ESPIAL OF APPLICATION-LAYER DDoS ATTACK USING MACHINE LEARNING

Project Reference No.: *47S_BE_4215*

College ***:*** *Don Bosco Institute of Technology, Bengaluru*
Branch ***:*** *Department of Information and Science Engineering*
Guide(s) ***:*** *Mrs. Deepika A. B*
Student(S) ***:*** *Ms. Bhoomika K.*
Ms. Kavanashree S.
Ms. Bhoomika B. S.
Ms. Kavana G. P

Keywords: Machine Learning, DDoS Attacks, Deep Learning.

Introduction

Distributed Denial of Service (DDoS) attacks are a growing threat to online services, and various methods have been developed to detect them. However, past research has mainly focused on identifying attack patterns and types, without specifically addressing the role of freely available DDoS attack tools in the escalation of these attacks. A DDoS attack is a malevolent effort to stop a specific website, computer, or network from operating normally by saturating it with traffic from numerous sources. In this kind of attack, the perpetrator employs a network of the associate editor coordinating the review of this manuscript and approving it for publication was Giovanni Pau. computers or other devices (referred to as a ‘‘botnet’’) to overwhelm the target system with an excessive quantity of data, rendering it inaccessible to authorized users. DDoS attacks are complex attacks since (1) they can generate a large volume of traffic from a wide variety of sources, and (2) the traffic appears to originate from a wide variety of locations. DDoS attacks manifest in diverse forms, with application layer attacks being one of them.

Application-layer DDoS attacks target the application layer of the victim system, aiming to exhaust its resources or cause the application to fail. Illustrative examples of such attacks include HTTP floods and Slowloris attacks. The primary goal of application-layer DDoS attacks is to disable a network by overwhelming it with traffic, leading to system crashes or unavailability.

Objectives

- Our aim is to analyze the impact of easy access to such tools on the

frequency and severity of DDoS attacks, and to explore potential solutions for detecting this threat.

- The solution aims to identify and distinguish between traffic produced by four freely accessible tools and legitimate traffic.
- To further enhance the system's speed and efficiency, a thorough feature selection technique, which is resulted in a significant reduction in the feature subset.
- To developing a more efficient and accurate method for detecting application-layer DDoS attacks.
- To improving the speed of detection for application-layer DDoS attacks.

Methodology

Most DDoS detection and mitigation methods to date have focused on specific attack types, such as HTTP floods, FTP floods, website spider attacks, asymmetric attacks, and slow header attacks. However, based on our extensive research in academia and industry, we have not found any evidence of researchers investigating the variety and accessibility of DDoS toolkits. In this study, we use MLP to classify 4 DDoS attacks and benign traffic with 6 features only. To ensure a systematic approach to network classification, we adopt a stepwise strategy in selecting theappropriate number of neurons and hidden layers. Specifically, the procedure begins with no hidden layers, and considers the accuracy, precision, recall, and F1 score as performance metrics.

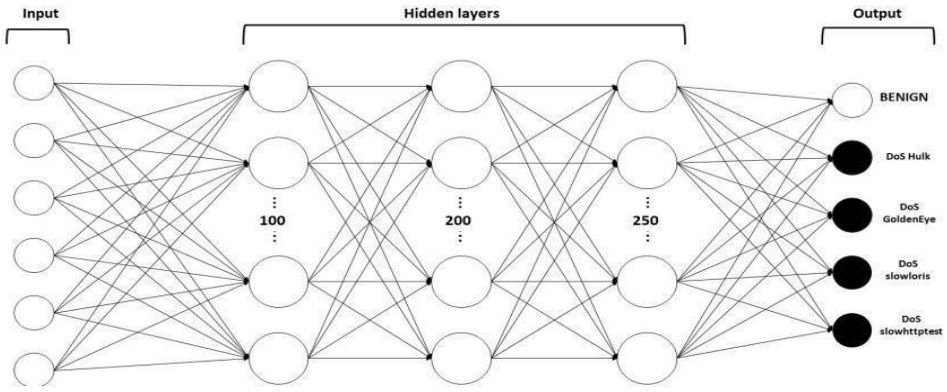


Figure 1.1: MLP Architecture.

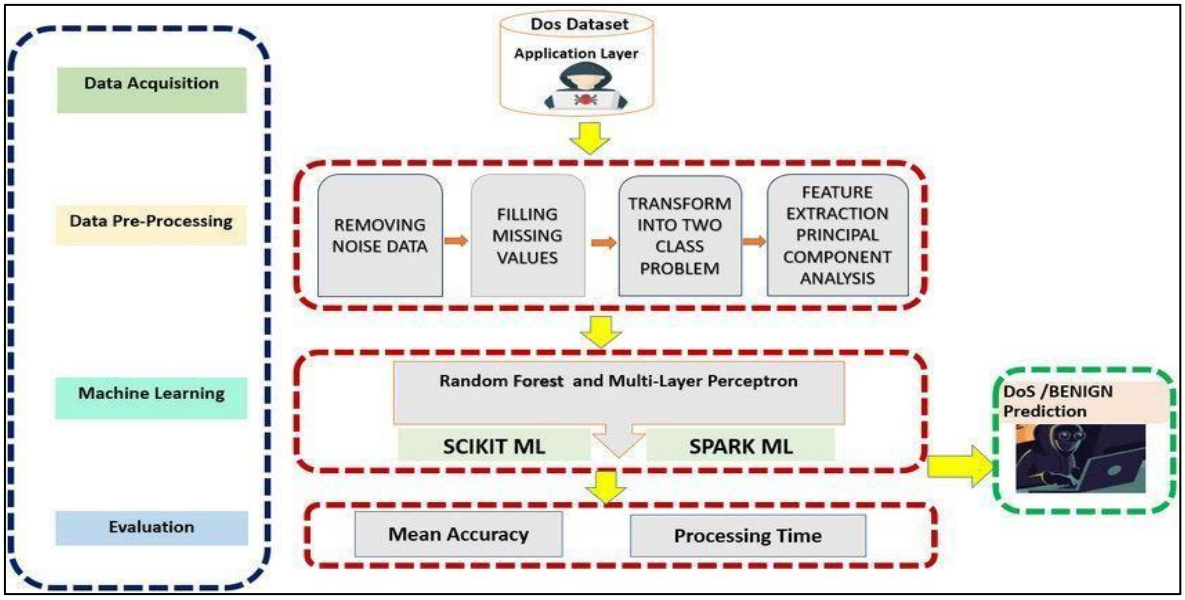
The Figure 1.1 represents the multilayer perceptron architecture represents the inputs with the hidden layer by taking the inputs and giving the output.

Figure 1.2: System Architecture for predicting DoS and DDoS attacks.

Figure 1.2 present the proposed approach. In this work, we propose a MLP to detect 5 classes of which 4 are DoS/DDoS attack types and one is the benign traffic. We start with a representative dataset, Nbaiot and NSL-KDD.

Software Requirement:

- Operating System: Windows 64-bit
- Technology: Python
- IDE: Python IDLE
- Tools: Anaconda
- Python Version: Python 3.6



Results and Conclusion:

Distributed denial of service attacks (DDoS) has the potential to disable essential online services and prevent Users from accessing them. DDoS attack tools are of worry to the defence community for two reasons: (1) they are cheap or even free and easy to find online, and (2) hackers frequently employ them because they can be deployed with little in the way of technological expertise. According to our research, ML can be utilized to successfully classify DDoS tools into a total of five different categories; four of these classes pertain to DDoS tool attacks, and the remaining class contains innocuous data. By establishing a fast and reliable technique for selecting the features, we were able to cut the total number of options available in our model from 78 to 6. Our model has a high level of performance

across the board, with a 99.2% accuracy, 97.1% precision, 96.1% recall, and 96.6% F1 score when using Adam as the optimizer.

Title: Application layer DDoS attack detection using cuckoo search algorithm-trained radial basis function Author: H. Beitollahi, D. M. Sharif, and M. Fazeli, Year: 2022.

Researchers have suggested several techniques for detecting App-DDoS traffic. There is, however, no clear distinction between legitimate and attack traffic. In this paper, we go a step further and propose a Machine Learning (ML) solution by combining the Radial Basis Function (RBF) neural network with the cuckoo search algorithm to detect App-DDoS traffic. We begin by collecting training data and cleaning them, then applying data normalizing and finding an optimal subset of features using the Genetic Algorithm (GA). Next, an RBF neural network is trained by the optimal subset of features and the optimizer algorithm of cuckoo search. Finally, we compare our proposed technique to the well-known k-nearest neighbor (k-NN), Bootstrap Aggregation (Bagging), Support Vector Machine (SVM), Multi-layer Perceptron) MLP, and (Recurrent Neural Network) RNN methods. Our technique outperforms previous standard and well-known ML techniques as it has the lowest error rate according to error metrics. Moreover, according to standard performance metrics, the results of the experiments demonstrate that our proposed technique detects App-DDoS traffic more accurately than previous techniques.

Description of the innovation in the project:

Our project not only detects DoS and DDoS attacks, it can also predict other types of attacks such as Benign, ping of death etc., Since we have used tow datasets NBaIoT and NSL-KDD where these datasets contains data of other such attacks. This has made our project even more accurate and précised model to use in industries to detect harmful attacks and predict them in early stage. As Dos and DDoS are done mostly in the application of the network or models like OSI or TCP, they are very difficult to predict. But our model has overcome this difficulty by achieving 98-99% of accuracy in predicting the attacks.

Future work scope:

In the future, the detection of application layer DDoS attacks using machine learning holds significant potential for advancement. Through ongoing research and development efforts, there is a clear trajectory toward improving detection accuracy by refining algorithms and incorporating diverse data sources. Behavioural analysis techniques are expected to become more sophisticated, enabling systems to better differentiate between legitimate user traffic and malicious activities. Real-time adaptation will be a key focus, with machine

Slearning models dynamically adjusting to evolving attack strategies and traffic patterns. Additionally, a multi-layered defence approach, integrating machine learning with other detection methods, will provide comprehensive protection against a wide range of DDoS threats. Integration with SDN and NFV technologies will facilitate more flexible and scalable deployment, while privacy- preserving techniques will address growing concerns about data privacy. When constructing a predictive model, one of the first steps is feature selection, which is the process of choosing the optimal feature subset. It is desired to limit the number of input features in order to increase the performance of the model and, in certain situations, to lower the computational cost of modeling. In our work, we use DT for the purpose of feature selection. We populate all the data in the dataset to the DT then we consider the importance of features. The importance of a feature is determined as the (normalized) total reduction of the criteria that the feature contributes. It is also referred to as the Gini importance. After that, we take the mean of feature importance of all features and use it as a threshold. Any feature with feature importance below the threshold is discarded. The importance of each feature is then normalized and ranked, so that the most important feature has a score of 1, and the least important feature has a score of 0.