Medicon Engineering Themes
Volume 6 Issue 3 March 2024
Article Type: Review Article
ISSN: 2834-7218

Distributed Denial of Service Attack Detection using Sequence-to-Sequence LSTM

Anand Parmar[1]* and Hemraj Lamkuche[2]

[1]*Symbiosis University of Applied Sciences (SUAS), Indore, India*

[2]*School of Computing Science and Engineering, VIT Bhopal University, Kothrikalan, Sehore Madhya Pradesh*

***Corresponding Author:** Anand Parmar, Symbiosis University of Applied Sciences (SUAS), Indore, India.

Received: January 27, 2024; **Published:** March 12, 2024

Abstract

Log files are a great way to find out what's wrong with a system and how secure it is. They can be very large and have a complicated structure, which is why they are so useful. We use Machine Learning (ML) to find network anomalies and build different models that are driven by data to find DDoS attacks. The main goal of this article is to reduce the number of times that DDoS detection is wrongly labeled. In this paper, we describe a method for security analysis that uses Deep Learning techniques like simple LSTM, LSTM with embedding, and Seq-to-Seq LSTM on several systems log files to find and extract data that may be related to distributed denial of service (DDoS) attacks made by malicious users who want to break into a system. Through a process of learning, these data will help to find attacks, predict attacks, or find intrusions. In this study, we looked at how different optimizers, the size of the hidden state, and the number of layers affected the same architecture to find the best way to set it up. When compared to other models, the proposed model was able to correctly identify DoS/DDoS packets that had never been seen before with a 98.95% level of accuracy.

Keywords: DDoS Attack Detection; Cyber Physical System; LSTM; Deep Learning; Machine Learning

Introduction

Denial of Service (DoS) attacks aim to crash a network or service by overwhelming it with traffic/requests, causing it to become unavailable to intended users. They can result in financial loss, reputational damage, and loss of data. They can be carried out by a single source or multiple sources (DDoS). DoS/DDoS attacks are a critical and rapidly evolving threat. This research has been conducted to analyze various DDoS attacks in different environments and setups. Some examples include, the study of the "ping of death" attack and its impact [1], the effect of such attacks on Software Defined Networking (SDN) [2], and the examination of how DDoS attacks can be executed and their impact on popular web servers such as Apache and IIS [3]. There have also been efforts to detect and counter DDoS/DoS attacks using Artificial Neural Networks (ANNs) and Machine Learning techniques. Some works utilize Convolutional Neural Networks (CNNs) [4, 5] to identify attacks, and others employ popular ML techniques [6] for detection and analysis. A few studies have employed Feed-Forward ANNs [7, 8] to detect DDoS attacks, and some examine the detection of DDoS/DoS attacks as a sequential problem and attempt to use Long-Short Term Memory (LSTM) or Recurrent Neural Network (RNN) networks to achieve this [9].

The DoS/DDoS attacks are difficult to detect and lack a comprehensive solution despite being one of the earliest cyber threats. They closely resemble legitimate traffic, making traditional techniques inadequate. The largest recorded DDoS attacks to date were on Google (2.54 Tbps), AWS (2.3 Tbps), and GitHub (1.3 Tbps). Building on prior work, our new methodology for detecting DoS/DDoS attacks using ANN addresses various issues seen in previous studies such as data imbalance. We collect data in sessions, four sessions were recorded in our study. Unlike many previous studies, we also incorporate IP addresses as a feature by utilizing the Embedding tech-

nique. We use accuracy, precision, recall, and F1 score for performance measurement using an unseen dataset. For the classification technique, we present a novel approach to counter Dos/DDos attacks using ANN, utilizing a Sequence-to-Sequence architecture with LSTM and attention mechanism. This is a new approach in the field, as prior works have limited use of this technology. We chose the Sequence-to-Sequence model for its efficiency in processing sequential data, and LSTM for its ability to handle large data sets in the context of Dos/DDoS attacks. Our approach emphasizes utilizing underutilized features. We further improve our model by testing different configurations of hidden state, layer count, batch size, activation functions, and optimizers during training. Finally, we compare our approach against other methods.

The paper is structured as follows: Section 2 reviews prior research in the field. Section 3 covers the data collection, preprocessing, analysis, and storage, as well as the description and comparison of our deep learning model architecture and its results. Section 4 looks at conventional DDoS detection methods, and Section 5 summarizes and concludes our work.

Literature Review

In the past, various solutions have been created to combat DoS/DDoS attacks on SDN and other types of networks. As noted in [2], most of these solutions as of 2021 have been tested using simulators. However, there are a few strategies that are focused on working under an attack, such as [10-12], while others are focused on mitigation or prevention. Some strategies typically involve using statistical analysis to identify the network's status, such as tracking resource usage and request volume. Additionally, a few solutions involve adding extra hardware to the network [10, 13], such as caches, middleboxes, and third-party servers.

Research has been conducted to assess the effects of DoS/DDoS attacks on cloud environments [14]. Specifically examining the impact of slow HTTP headers (slow-loris), slow HTTP POST, and slow read attacks on the target virtual machine in terms of CPU, RAM, and network usage, as well as the impact on neighboring machines in terms of response time. The research found that even a small amount of low-rate traffic from a single attacker can negatively affect neighboring VMs and reduce web server response time by 2.09% and 11% when using a distributed DoS attack.

Further studies have been conducted to examine the use of machine learning algorithms such as Random Forests, Naive Bayes, and Support Vector Machines (SVMs) to detect DDoS attacks on cloud platforms [6]. SNORT was also utilized to detect all the attacks and provide data for the study. The study found that among the three algorithms used, SVM had the highest accuracy, recall, and precision. However, the study also highlights that the data used in the study was imbalanced.

Fekadu Yihunie et al describe the effects of 'Ping of Death' DoS and DDoS attacks in their research [1]. Their work examined the effects of the respective attack in three different network topologies, 1) A well-functioning network serves as a reference point for comparing response time with the next two scenarios. 2) The first scenario is replicated and a malicious node is added to the network. 3) The second scenario is replicated and two more attacking machines are added, increasing the number of ping requests by the malicious nodes on the server.

Additionally, research has been conducted to examine the impact of DoS/DDoS at-tacks on web servers [3]. Rizgar et al in their study compared the effects of SYN and HTTP flood on two web servers Apache 2 and IIS 10.0 using the HOIC [15] tool. The study shows that both servers were unable to function during the attack, but IIS was found to be more stable under both attacks. The study also focuses on the impact of SYN flood attacks, as there is limited research on this type of attack.

Recently, several proposals have been made to use neural networks for identifying DoS/DDoS attacks, one such example of the use of ANN for DDoS detection includes, Multi-layer perceptron (MLP) [8] on CharGen, DNS, and UDP attacks. Evaluated through computer simulation, this work has shown outcomes with higher accuracy of 95.6%.

Another research on the use of Artificial Neural Networks (ANN) for DDoS detection is the research work done by Shahzeb et al [4]. Shahzeb et al in their work proposes a CNN-based ensemble mechanism for the detection of different flow-based DDoS attacks in SDN.

The CICIDS2017 dataset is used for evaluation, the dataset contains 40% DDoS and 60% normal traffic. The study concludes that the use of multiple CNN models in an ensemble-style architecture improves both detection accuracy and computational efficiency.

Jieren et al [5] suggest a method for identifying DDoS attacks using a convolutional neural network to address the issue of high false and missing alarm rates in big data environments. They introduce the use of Gray Scale Matrix Features (GMF) to train the CNN model, and test it using SYN/SYN+ACK flood attacks from the CAIDA "DDoS Attack 2007" dataset. Their proposed method is reported to be more accurate than comparable detection techniques, it also shows lower rates of false alarms and missed alarms. Additionally, it can effectively detect DDoS attacks in big data environments.

Although most research proposals focus on detecting Application layer DDoS attacks, there is also a need for Network layer DDoS detection systems, particularly for IoT devices. The research [16] proposes a simple classifier utilizing a feedforward neural network with backpropagation, it is capable of distinguishing between normal and malicious traffic produced by IoT devices communicating with machines via the MQTT protocol. The study covers TCP, UDP, ARP, and ICMP attacks [17]. The final model that uses 8 features was found to be superior to the same model using 10 features and an RNN using 10 features.

Another interesting study consisting of the use of DNN is the work done by Aanshi et al [18] Their work presents an architecture that utilizes a well-designed Autoencoder (AE)[19] to address the difficulties of efficient feature learning, dealing with noisy data, and avoiding overfitting. The proposed architecture is compared to ten other machine-learning techniques on NSL-KDD and CICIDS2017 datasets. As a result, the proposed methodology was able to classify DDoS/DoS traffic with an accuracy of 98.43% and 98.92% on the NSL-KDD and CICIDS 2017 datasets.

Another popular study "Detection of known and unknown DDoS attacks using Artificial Neural Networks" [7] aims to classify DDoS attacks and to evaluate the performance of an Artificial Neural Network (ANN) when it is trained with both old and recent datasets. Features such as IP address, TCP sequence number, and port numbers were used. The results indicate that the trained model classified 95% of unknown and 100% of known DDoS attacks when tested on old and recent datasets.

Another study that uses Recurrent Neural Networks (RNNs) for DDoS detection is the research by Xiaoyong et al [9]. In their work, they tackle the issue of DDoS detection by treating it as a sequence classification problem and converting packet-based detection to window-based detection. The study trains various models including CNN, RNN, LSTM, and GRU [20], on the UNB ISCX dataset. The findings reveal a decrease in error rate by 39.69% when compared to the shallow machine learning methods on a small dataset, and a decrease in error rate from 7.517% to 2.103% on a larger dataset.

Proposed Methodology

Here we present our deep learning-based approach to classify DoS/DDoS attacks, we will detail its architecture and the methods used for data preprocessing, collection, storage, and analysis. We will then discuss the experiments conducted and the DDoS strategies employed.

Data Collection

We employ our solution using AWS cloud services, the lab setup is as follows:

I. 5 machines running bot-scripts to simulate benign traffic. These machines randomly send requests to random routes in our web server within random intervals.
II. Kali-Linux machine acting as the attacker.
III. The target web server running on PORT 80, which uses Flask, has five distinct routes that support both POST and GET requests along with authentication, as well as JavaScript, CSS, HTML, and image resources.

Our proposed method involves collecting and organizing data into sessions, as this approach addresses the issue of imbalanced data

caused by the faster speed of DoS/DDoS packets compared to normal packets. It also allows for the capture of a wider range of features, such as variations in size, balance, addresses, and timestamps.

The following datasets were recorded as follows:

Dataset	*Start Time*	*Attack Started*	*Stop Time*	*Length*	*Benign Count*	*DDOS Count*
dataset-1	22:00	22:30	22:37	121784	70956 (59%)	50828 (41%)
dataset-2	23:17	23:50	23:55	60543	50663 (83%)	9880 (17%)
dataset-3	16:10	16:41	17:00	171296	89229 (53%)	82067 (47%)
dataset-4	9:30	10:05	10:17	267830	120292 (45%)	147538 (55%)

Table 1: Datasets.

Data Processing

Our method uses TCP and IP properties of packets as captured in our datasets. It aims to make use of features, such as IP addresses, that are not commonly utilized.

We utilize Source and Destination IP addresses by Embedding them. Embedding, published in 2013 is a technique used for NLP tasks [21]. Embedding employs a neural network to discover associations, which we utilize to link various IP addresses. This approach allows our model to view the addresses as distinct but related entities instead of continuous values, and also aids in identifying addresses that belong to the same subnet. We input each octet of the IP address into the embedding process, converting it into a single value between 0 and 1. Finally, we concat the result of embedding all octets of the address. We repeat this process for every octet in both the Source and Destination IP addresses. for example: "127.0.0.1" becomes "0.98, 0, 0, 0.01". These values are learned during training. The same is done for destination and source PORT numbers. We also introduce the "direction" feature, which allows us to categorize packets as incoming or outgoing. This is a simple yet effective feature. Finally, TCP and IP flags, and their other properties were used as features.

Data Storage

Our goal is to make the data easily accessible for various purposes. To achieve this, we store the data in a CSV format, which is widely supported. This allows for easy visualization and analysis using various tools and can also be easily consumed for various purposes.

Field	*Field Type*	*Field Example*
SRC	String	"192.168.1.1"
DST	String	"192.168.1.2"
DIRECTION	Bool	[1,0]
IP_TOS	Int	[1,0]
DF	Int	[1,0]
IP_TTL	Int	64
TCP_SPORT	Int	80
TCP_DPORT	Int	4456
TCP_RESERVED	Int	[1,0]
FPA	Int	[1,0]
FA	Int	[1,0]
A	Int	[1,0]

S	Int	[1,0]
SA	Int	[1,0]
PA	Int	[1,0]
R	Int	[1,0]
RA	Int	[1,0]
TCP_WINDOW	Int	489
TCP_MSS	Int	1460
TCP_WSCALE	Int	7
TYPE	Int	[1,0]

Table 2: Feature List.

Data Analysis

Our method for identifying DDoS/DoS attacks utilizes Deep Learning techniques, specifically RNN/LSTM, and considers the task as a sequence problem. We experiment with various architectures and evaluate their performance. We contrast models that do not incorporate Embedding with models that do and models that employ Seq2Seq architecture.

Our solution functions as a filter that examines each incoming packet, discards any suspicious packets, and records all transaction data for analysis and to retrain the classifier as shown in Figure 1.

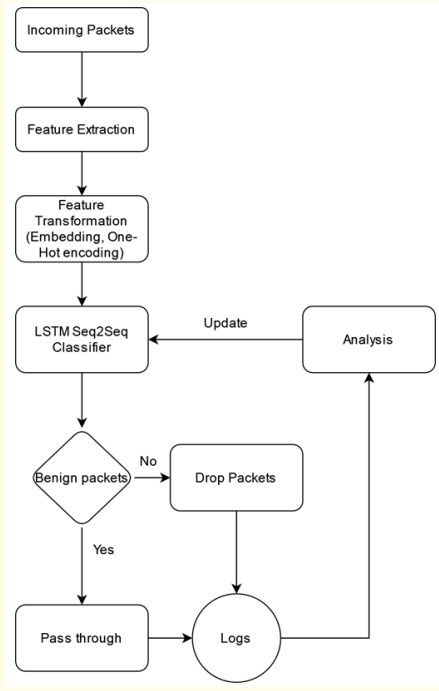


Figure 1: Flow diagram of DoS/DDoS detection system.

Based on previous studies using Encoder-Decoder architecture to detect DDoS/DoS attacks [18], we propose utilizing Seq2Seq architecture with attention for improved DDoS/DoS attack identification.

In the past Seq2Seq models [22] have shown success as generative models in various natural language processing tasks. In our approach, we aim to harness this generative capability of recurrent neural networks by using Seq2Seq architecture with an Attention mechanism.

We utilize an Encoder that condenses a batch of input data into a batch of outputs and a single Hidden State using LSTM, which is then passed to the Decoder mechanism that applies attention and generates its own output. This output, when processed through a feed-forward network, can predict the type of input.

We choose to employ LSTM in our proposed methodology because of its ability to handle longer sequences more effectively than traditional RNN and GRU models.

As illustrated in Figure 2.

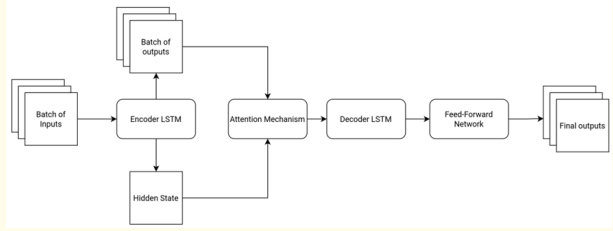


Figure 2: Deep Learning Based DDoS/DoS classifier.

Results

As described before data used in this proposal is based on sessions. We recorded 4 datasets out of which datasets 1-3 are used for training and dataset 4 is used for validation. The study examines three distinct architectures, including Simple LSTM, LSTM with Embedding, and LSTM in the Seq2Seq technique.

The training approach involved, training and validating the model on different optimizers and learning rates and choosing the most optimal one. This was done for each architecture. A module was used to calculate the steepest gradient and use the findings to figure out the most optimal optimizer with the most optimal Learning rate. Figure 3, 4, and 5 shows the steepest gradient for three different optimizers.

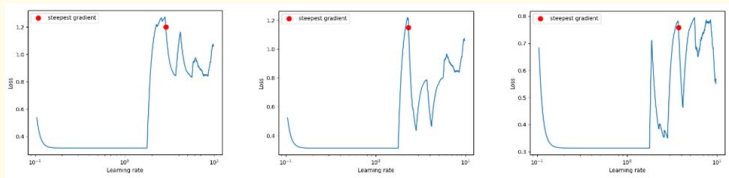


Figure 3: Adam. ***Figure 4:*** AdamW. ***Figure 5:*** SGD.

After identifying the optimal set of hyperparameters, each architecture was evaluated, and the best one was chosen for additional fine-tuning. We show the performance difference between the three architectures based on Accuracy, Precision, and Recall in Table 3.

Architecture Used	*Accuracy*	*Precision*	*Recall*	*F1*
Simple LSTM	94.80	0.923	0.963	0.942
LSTM with Embedding	97.80	0.923	0.974	0.982
Seq-to-Seq LSTM	98.95	0.992	0.983	0.974

Table 3: Model Comparison.

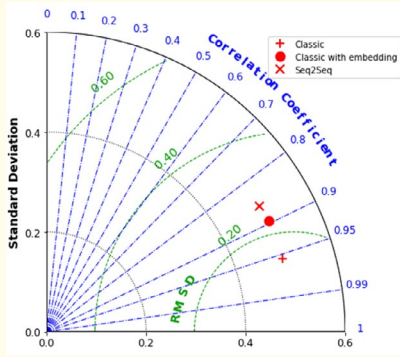


Figure 6: Taylor representation of different models.

Based on these results we selected Seq2Seq LSTM with Embedding as our final model architecture. We further re-evaluate the selected model for its best configuration. Figure 7 to Figure 10 shows the results of training and evaluating different configurations for 10 iterations based on the different number of LSTM layers, Hidden state size, different activation functions, and different batch sizes.

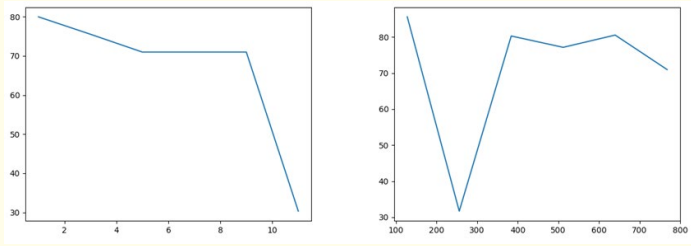
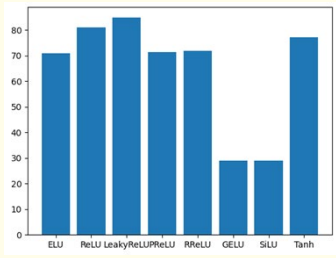


Figure 7: Number of layers. ***Figure 8*****:** Hidden state size.



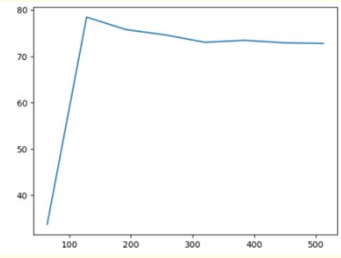


Figure 9: Activation functions. ***Figure 10:*** Batch sizes.

The finalized model was trained for 30 iterations and validated on dataset 4. The model was able to accurately identify previously unseen DoS/DDoS packets with an accuracy of 98.95%, which was the highest achieved with this approach.

In this study, we showed that log-based methodologies for detecting Dos/DDoS attacks are becoming obsolete and how the application of Artificial Neural Networks and Machine Learning Techniques is becoming more prominent in this field. This study also shows that there are still some gaps left for improvement in current work. We prove that by utilizing LSTM based Sequence-to-Sequence architecture for the said task and comparing it with other methodologies. Clearly the proposed method outperforms rest of the techniques, as it can be clearly seen in Table 3 and Figure 6 above. This study also contributes new data collection and utilization techniques that could be used as a base for future studies.

DDoS Strategies

Our methodology is based on the TCP SYN Flood attack. The impact of the attack is recorded in all benign machines connected to the network in terms of Response time, Status codes, etc. Below we discuss the impact of DDoS attacks and a primitive technology to detect such attacks.

Bandwidth Analysis

Bandwidth analysis is a technique used to detect and mitigate DoS/DDoS attacks. During a DDoS attack, an attacker floods a network with traffic from multiple sources in an attempt to overload the system and disrupt service [26]. Bandwidth analysis can be used to identify the attack and the type of attack being used. This information can then be used to take appropriate countermeasures, such as blocking traffic from the identified sources or implementing rate limiting to mitigate the attack. Additionally, monitoring the traffic flow over time can also help identify patterns that may indicate an ongoing attack, allowing for early detection and preve2ntion [27].

Resource Consumption

During a DDoS attack, an attacker floods a network with traffic from multiple sources in an attempt to overload the system and disrupt service. One of the ways this is done is by consuming resources on the targeted system [23]. Resource consumption attacks are designed to consume resources such as CPU, memory, and network bandwidth, making the targeted system unavailable to legitimate users. There are several types of resource consumption attacks such as CPU, Memory, and bandwidth consumption attacks.

In our case, the impact of the attacks can be observed on non-malicious machines attempting to communicate with the targeted server.

As illustrated in Figure 11 to Figure 14, during an attack, the response time from the server increases due to the overload caused by excessive DoS/DDoS requests, resulting in the inability to promptly process legitimate requests.

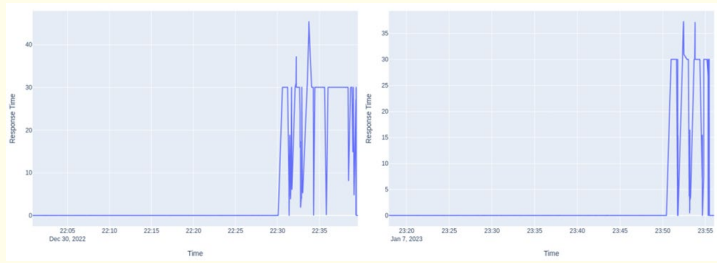


Figure 11: Dataset-1 response time. ***Figure 12:*** Dataset-2 response time.

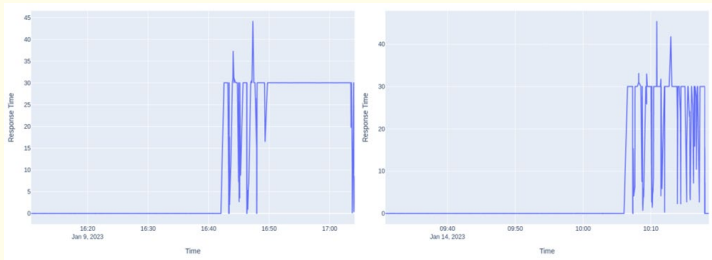


Figure 13: Dataset-3 response time. ***Figure 14:*** Dataset-4 response time.

Figure 15 to Figure 18 shows that during an attack, a higher number of 500 status codes are returned, indicating that the server encountered an internal error and was unable to process the request and provide a valid response.

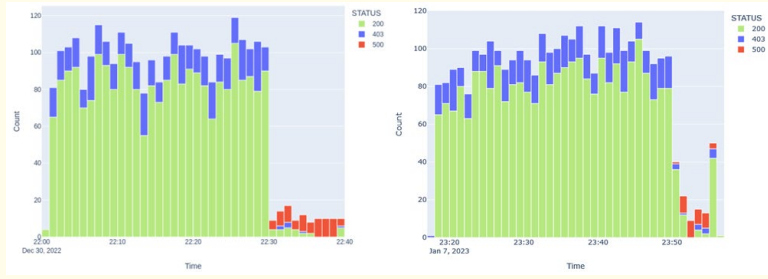


Figure 15: Dataset-1 status count. ***Figure 16:*** Dataset-2 status count.

Figure 17: Dataset-3 status count. ***Figure 18:*** Dataset-4 status count.

Figure 19 to Figure 22 illustrates a comparison of various HTTP methods in terms of the count and status codes returned by the server.

Figure 19: Dataset-1 method count. ***Figure 20:*** Dataset-2 method count.

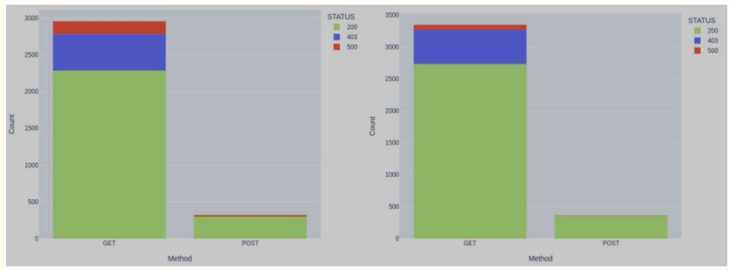


Figure 21: Dataset-3 method count. ***Figure 22:*** Dataset-4 method count

Software Flaws

Software flaws, also known as vulnerabilities, can be exploited by attackers. These flaws can exist in a wide range of software, including operating systems, web servers, and application software. These flaws are often used by attackers to launch attacks, such as the GitHub attack in 2018 [24, 25]. This was a Memcached DDoS attack without the use of botnets, the attack relied on the amplification

effect of Memcached, a widely-used database caching system. The attackers sent spoofed requests to Memcached servers, amplifying the attack by a factor of 50,000 times.

Conclusion

DDoS attacks can affect any individual or organization that has an online presence. This includes businesses, government agencies, educational institutions, non-profit organizations, and individuals who operate websites, servers, or other online services.

And that's why a robust protection mechanism is required to prevent such attacks. As we have seen before, the conventional methods for detecting DoS/DDoS attacks include log-based approaches which are not very efficient. While Modern approaches like Machine Learning and Artificial Neural Networks have been proven very effective for this task, they still have room for improvements. This research aims to fill those gaps while showing relevant results. To counter the imbalance issue seen in previous studies, this study proposes a novel data collection and data utilization approach. This study also shows comparison between various deep learning model architectures and their configurations on this task. The final trained model was able to accurately identify previously unseen DoS/DDoS packets with an accuracy of 98.95%, which was the highest achieved with this approach. For future work, we aim to propose a multi-model approach for classifying DoS/DDoS attacks in both TCP and UDP networks, as well as incorporating functionality to reduce load during attacks.

References

1. F Yihunie, E Abdelfattah and A Odeh. "Analysis of ping of death DoS and DDoS attacks". in 2018 IEEE Long Island Systems, Applications and Technology Conference, LISAT 2018 (2018).
2. LF Eliyan and R di Pietro. "DoS and DDoS attacks in Software Defined Networks: A survey of existing solutions and research challenges". Future Generation Computer Systems 122 (2021).
3. RR Zebari, SRM Zeebaree and K Jacksi. "Impact Analysis of HTTP and SYN Flood DDoS Attacks on Apache 2 and IIS 10.0 Web Servers". in ICOASE 2018 - International Conference on Advanced Science and Engineering (2018).
4. S Haider., et al. "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks". IEEE Access 8 (2020).
5. J Cheng., et al. "DDoS attack detection via multi-scale convolutional neural network". Computers, Materials and Continua 62.3 (2020).
6. AR Wani., et al. "Analysis and Detection of DDoS Attacks on Cloud Computing Environment using Machine Learning Techniques". in Proceedings - 2019 Amity International Conference on Artificial Intelligence, AICAI 2019 (2019).
7. A Saied, RE Overill and T Radzik. "Detection of known and unknown DDoS attacks using Artificial Neural Networks". Neurocomputing 172 (2016).
8. D Peraković., et al. "Model for detection and classification of DDoS traffic based on artificial neural network". Telfor Journal 9.1 (2017).
9. X Yuan, C Li and X Li. "DeepDefense: Identifying DDoS Attack via Deep Learning". in 2017 IEEE International Conference on Smart Computing, SMARTCOMP 2017 (2017).
10. H Wang, L Xu and G Gu. "FloodGuard: A DoS Attack Prevention Extension in Software-Defined Networks". in Proceedings of the International Conference on Dependable Systems and Networks (2015).
11. S Singh, RA Khan and A Agrawal. "Prevention mechanism for infrastructure based Denial-of-Service attack over software Defined Network". in International Conference on Computing, Communication and Automation, ICCCA 2015 (2015).
12. H Bedi, S Roy and S Shiva. "Mitigating congestion-based denial of service attacks with active queue management". in GLOBECOM - IEEE Global Telecommunications Conference (2013).
13. K Giotis, G Androulidakis and V Maglaris. "A scalable anomaly detection and mitigation architecture for legacy networks via an OpenFlow middlebox". Security and Communication Networks 9.13 (2016).

14. O Yevsieieva and SM Helalat. "Analysis of the impact of the slow HTTP DOS and DDOS attacks on the cloud environment". in 2017 4th International Scientific-Practical Conference Problems of Infocommunications Science and Technology, PIC S and T 2017 - Proceedings, 2017 (2018).
15. Mahadev V Kumar and K Kumar. "Classification of DDoS attack tools and its handling techniques and strategy at application layer". in Proceedings - 2016 International Conference on Advances in Computing, Communication and Automation (Fall), ICACCA 2016 (2016).
16. V Ivanova, T Tashev and I Draganov. "Detection of IoT based DDoS Attacks by Network Traffic Analysis using Feedforward Neural Networks". International Journal of Circuits, Systems and Signal Processing 16 (2022).
17. HS Lamkuche., et al. "SAL - A lightweight symmetric cipher for Internet-of-Things". International Journal of Innovative Technology and Exploring Engineering 8.11 (2019).
18. A Bhardwaj, V Mangat and R Vig. "Hyperband tuned deep neural network with well posed stacked sparse autoencoder for detection of ddos attacks in cloud". IEEE Access 8 (2020).
19. HS Lamkuche and D Pramod. "Csl: Fpga implementation of lightweight block cipher for power-constrained devices". International Journal of Information and Computer Security 12.2-3 (2020).
20. SNS Sarma, HH Lamkuche and S Umamaheswari. "A review of secret sharing schemes". Research Journal of Information Technology 5.2 (2013).
21. KW Church. "Word2Vec". Nat Lang Eng 23.1 (2017).
22. I Sutskever, O Vinyals and Qv Le. "Sequence to sequence learning with neural networks". in Advances in Neural Information Processing Systems 4 (2014).
23. A Ramesh, V Pradhan and H Lamkuche. "Understanding and Analysing Resource Utilization, Costing Strategies and Pricing Models in Cloud Computing". in Journal of Physics: Conference Series 1964.4 (2021).
24. S Kumar, D Kumar and HS Lamkuche. "TPA Auditing to Enhance the Privacy and Security in Cloud Systems". Journal of Cyber Security and Mobility 10.3 (2021).
25. HS Lamkuche., et al. "Enhancing the security and performance of cloud for e-governance infrastructure: Secure E-MODI". International Journal of Cloud Applications and Computing 12.1 (2022).
26. Lamkuche HS, Singh K and Shirkhedkar K. A lightweight block cipher for cloud-based healthcare systems. In Computing, Communication and Learning: First International Conference, CoCoLe 2022, Warangal, India, 27-29 October 2022, Proceedings, Springer, Cham (2023): 3-14.

Volume 6 Issue 3 March 2024