e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:08/August-2022 Impact Factor- 6.752 www.irjmets.com

DDOS ATTACK DETECTION USING SUPERVISED MACHINE LEARNING

Neetu Bala[*1], **Gurjit Kaur**[*2]

[*1]Computer Science Engineering, Sant Baba Bhag Singh University, Jalandhar, Punjab, India.

[*2]Assistant Professor, Computer Science Engineering, Sant Baba Bhag Singh University, Jalandhar, Punjab, India.

ABSTRACT

Websites can be typical if they are often accessed by a lot of people or if they offer some helpful information. Once many people have access to it, the servers are frequently overloaded, which might result in an attack. The rapid advancement of technology has resulted in an increase in hacking cases. Threats of all shapes and sizes emerge daily with the intent of blocking user access to services. The quantity of traffic a DDoS assault sends to the host system per second can be used to identify it. For example, modest attacks may only send a few megabits (Mpbs), whereas massive attacks may send a terabit per second (Tbps). Attackers employ the botnet for high-volume attacks in order to carry them out effectively. If it happens infrequently, the server won't reply to the authorized user. Network administrators use this tool to protect internet devices from attack by comparing and recording incoming packet traffic with traffic signatures.

Keywords: Attack, Ddos Attack, Machine Learning, Types Of Ddos Attack, Types Of Machine Learning Algorithms.

I. INTRODUCTION

A denial of service (DoS) attack aims to disable network resources by overwhelming the service's host, rendering it unusable to the service's legitimate users. A distributed denial of service (DDoS) attack is a DoS attack that comes from several different sources. DoS attacks are often started from a single infected computer or virtual machine utilising an Internet connection, whereas DDoS assaults start from multiple compromised computers or virtual machines to overwhelm victims' networks. DDoS attacks include concurrently sending numerous requests to the target's exhausted computational resource using botnets and hijacked IoT devices (Bandwidth and Traffic).The infected computers, also known as bots or zombies, are controlled remotely by one or more bot-masters and engage in botnet attacks as shown in Figure 1. Bots might be infected legitimate individuals or malevolent users who are preparing an assault.

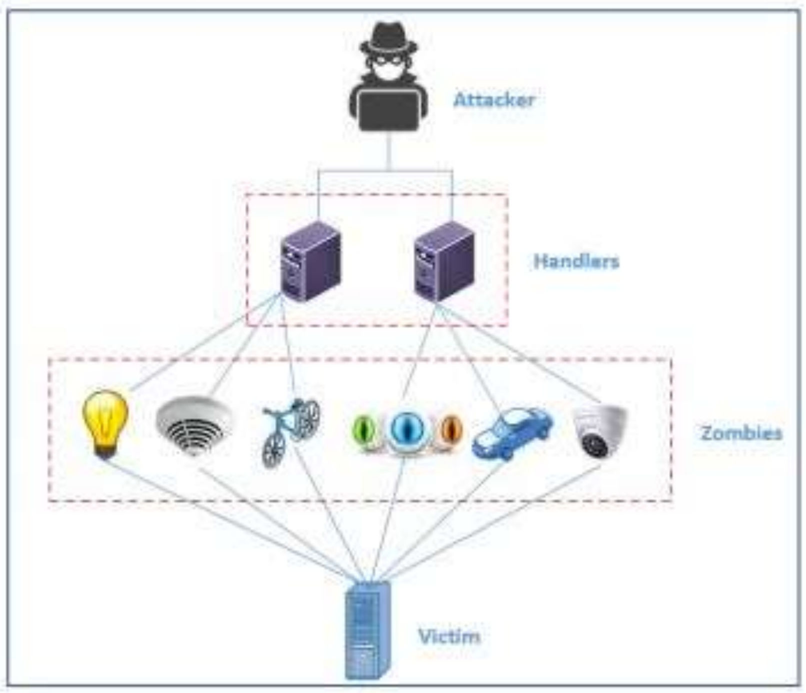


Fig.1: DDOS attack network infrastructure

TYPES OF DDOS ATTACKS

This harmful assault utilizes a number of globally hacked services. The three main types of DDoS assaults are as follows. They are

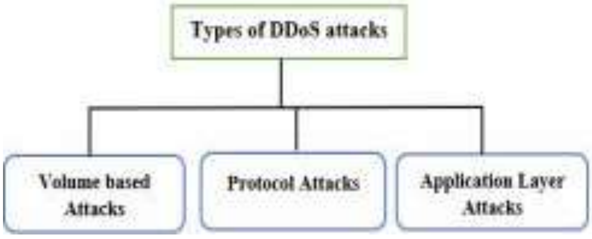


Fig. 2: DDOS attack types

Volume based Attacks: It remains one of the most frequent attacks even now. This assault uses the available network's bandwidth to send out enormous amounts of data. This causes a congestion, which causes an acceleration, which requests to send a lot of traffic to the victim's services, which will slow down responses and choke the server.

Protocol Attacks: Protocol attacks, often referred to as state exhaustion attacks, have an effect on the network layer of the victim's system. Table spaces that pass requests to the target, such firewalls and load balancers, are hindered by this.

Application Layer Attacks: Interruption of the victim's website or service by a huge number of requests before the submitted requests have reached their final stage of exhaustion large data downloads or queries from databases are two examples of requests. When millions of requests are made to the same target at once, the device operation may sluggish or even shut down automatically.

CHALLENGES WITH DDOS ATTACKS

Due to the complexity of DoS assaults, it is impossible to pinpoint the precise IP address from which they were launched. It is feasible to fake its source and use many systems (nodes) as sources while attacking them over the internet from different places. Its source is frequently hacked to carry out assaults as well. The kind of assault is extremely sophisticated because it involves human interaction and has several sources. When a DDoS assault affects any device in 2017, there is a dearth of quick, precise solutions that can exacerbate problems with financial processing power.

MACHINE LEARNING

Machine learning is the capacity of a system to learn from experience and advance without extensive programming. The techniques of machine learning aid in the effective resolution of numerous issues in the field of IT. It is now fashionable and aids in resolving significant network security challenges. It is narrowly categorized using a variety of learning techniques.

CLASSIFICATION OF MACHINE LEARNING ALGORITHMS

Every task in Machine learning is broadly classified into several categories. They are

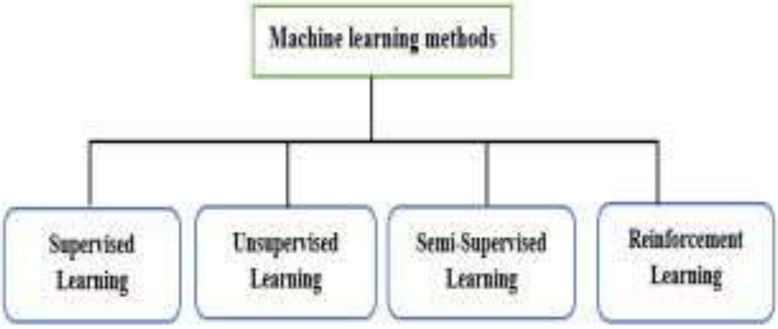


Fig. 3: Types of machine learning

Supervised Learning Techniques: The goal is to gather data and learn from it using different supervised machine learning algorithms. It is a mathematical model used for predictive modelling that consists of a set of inputs (labelled inputs) and intended outcomes. Nearest Neighbor, Decision Tree, Support Vector Machines, Naive Bayes, and Linear Regression are some examples of frequently used algorithms.

Unsupervised Learning Techniques: The system learns from the unlabelled data in this way, then generates the output. This aids in selecting the crucial data needed for the analyses. We employ the unsupervised learning method if we need any details about how a set of data is related. There are many clustering algorithms, including k-means clustering and association rules.

Semi-supervised learning Techniques: We will use both labelled and unlabelled data and combine supervised and unsupervised learning techniques. It generates the desired outcome with crucial parameters needed for studies.

Reinforcement Learning Techniques: Reinforcement training is based on trial and error method for a particular decision. It gains experiences from the previous trainings and gives accurate knowledge based on the response received.

II. METHODOLOGY

Logistic Regression

One of the most often used Machine Learning algorithms, within the category of Supervised Learning, is logistic regression. Using a predetermined set of independent factors, it is used to predict the categorical dependent variable. In a categorical dependent variable, the output is predicted via logistic regression. As a result, the result must be a discrete or categorical value. Rather of providing the exact values of 0 and 1, it provides the probabilistic values that fall between 0 and 1. It can be either Yes or No, 0 or 1, true or false, etc. With the exception of how they are applied, logistic regression and linear regression are very similar. While logistic regression is used to solve classification difficulties, linear regression is used to solve regression problems. In logistic regression, we fit a "S" shaped logistic function, which predicts two maximum values, rather than a regression line (0 or 1). The logistic function's curve shows the possibility of several things, including whether or not the cells are malignant, whether or not a mouse is obese depending on its weight, etc. Because it can classify new data using both continuous and discrete datasets, logistic regression is a key machine learning approach. Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification.

Decision Tree

A supervised learning method called a decision tree can be used to solve classification and regression problems, but it is typically favoured for doing so. It is a tree-structured classifier, where internal nodes stand in for the dataset's features, branches for the decision-making processes, and each leaf node for the result.

The Decision Node and Leaf Node are the two nodes of a decision tree. While Leaf nodes are the results of decisions and do not have any more branches, Decision nodes are used to create decisions and have numerous branches.

The given dataset's features are used to execute the test or make the decisions.

It is a graphical depiction for obtaining all feasible answers to a choice or problem based on predetermined conditions.

It is known as a decision tree because, like a tree, it begins with the root node and grows on subsequent branches to form a structure resembling a tree.

The CART algorithm, which stands for Classification and Regression Tree algorithm, is used to construct a tree.

A decision tree only poses a question and divides the tree into sub-trees according to the response (Yes/No).

Below diagram explains the general structure of a decision tree:

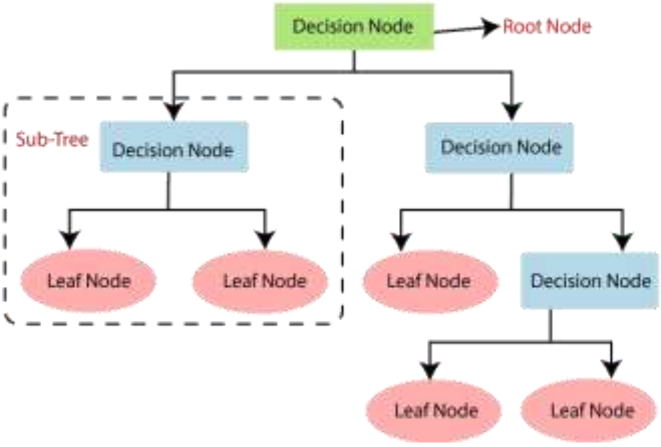


Fig. 4: Decision Tree

Why use Decision Trees?

- The most important thing to keep in mind while developing a machine learning model is to select the optimal method for the dataset and task at hand. The two rationales for employing the decision tree are as follows:
- Decision trees are typically designed to resemble how people think when making decisions, making them simple to comprehend.
- The decision tree's reasoning is clear because it has a tree-like structure.

Decision Tree Terminologies

Root Node: The decision tree begins at the root node. The full dataset is represented, which is then split into two or more homogeneous sets.

Leaf Node: Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.

Splitting: Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.

Branch/Sub Tree: A tree formed by splitting the tree.

Pruning: Pruning is the process of removing the unwanted branches from the tree.

Parent/Child node: The root node of the tree is called the parent node, and other nodes are called the child nodes.

How does the Decision Tree algorithm Work?

In a decision tree, the algorithm begins at the root node and works its way up to forecast the class of the given dataset. This algorithm follows the branch and jumps to the following node by comparing the values of the root attribute with those of the record (real dataset) attribute.

The algorithm verifies the attribute value with the other sub-nodes once again for the following node before continuing. It keeps doing this until it reaches the tree's leaf node. The following algorithm can help you comprehend the entire procedure:

- **Step-1:** Begin the tree with the root node, says S, which contains the complete dataset.
- **Step-2:** Find the best attribute in the dataset using **Attribute Selection Measure (ASM).**
- **Step-3:** Divide the S into subsets that contains possible values for the best attributes.
- **Step-4:** Generate the decision tree node, which contains the best attribute.
- **Step-5:** Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

K Neighbours Classifier

○ K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

○ The K-NN algorithm makes the assumption that the new case and the existing cases are comparable, and it places the new instance in the category that is most like the existing categories.

○ The K-NN algorithm saves all the information that is accessible and categorises fresh input based on similarity. This means that utilising the K-NN method, fresh data can be quickly and accurately sorted into a suitable category.

○ The K-NN approach can be used for both classification and regression problems, but it is more frequently utilised for classification issues.

○ Because K-NN is a non-parametric method, it makes no assumptions about the underlying data.

○ It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

○ KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Example: Let's say we have a picture of a species that resembles both cats and dogs, but we aren't sure if it is one or the other. Therefore, since the KNN algorithm is based on a similarity metric, we can utilise it for this identification. Our KNN model will look for similarities between the new data set's features and those in the photos of cats and dogs, and based on those similarities, it will classify the new data set as either cat- or dog-related.

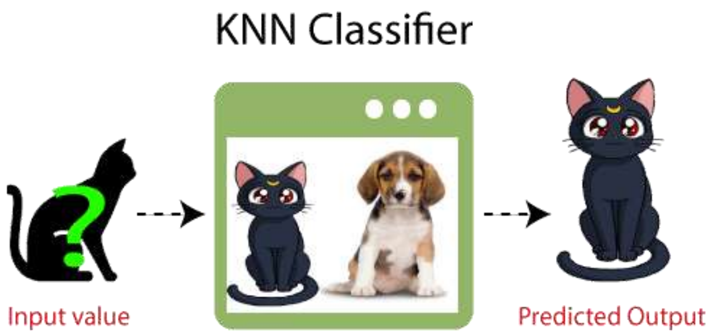
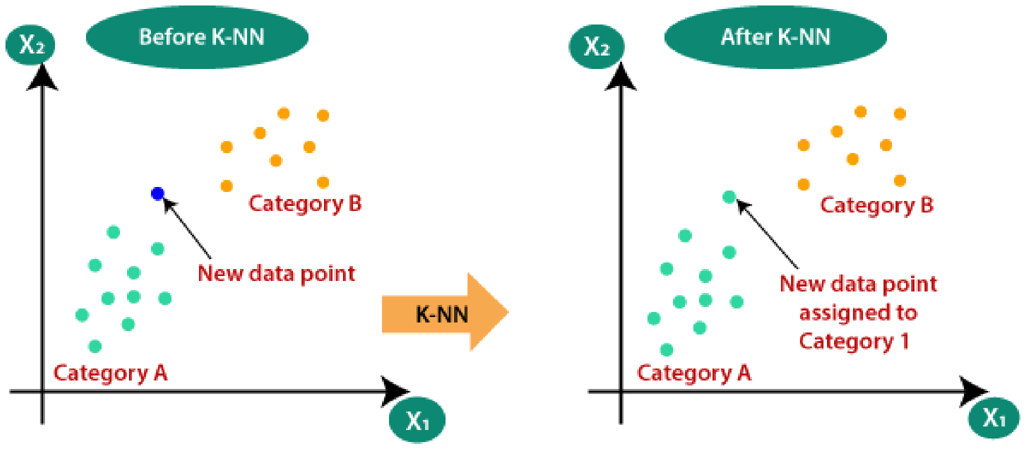


Fig. 5: KNN classifier

Why do we need a K-NN Algorithm?

If there are two categories, Category A and Category B, and we have a new data point, x1, which category does this data point belong in? We require a K-NN algorithm to address this kind of issue. K-NN makes it simple to determine the category or class of a given dataset. Take a look at the diagram below:

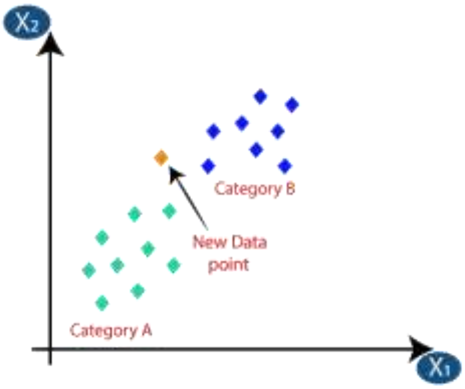


How does K-NN work?

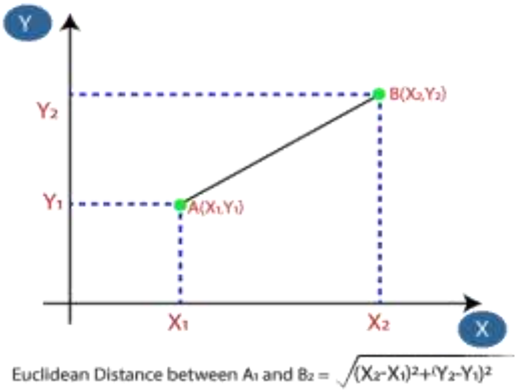
The K-NN working can be explained on the basis of the below algorithm:

- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance of **K number of neighbors**
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
- **Step-6:** Our model is ready.

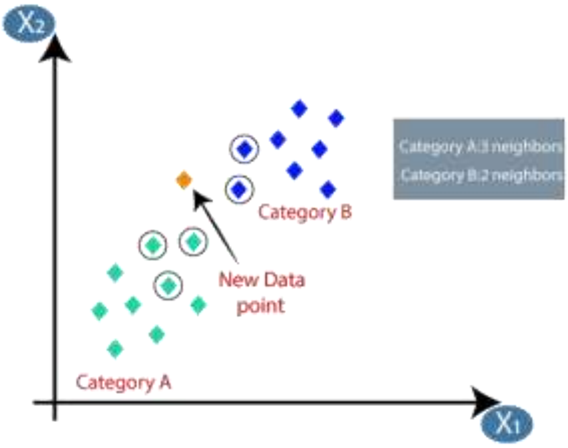
Suppose we have a new data point and we need to put it in the required category. Consider the below image:



- First, we'll decide on the number of neighbors, thus we'll pick k=5.
- The Euclidean distance between the data points will then be calculated. The distance between two points, which we have already examined in geometry, is known as the Euclidean distance. It is calculable as:



- By calculating the Euclidean distance we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B. Consider the below image:



- As we can see the 3 nearest neighbours are from category A, hence this new data point must belong to category A.

Advantages of KNN Algorithm:

- It is simple to implement.
- It is robust to the noisy training data
- It can be more effective if the training data is large.

Disadvantages of KNN Algorithm:

- Always needs to determine the value of K which may be complex some time.
- The computation cost is high because of calculating the distance between the data points for all the training samples.

MLP Classifier

The most complicated architecture of artificial neural networks is defined by the multi-layer perceptron. It is largely constructed from several perceptron layers. This notebook will walk you through creating a neural network using the widely used deep learning framework TensorFlow. We must use Numpy to create a multi-layer perceptron from scratch in order to comprehend what a multi-layer perceptron is. Below is a visual illustration of multi-layer perceptron learning.-

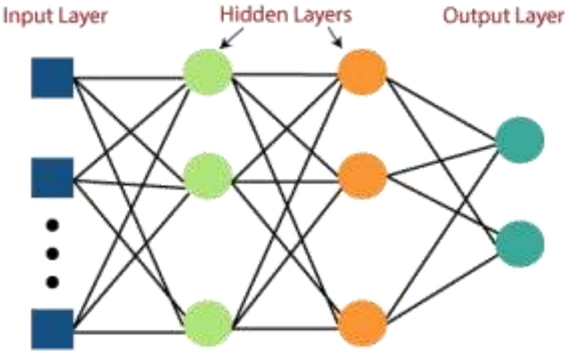


Fig. 6: multilayer perceptron

To implement supervised learning, MLP networks are employed. Back propagation's algorithm is another name for a typical learning algorithm for MLP networks.

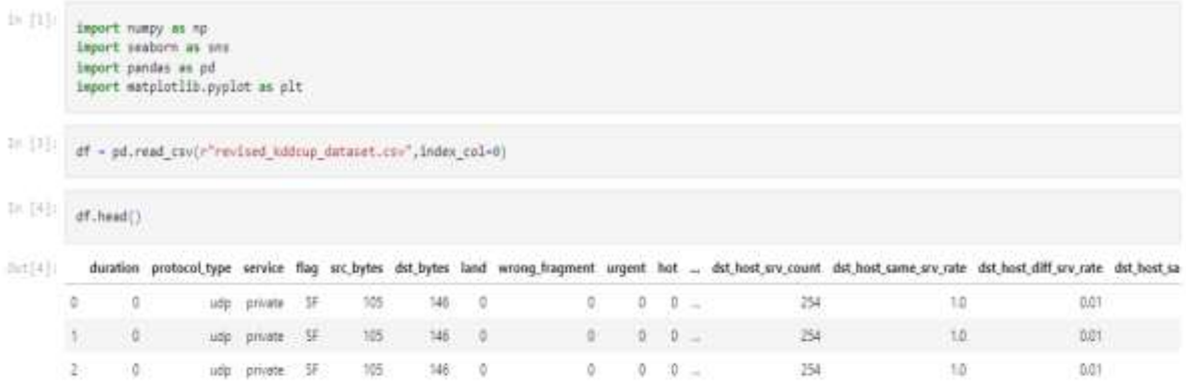
A feed-forward artificial neural network that produces a set of outputs from a collection of inputs is called a multilayer perceptron (MLP). A directed graph connecting the input nodes in multiple layers of input nodes connected between the input and output a layer is an MLP's defining feature. Back propagation is used by MLP to train the network. A deep learning technique is MLP.

III. RESULT AND DISCUSSION

1)UDP Flood 2)TCP_SYN Flood 3)ICMP Flood

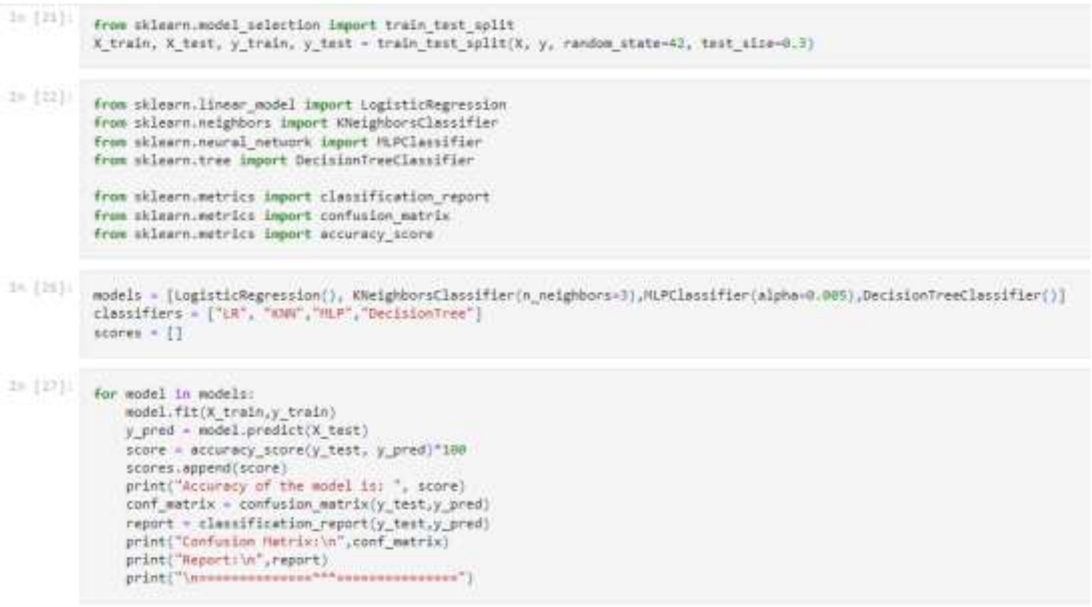
1. UDP FLOOD

Import libraries and read csv file



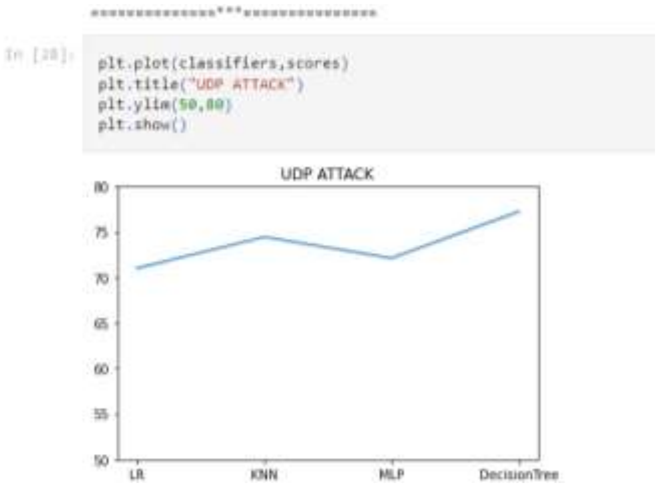
```
In [1]: import numpy as np
        import seaborn as sns
        import pandas as pd
        import matplotlib.pyplot as plt

In [3]: df = pd.read_csv(r"revised_kddcup_dataset.csv",index_col=0)

In [4]: df.head()

Out[4]:
   duration protocol_type service flag src_bytes dst_bytes land wrong_fragment urgent hot ... dst_host_srv_count dst_host_same_srv_rate dst_host_diff_srv_rate dst_host_sa
0         0           udp private   SF       105       146    0              0      0   0 ...                254                    1.0                   0.01
1         0           udp private   SF       105       146    0              0      0   0 ...                254                    1.0                   0.01
2         0           udp private   SF       105       146    0              0      0   0 ...                254                    1.0                   0.01
```

Import, train and test models



```
In [21]: from sklearn.model_selection import train_test_split
         X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=42, test_size=0.3)

In [22]: from sklearn.linear_model import LogisticRegression
         from sklearn.neighbors import KNeighborsClassifier
         from sklearn.neural_network import MLPClassifier
         from sklearn.tree import DecisionTreeClassifier

         from sklearn.metrics import classification_report
         from sklearn.metrics import confusion_matrix
         from sklearn.metrics import accuracy_score

In [26]: models = [LogisticRegression(), KNeighborsClassifier(n_neighbors=3),MLPClassifier(alpha=0.005),DecisionTreeClassifier()]
         classifiers = ["LR", "KNN","MLP","DecisionTree"]
         scores = []

In [27]: for model in models:
             model.fit(X_train,y_train)
             y_pred = model.predict(X_test)
             score = accuracy_score(y_test, y_pred)*100
             scores.append(score)
             print("Accuracy of the model is: ", score)
             conf_matrix = confusion_matrix(y_test,y_pred)
             report = classification_report(y_test,y_pred)
             print("Confusion Matrix:\n",conf_matrix)
             print("Report:\n",report)
             print("\n==============***==============")
```

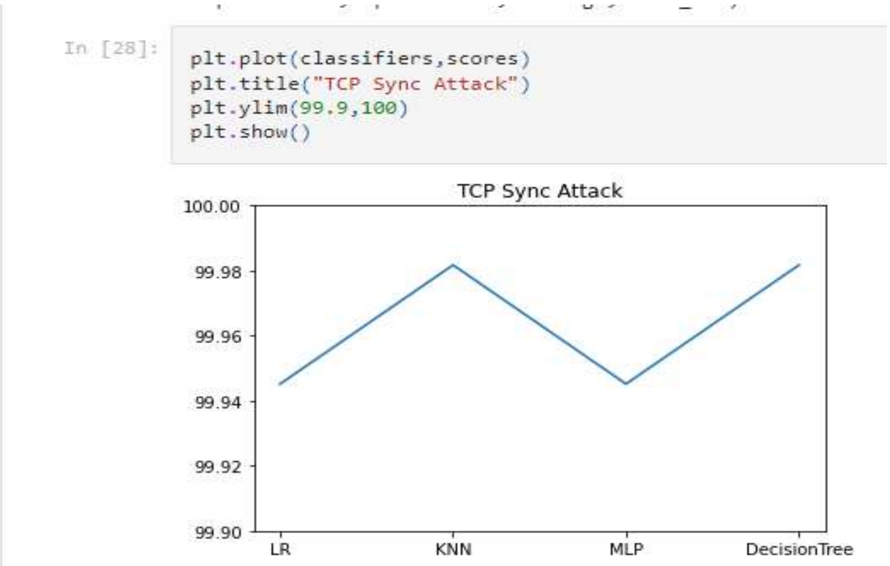
Models accuracy score



```
==============***==============

In [28]: plt.plot(classifiers,scores)
         plt.title("UDP ATTACK")
         plt.ylim(50,80)
         plt.show()
```

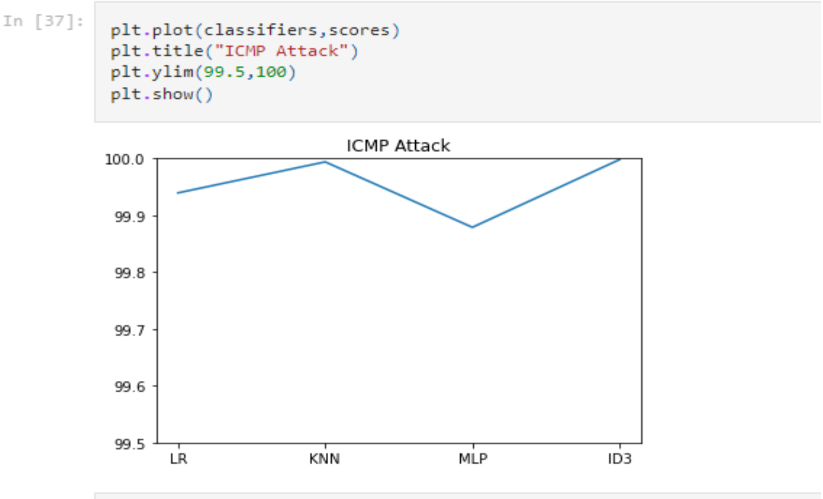
2. TCP_SYN flood

Model accuracy score



```
In [28]: plt.plot(classifiers,scores)
         plt.title("TCP Sync Attack")
         plt.ylim(99.9,100)
         plt.show()
```

3. ICMP flood



```
In [37]: plt.plot(classifiers,scores)
         plt.title("ICMP Attack")
         plt.ylim(99.5,100)
         plt.show()
```

IV. CONCLUSION

Attacks are when a system's regular behavior is disrupted or damaged due to the employment of various methods and tactics to take advantage of vulnerabilities. Attacks have unique motivations for many different reasons. One sort of attack monitors unencrypted network traffic in an aggressive manner to uncover sensitive data. Another type of attack is the passive attack, which scans weakly encrypted communication for authentication information. The most frequent types of assaults are control attacks, physical attacks, distributed denial of service attacks, privacy attacks such password base attacks, cyber espionage, and eavesdropping.

V. REFERENCES

[1] N. Ravi and S. M. Shalinie, "Learning-Driven Detection and Mitigation of DDoS Attack in IoT via SDN-Cloud Architecture," in IEEE Internet of Things Journal, vol. 7, no. 4, pp. 3559-3570, April 2020, doi: 10.1109/JIOT.2020.2973176.

[2] K. Huang, L. Yang, X. Yang, Y. Xiang and Y. Y. Tang, "A Low-Cost Distributed Denial-of-Service Attack Architecture," in IEEE Access, vol. 8, pp. 42111-42119, 2020, doi: 10.1109/ACCESS.2020.2977112.

[3] S. Velliangiri, P. Karthikeyan& V. Vinoth Kumar (2020) Detection of distributed denial of service attack in cloud computing using the optimization-based deep networks, Journal of Experimental & Theoretical Artificial Intelligence, DOI: 10.1080/0952813X.2020.1744196

[4] Asad, Mohammad &Khrais, Rami &Yateem, A.Rahman. (2020). DoS and DDoS Attack Detection Using Deep Learning and IDS. International Arab Journal of Information Technology. 17. 655-661. 10.34028/iajit/17/4A/10.

[5] Dwivedi, Shubhra & Vardhan, Manu &Tripathi, Sarsij. (2020). Defense against distributed DoS attack detection by using intelligent evolutionary algorithm. International Journal of Computers and Applications.1-11. 10.1080/1206212X.2020.1720951.

[6] Tuan, N.N.; Hung, P.H.; Nghia, N.D.; Tho, N.V.; Phan, T.V.; Thanh, N.H. A DDoS Attack Mitigation Scheme in ISP Networks Using Machine Learning Based on SDN. Electronics **2020**, 9, 413.

[7] T. V. Phan and M. Park, "Efficient Distributed Denial-of-Service Attack Defense in SDN-Based Cloud," in IEEE Access, vol. 7, pp. 18701-18714, 2019, doi: 10.1109/ACCESS.2019.2896783.

[8] Khalaf, Bashar & Mostafa, Salama & Mohammed, Mazin & Abduallah, Wafaa & Mustapha, Aida. (2019). Comprehensive Review of Artificial Intelligence and Statistical Approaches in Distributed Denial of Service Attack and Defense Methods. IEEE Access. PP. 2169-3536. 10.1109/ACCESS.2019.2908998.

[9] Sahoo, K.S., Panda, S.K., Sahoo, S. et al. Toward secure software-defined networks against distributed denial of service attack. JSupercomput 75, 4829–4874 (2019).

[10] https://doi.org/10.1007/s11227-019-02767-z

[11] Saritha et al. “Prediction of DDoS Attack susing Machine Learning and Deep Learning Algorithms.” (2019).