IP traceback procedure. The setup comprises 4 hosts (h1, h2, h3, and h4) and three controllers (C1, C2, and C3). While h3 and h4 serve as typical user hosts, h1 and h2 take on the role of attackers. Specifically, h1 and h2 initiate a DDoS attack on host h4 using the hping3 traffic generator. The attributes utilized for traceback in this investigation follow the sequence of dpid, in port, src IP, dst IP, protocol, eth type, src MAC, dst MAC, origin, and timestamp, enabling the effective identification of the attack's source. The message sent to the controller specifies the switch ID through the 'dpid' property and the input port number through the 'in port' property within the ofpt packet.

Both properties specify where the packet entered the network at its ingress point. The packet's header fields include src ip, dst ip, protocol, eth type, src mac, and dst mac. Identifying the origin is pivotal in determining whether the attack stems from within the current domain. The variable 't stamp' signifies the time when the ofpt packet arrives at the controller in the message. It's essential to underscore that the Source Mac and Destination Mac addresses remain constant during the transmission of packets within the identical control domain, ensuring traceability. These ten attributes play a key role in precisely identifying the origin of an attack. In the event of a network 3 host being targeted, the SDN security defence system relies on data from these ten characteristics. This process activates the trace-back module, which blacklists the unusual flow within the controller. This blacklist uses the Source IP address and Destination IP address as reference points, sorting them according to the 't stamp' attribute.

if the attack packet comes from network 3, the trace-back procedure can reveal the entire path, Figure 6 illustrating the step-by-step process. Using the East-West interface, the controller within network 3 will communicate with adjacent control domains to pinpoint the source IP of the packet. Once the control domain housing the attacker is determined, this paper will arrange the attack paths according to the timestamp attribute. Table 3 exhibits the attack traceback table for the source IP 10.0.0.2 and destination IP 10.0.0.4.

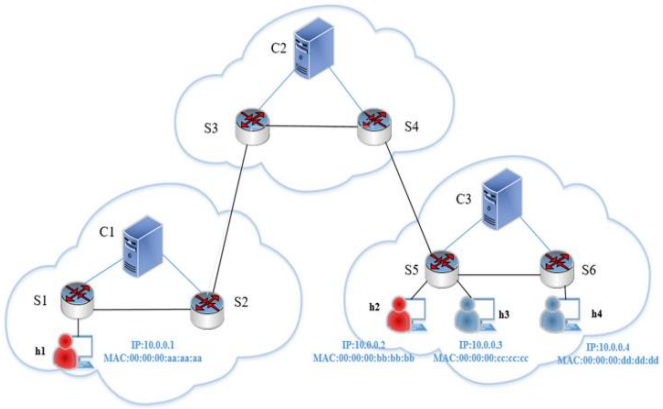

Figure 4: Network topological diagram.

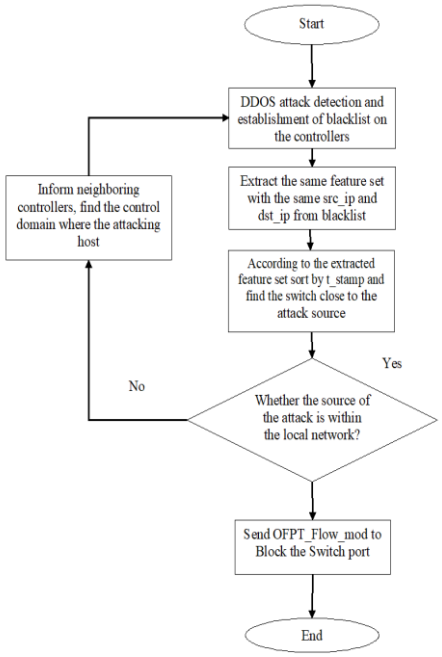

Figure 5: Flow chart of mitigation mechanism based on IP traceback.

Table 1: IP traceback table for S5 in control domain 3

Dpid	In-Port	Src_IP	Dst_IP	Protocol	Src_Mac	Dst_mac	Origin	Controller-IP
5	1	10.0.0.2	10.0.0.4	6	00:00:00 Bb:bb:bb	00:00:00 Dd:dd:dd	Yes	192.168.1.1
5	1	10.0.0.2	10.0.0.2	6	00:00:00 Bb:bb:bb	00:00:00 Dd:dd:dd:	Yes	192.168.1.1.
5	1	10.0.0.2	10.0.0.4	6	00:00:00 Bb:bb:bb	00:00:00 Dd:dd:dd	Yes	192.168.1.1

In this scenario, it is assumed that the attack flow originates from the source IP 10.0.0.1 and targets the destination IP 10.0.0.4 for cross-domain attacks. The traceback table for S5 in control domain C3 is presented in Table 2.

Table2. IP traceback table for S5 in control domain 3

Dpid	In_port	Src_IP	Dst_IP	Protocol	Src_Mac	Dst_Mac	Origin	Controller_IP
5	1	10.0.0.1	10.0.0.4	6	00:00:00 Aa:aa:aa	00:00:00 Dd:dd:dd	No	192.168.1.1
5	1	10.0.0.1	10.0.0.4	6	00:00:00 Aa:aa:aa	00:00:00 Dd:dd:dd	No	192.168.1.1
5	1	10.0.0.1	10.0.0.4	6	00:00:00 Aa:aa:aa	00:00:00 Dd:dd:dd	No	192.168.1.1

Experimental Results and Discussion

The Ubuntu 16.04 LTS operating system and a VMware with 8 GB of RAM were used for this experiment. To simulate SDN, a mininet emulator was employed, and a POX controller was linked to a network comprising two switches. These switches are linked to 43 hosts. Among them, four hosts (h1, h2, h3, and h4) serve as clients, while three hosts (h13, h23, and h33) are designated as attackers. Host h52 serves as the server. Attack traffic has been sent using the Hping3 tool, while traffic flow monitoring and analysis are done with Netdata Agent. A setup featuring a core i3 processor and 4 GB of RAM is running Netdata, serving as a real-time traffic monitoring tool.

Datasets

The efficiency of an Intrusion Detection System (IDS) greatly hinges on the calibre of its training data. Regrettably, advancements in anomaly detection systems encounter challenges primarily due to the lack of benchmark datasets tailored specifically for intrusion detection objectives. Numerous datasets are available for testing diverse machine-learning methods across various industries such as biomedicine, language translation, and more. However, the paucity of network intrusion detection datasets is primarily due to privacy and security concerns.

Limited access to network intrusion detection datasets arises due to the existence of sensitive information within them. Disclosing such data could potentially damage the reputation of the individuals or entities involved. Furthermore, the majority of easily accessible datasets, like the traditional KDDCUP99 and NSLKDD, have become outdated and no longer provide an accurate representation of the latest network traffic patterns. Additionally, several other datasets lack traffic diversity and don't include all known attack methods.

The CICIDS-2017 dataset stands out as one of the latest additions in the realm of intrusion detection datasets. Transformed into CSV format to enhance usability in intrusion detection tasks, this dataset reflects real-world data (PCAPs) and covers recent benign attacks. In this investigation, we utilize the InSDN dataset (Elsayed et al.) to assess how well our suggested deep learning model performs. This stage is crucial in effectively showcasing the performance of our suggested model. The InSDN dataset holds current examples of commonly seen attack styles. Additionally, it contains typical traffic for essential services. Noteworthy is that the InSDN dataset employs the CICFlowMeter app, a freely accessible tool, to extract over 80 statistical measures from a large dataset consisting of 343,939 instances of both regular and attack-related traffic. This dataset aims to simulate both

internal and external SDN attacks. As a result, the dataset closely resembles attack data from actual network environments.

Data Pre-Processing

Before supplying the data to the learning classifier, we must pre-process them to simplify system operation and create a real-time intrusion detection system. These details outline the task requirements.

- The data must be rescaled because the ranges of the features vary. All data are mapped to a range between 0 and 1.
- By employing the train-test split functionality from the Scikit-learn library, the experimental dataset is divided, allocating 20% for testing and 80% for training.
- The labelled category transforms the label into a distinct number using the special heat encoding method. In this paper, the anomaly detection technology is denoted as the binary classification technique. Specifically, binary 0 signifies the normal category, while binary 1 represents the abnormal category in the context of binary classification detection.

Detection of DDOS Attacks Using Proposed Hybrid CNN-LSTM

Initially, we conducted training and assessment on our CNN-ELM model using 12 feature subsets sourced from the CICIDS-2017 dataset. Following that, we replicated the process using 48 distinct feature subsets from the same dataset. Subsequently, we compared the achieved results with those of alternative models.

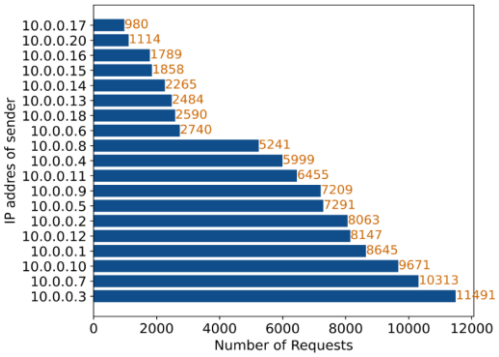

Figure 6: The average IP address of the sender for various numbers of requests

Figure 6 shows that for the typical TCP traffic, a total of around 100 kilobits per second of bandwidth was used. Throughout the entire experiment, the received and sent N values remained consistent. However, in the case of the attack with mitigation, we witnessed an initial surge in bandwidth consumption, reaching up to 150 kilobits per second. Fortunately, the mitigation measures swiftly kicked in, returning the bandwidth consumption to its normal levels. At the onset of the attack, there was a rapid escalation in bandwidth usage, reaching nearly 3 Mbps and fully saturating the network bandwidth. Subsequently, around t=135 seconds, a slight increase was noted, attributed to the mitigation algorithm temporarily allowing traffic to prevent false positives.

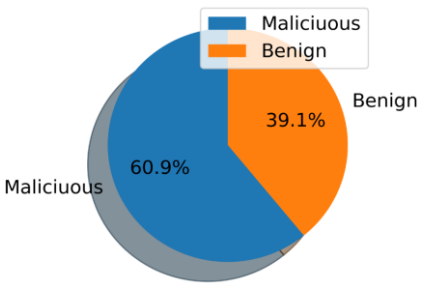

Figure 7: Pie chart representing malicious and benign attacks

Figure 7 shows the pie chart representation of malicious and benign attacks. Nearly 60% of data are affected by malicious attacks. Nearly 40% of data are affected by benign attacks.

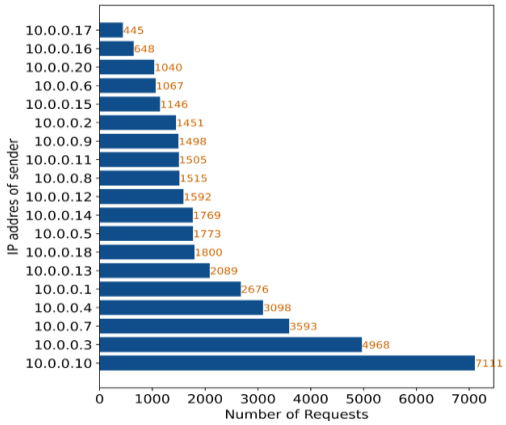

Figure 8: The average IP address of the sender for various numbers of requests

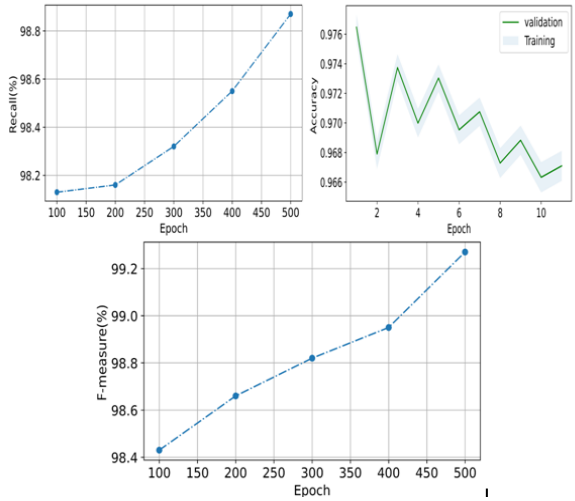

Figure 9: Comparison between epoch and various performance measures like recall, accuracy, and F-measure.

Figure 9 compares the epoch with several performance metrics such as recall, accuracy, and F-measure. The value of recall starts low and rises as the epoch value rises. Initially, the value of the F-measure is similarly quite low before progressively rising as the epoch value rises.

The CNN-LSTM hybrid model demonstrates superior performance compared to other ML models when applied to both the 12-feature and 48-feature subsets. This indicates that the 12 feature subset, specifically chosen in this study, effectively captures the distinguishing characteristics of attack traffic. Additionally, it demonstrates efficiency by being more concise and quicker during model training compared to the 48-feature subset.

Table 4: Comparison results of CNN-LSTM models of 5 feature subsets based on the CICIDS-2017 dataset

Method	Accuracy(%)	Recall(%)	Precision(%)	F1-Score(%)	TestTime(s)
CNN (Aydin et al.,2022)	97.98%	99.45%	96.45%	98.08%	26.13s
SVM (Jagtap et al.,2022)	98.45%	99.38%	97.69%	98.48%	34.56s
CNN-ELM (Bhale et al.,(2023)	89.45%	99.21%	96.92%	98.28%	5.13s
ELM (Mihoub et al.,2022)	93.02%	99.45%	85.43%	91.02%	4.26s
CNN-LSTM	98.92%	99.67%	97.82%	98.74%	3.65s

Table 4 shows that the ultimate CNN-LSTM hybrid model outperformed other ML models in accuracy, recall, precision, and F1-score. This superiority was evident across both the 12-feature and 48-feature subsets. Despite the CNN-LSTM model exhibiting only a marginal accuracy improvement over the CNN, CNN-ELM, and CNN-SVM models (0.01–0.02%), its testing time is significantly faster than that of the other models.

Conclusion

This research addresses the issue of ineffective DDoS attack detection in SDN by proposing a hybrid model called CNN-LSTM. The CNN component is utilized for extracting network traffic features, while the LSTM algorithm is employed for data classification. Utilizing the centralized control and management features of SDN on a global scale, this research aims to reduce abnormal traffic, pinpoint the origins of attacks, and promptly alert the nearest controller to eliminate disruptive traffic, thus preventing DDoS attacks at their source. The efficiency of the SDN defensive system was confirmed through simulations conducted on the Mininet platform. The results of these experiments confirm the robust detection capabilities of the proposed CNN-LSTM model. The hypothesis test accuracy for the CICIDS-2017 dataset is 98.92%, and for the InSDN dataset, it is 99.91%. During these experiments, the suggested SDN-based IP traceback approach adeptly traces the origin of DDoS attacks. The study's method of identifying abnormal traffic relies on supervised learning, which can be expensive for classifying essential data. In our forthcoming research, we aim to investigate the utilization of graph neural networks and unsupervised learning techniques for attack traceback. The goal is to detect anomalies in DDoS attacks within genuine network settings.

References

Adimas, Ayenew Kassie, et al. "DDOS ATTACK DETECTION IN DISTRIBUTED SDN SYSTEM USING DEEP LEARNING."

Aydın, Hakan, Zeynep Orman, and Muhammed Ali Aydın. "A long short-term memory (LSTM)-based distributed denial of service (DDoS) detection and defence system design in public cloud network environment." *Computers & Security* 118 (2022): 102725.

Dora, V. Raghava Swamy, and V. Naga Lakshmi. "Optimal feature selection with CNN-feature learning for DDoS attack detection using meta-heuristic-based LSTM." *International Journal of Intelligent Robotics and Applications* 6.2 (2022): 323-349.

Jagtap, Monica Murlidhar, and Renuka Devi Saravanan. "Intelligent software-defined networking: Long short term memory-graded rated unit enabled the block-attack model to tackle distributed denial of service attacks." *Transactions on Emerging Telecommunications Technologies* 33.11 (2022): e4594.

Meamarian, Mohsen, and Naser Yazdani. "A Robust, Lightweight Deep Learning Approach for Detection and Mitigation of DDoS Attacks in SDN." *2022 27th International Computer Conference, Computer Society of Iran (CSICC)*. IEEE, 2022.

Bhale, Pradeepkumar, et al. "OPTIMIST: Lightweight and Transparent IDS with Optimum Placement Strategy to Mitigate Mixed-rate DDoS Attacks in IoT Networks." *IEEE Internet of Things Journal* (2023).

Lent, Daniel M. Brandão, et al. "A gated recurrent unit deep learning model to detect and mitigate distributed denial of service and portscan attacks." *IEEE Access* 10 (2022): 73229-73242.

Brindha Devi, V., Nihar M. Ranjan, and Himanshu Sharma. "IoT Attack Detection and Mitigation with Optimized Deep Learning Techniques." *Cybernetics and Systems* (2022): 1-27.

Mihoub, Alaeddine, et al. "Denial of service attack detection and mitigation for Internet of things using looking-back-enabled machine learning techniques." *Computers & Electrical Engineering* 98 (2022): 107716.

Wang, Jin, and Liping Wang. "SDN-Defend: A Lightweight Online Attack Detection and Mitigation System for DDoS Attacks in SDN." *Sensors* 22.21 (2022): 8287.

Dahiya, Deepak. "DDoS Attacks Detection in 5G Networks: Hybrid Model with Statistical and Higher-Order Statistical Features." *Cybernetics and Systems* (2022): 1-26.

Kathirkamanathan, Nisanthan, et al. "Prevention of DDoS attacks targeting financial services using supervised machine learning and stacked LSTM." *2022 IEEE 7th International Conference for Convergence in Technology (I2CT)*. IEEE, 2022.

Yousuf, Omerah, and Roohie Naaz Mir. "DDoS attack detection in Internet of Things using recurrent neural network." *Computers and Electrical Engineering* 101 (2022): 108034.

Basnet, Manoj. *Deep Learning-Powered Computational Intelligence for Cyber-Attacks Detection and Mitigation in 5G-Enabled Electric Vehicle Charging Station*. Diss. The University of Memphis, 2022.

Myneni, Sowmya, et al. "SmartDefense: A distributed deep defence against DDoS attacks with edge computing." *Computer Networks* 209 (2022): 108874.

Maheshwari, Aastha, et al. "An optimized weighted voting based ensemble model for DDoS attack detection and mitigation in SDN environment." *Microprocessors and Microsystems* 89 (2022): 104412.

Aldhyani, Theyazn HH, and Hasan Alkahtani. "Artificial Intelligence Algorithm-Based Economic Denial of Sustainability Attack Detection Systems: Cloud Computing Environments." *Sensors* 22.13 (2022): 4685.

Adefemi Alimi, Kuburat Oyeranti, et al. "Refined LSTM Based Intrusion Detection for Denial-of-Service Attack in Internet of Things." *Journal of Sensor and Actuator Networks* 11.3 (2022): 32.

Ravinarayanan, B., and H. R. Nagesh. "A Hybrid Model for DDoS Attack Detection Using LSTM-RNN." *Smart Intelligent Computing and Applications, Volume 2: Proceedings of Fifth International Conference on Smart Computing and Informatics (SCI 2021)*. Singapore: Springer Nature Singapore, 2022.

Yeom, Sungwoong, Chulwoong Choi, and Kyungbaek Kim. "LSTM-based collaborative source-side DDoS attack detection." *IEEE Access* 10 (2022): 44033-44045.

Sinthuja, M., and K. Suthendran. "DDoS Attack Detection using Enhanced Long-Short Term Memory with Hybrid Machine Learning Algorithms." *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*. IEEE, 2022.

Kar, Jishnudeep, and Aranya Chakrabortty. "An Integrated Generative Adversarial Network for Identification and Mitigation of Cyber-Attacks in Wide-Area Control." *2022 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2022.

Behera, Bibhuti Bhushan, Rajani Kanta Mohanty, and Binod Kumar Pattanayak. "Attack Detection and Mitigation in Industrial IoT: An Optimized Ensemble Approach." *Specialusis Ugdymas* 1.43 (2022): 879-905.

Gudla, Surya Pavan Kumar, et al. "DI-ADS: a deep intelligent distributed denial of service attack detection scheme for fog-based IoT applications." *Mathematical Problems in Engineering* 2022 (2022): 1-17.

Mittal, Meenakshi, Krishan Kumar, and Sunny Behal. "Deep learning approaches for detecting DDoS attacks: A systematic review." *Soft Computing* (2022): 1-37.