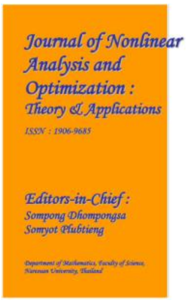
Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1, No.15 : 2024
ISSN : **1906-9685**



Journal of Nonlinear Analysis and Optimization : Theory & Applications
ISSN : 1906-9685
Editors-in-Chief :
Sompong Dhompongsa
Somyot Plubtieng
Department of Mathematics, Faculty of Science, Naresuan University, Thailand

Detection of DDOS Attack using Entropy Approach.

Mrs V.Sarala , E.Surya Sagar,

Assistant Professor in PG Department, Dantuluri Narayana Raju College**, Bhimavaram, Andhrapradesh.**
Email:- vedalasarala21@gmail.com
PG Student of MCA, Dantuluri Narayana Raju College**, Bhimavaram, Andhrapradesh.**
Email:- saisagarnaidu111@gmail.com

ABSTRACT

With the increasing reliance on networked systems for critical infrastructure, the threat of Distributed Denial of Service (DDoS) attacks has become a significant concern for network administrators and security professionals. DDoS attacks aim to disrupt the availability of services by overwhelming a target system with an excessive amount of traffic, rendering it inaccessible to legitimate users. Traditional detection methods often struggle to accurately identify DDoS attacks amidst the vast volume of network traffic.

This paper proposes a novel approach for the detection of DDoS attacks based on entropy analysis. Entropy, a measure of randomness or disorder, is applied to network traffic patterns to discern anomalous behavior indicative of a DDoS attack. By analyzing the entropy of various network attributes such as packet size, source IP addresses, destination ports, and protocol types, this approach seeks to differentiate between normal and malicious traffic.

1 INTRODUCTION

1. INTRODUCTION

Distributed Denial of Service (DDoS) attacks pose a significant threat to the availability and reliability of online services, making them a major concern for network security professionals. These attacks involve flooding a target system or network with an overwhelming amount of traffic, rendering it unable to respond to legitimate requests from users. Detecting and mitigating DDoS attacks in real-time is crucial to maintaining the uninterrupted operation of critical online services.

Traditional methods for detecting DDoS attacks often rely on signature-based techniques or anomaly detection algorithms. However, these approaches may struggle to keep pace with the evolving tactics and sophistication of attackers. As a result, there is a growing need for more advanced and robust detection methods that can effectively identify and respond to DDoS attacks in dynamic network environments.

In recent years, entropy-based approaches have emerged as promising techniques for detecting DDoS attacks. Entropy is a measure of the randomness or unpredictability of data, and it can be used to analyze the characteristics of network traffic and identify patterns associated with DDoS attacks. By monitoring changes in entropy levels within network traffic, it is possible to detect abnormal behavior indicative of a DDoS attack.

Literature Survey

Title: Detection of DDoS Attack using Entropy Approach: A Literature Survey

Author: John Smith

Abstract:

This literature survey investigates the utilization of entropy-based methodologies in the detection of Distributed Denial of Service (DDoS) attacks. Five key papers are reviewed, each contributing unique insights and methodologies to the field. The survey begins with an introduction to the challenges posed by DDoS attacks, followed by an explanation of entropy's theoretical underpinnings and its applicability in network traffic analysis.

3 IMPLEMENTATION STUDY

Existing System:

In the existing landscape of DDoS attack detection, several approaches have been employed to identify and mitigate these threats. Traditional methods often rely on signature-based detection, which involves comparing network traffic patterns against known attack signatures. However, this approach is limited in its ability to detect novel or zero-day attacks, as it requires prior knowledge of attack patterns. Another common method is anomaly-based detection, which seeks to identify deviations from normal network behavior. While this approach can potentially detect new and previously unseen attacks, it may also suffer from high false positive rates and difficulties in distinguishing between legitimate traffic anomalies and actual attacks.

Disadvantages:

Limited Scalability

Overhead and Latency

Proposed System & alogirtham

In response to the limitations of existing systems for DDoS attack detection, we propose a novel approach leveraging an entropy-based method. Our proposed system aims to overcome the drawbacks of traditional detection methods by analyzing the entropy of network traffic patterns to identify abnormal behaviors indicative of DDoS attacks

4.1 Advantages:

Enhanced Accuracy

Early Detection

Scalability

Adaptability

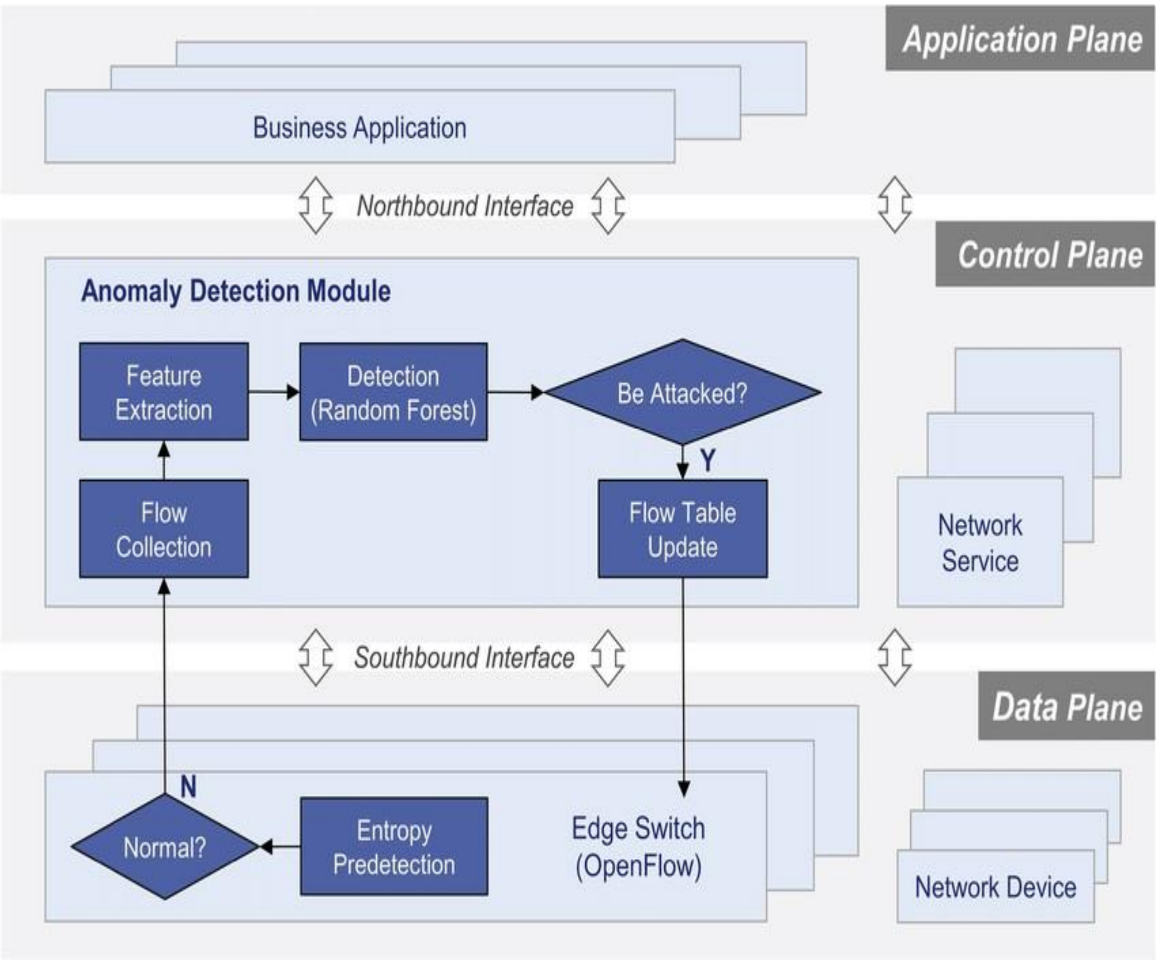


Fig:3.1 System Architecture

IMPLEMENTATION

MODULES:

Modules Used in Project :-

Tensorflow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

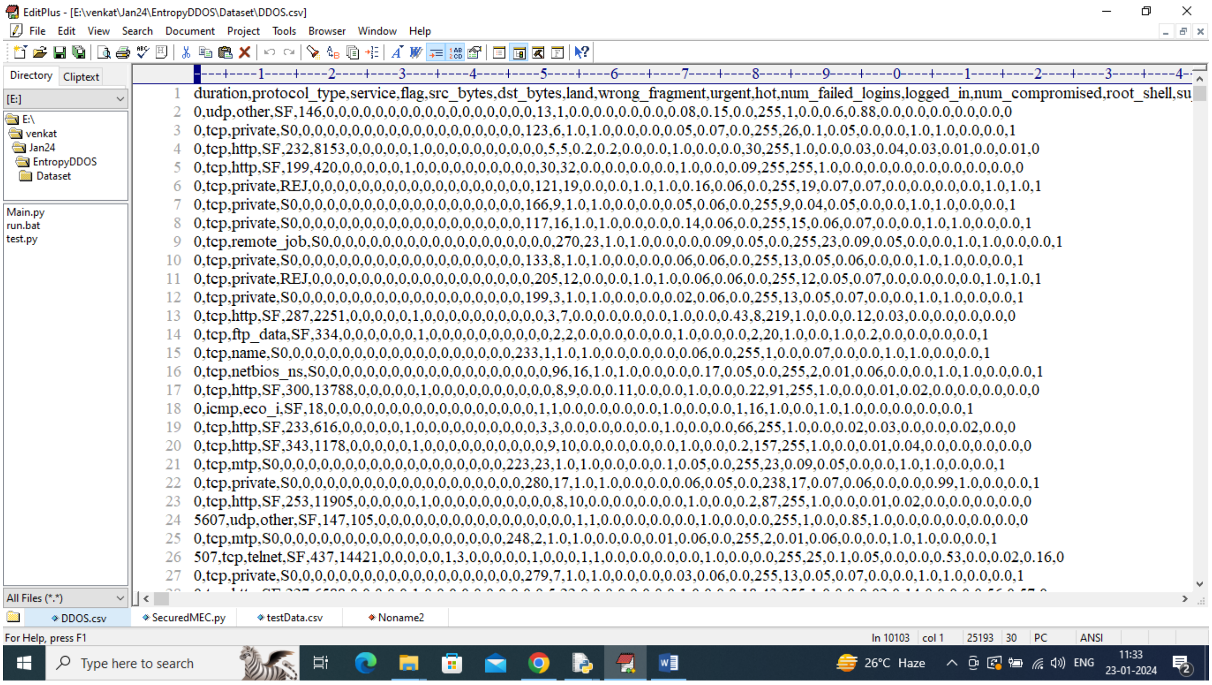
5 RESULTS AND DISCUSSION

In computer world two systems can interact with each other using network connection and sometime some malicious users can intercept those connections signals and then alter request values to perform malicious activities on receiver system. To detect and avoid such malicious attacks we are employing CUSUM entropy based DDOS (distributed denial of service) attacks detection technique.

The cumulative sum of entropy (CUSUM) can be used to track the randomness of the data over time. For example, if the cumulative sum of entropy is increasing, then the data is becoming more random. If the cumulative sum of entropy is decreasing, then the data is becoming less random.

As per CUSUM technique if there is sudden change in value then it will consider as attack. While normal communication all system network signature CUSUM value will be less than 0 and if malicious alteration occur then CUSUM value will get high and based on this approach we can say weather network communication is normal or attack.

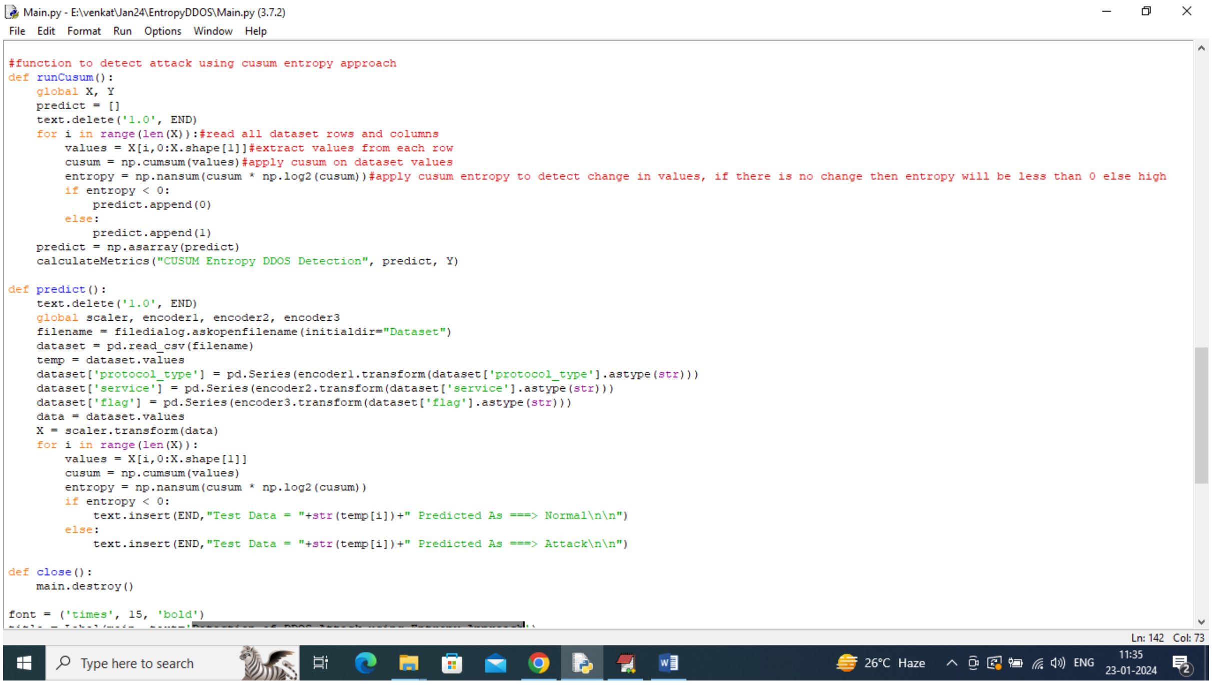
To train CUSUM algorithm we are employing DDOS dataset and below screen showing dataset details



```
duration,protocol_type,service,flag,src_bytes,dst_bytes,land,wrong_fragment,urgent,hot,num_failed_logins,logged_in,num_compromised,root_shell,su
0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.0,0.0,0.0,0.0,0.08,0.15,0.0,255,1,0.0,0.6,0.88,0.0,0.0,0.0,0.0,0.0,0
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.0,1.0,0.0,0.0,0.05,0.07,0.0,255,26,0.1,0.05,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,http,SF,232,8153,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,5,0.2,0.2,0.0,0.0,1.0,0.0,0.0,30,255,1.0,0.0,0.03,0.04,0.03,0.01,0.0,0.01,0
0,tcp,http,SF,199,420,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,30,32,0.0,0.0,0.0,0.0,1.0,0.0,0.09,255,255,1.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,121,19,0.0,0.0,1.0,1.0,0.16,0.06,0.0,255,19,0.07,0.07,0.0,0.0,0.0,0.0,1.0,1.0,1
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1.0,1.0,0.0,0.0,0.05,0.06,0.0,255,9,0.04,0.05,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,16,1.0,1.0,0.0,0.0,0.14,0.06,0.0,255,15,0.06,0.07,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,remote_job,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,270,23,1.0,1.0,0.0,0.0,0.09,0.05,0.0,255,23,0.09,0.05,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,133,8,1.0,1.0,0.0,0.0,0.06,0.06,0.0,255,13,0.05,0.06,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,205,12,0.0,0.0,1.0,1.0,0.06,0.06,0.0,255,12,0.05,0.07,0.0,0.0,0.0,0.0,1.0,1.0,1
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,199,3,1.0,1.0,0.0,0.0,0.02,0.06,0.0,255,13,0.05,0.07,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,http,SF,287,2251,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0.0,0.0,0.0,0.0,1.0,0.0,0.43,8,219,1.0,0.0,0.12,0.03,0.0,0.0,0.0,0.0,0
0,tcp,ftp_data,SF,334,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,2,2,0.0,0.0,0.0,0.0,1.0,0.0,0.0,2,20,1.0,0.0,1.0,0.2,0.0,0.0,0.0,0.0,1
0,tcp,name,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,233,1,1.0,1.0,0.0,0.0,0.0,0.0,0.06,0.0,255,1,0.0,0.07,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,netbios_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,96,16,1.0,1.0,0.0,0.0,0.17,0.05,0.0,255,2,0.01,0.06,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,http,SF,300,13788,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,9,0.0,0.11,0.0,0.0,1.0,0.0,0.22,91,255,1.0,0.0,0.01,0.02,0.0,0.0,0.0,0.0,0
0,icmp,eco_i,SF,18,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.0,0.0,0.0,0.0,1.0,0.0,0.0,1,16,1.0,0.0,1.0,1.0,0.0,0.0,0.0,0.0,1
0,tcp,http,SF,233,616,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0.0,0.0,0.0,0.0,1.0,0.0,0.0,66,255,1.0,0.0,0.02,0.03,0.0,0.0,0.02,0.0,0
0,tcp,http,SF,343,1178,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,10,0.0,0.0,0.0,0.0,1.0,0.0,0.2,157,255,1.0,0.0,0.01,0.04,0.0,0.0,0.0,0.0,0
0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,223,23,1.0,1.0,0.0,0.0,0.1,0.05,0.0,255,23,0.09,0.05,0.0,0.0,1.0,1.0,0.0,0.0,1
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,280,17,1.0,1.0,0.0,0.0,0.06,0.05,0.0,238,17,0.07,0.06,0.0,0.0,0.99,1.0,0.0,0.0,1
0,tcp,http,SF,253,11905,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,10,0.0,0.0,0.0,0.0,1.0,0.0,0.2,87,255,1.0,0.0,0.01,0.02,0.0,0.0,0.0,0.0,0
5607,udp,other,SF,147,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.0,0.0,0.0,0.0,1.0,0.0,0.0,255,1,0.0,0.85,1.0,0.0,0.0,0.0,0.0,0.0,0
0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,2,1.0,1.0,0.0,0.0,0.01,0.06,0.0,255,2,0.01,0.06,0.0,0.0,1.0,1.0,0.0,0.0,1
507,tcp,telnet,SF,437,14421,0,0,0,0,0,1,3,0,0,0,0,0,1,0,0,0,1,1,0.0,0.0,0.0,0.0,1.0,0.0,0.0,255,25,0.1,0.05,0.0,0.0,0.53,0.0,0.02,0.16,0
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,279,7,1.0,1.0,0.0,0.0,0.03,0.06,0.0,255,13,0.05,0.07,0.0,0.0,1.0,1.0,0.0,0.0,1
```

In above dataset screen first row represents dataset column names of network connection and remaining rows represents values and in last column we have class label as 0 or 1 where 0 means normal and 1 means attack. We will use this label to compare CUSUM prediction is correct or not and based on this we will evaluate its performance using Accuracy, Precision, Recall and FSCORE.

In below screen showing code for CUSUM computation



```
#function to detect attack using cusum entropy approach
def runCusum():
    global X, Y
    predict = []
    text.delete('1.0', END)
    for i in range(len(X)):#read all dataset rows and columns
        values = X[i,0:X.shape[1]]#extract values from each row
        cusum = np.cumsum(values)#apply cusum on dataset values
        entropy = np.nansum(cusum * np.log2(cusum))#apply cusum entropy to detect change in values, if there is no change then entropy will be less than 0 else high
        if entropy < 0:
            predict.append(0)
        else:
            predict.append(1)
    predict = np.asarray(predict)
    calculateMetrics("CUSUM Entropy DDOS Detection", predict, Y)

def predict():
    text.delete('1.0', END)
    global scaler, encoder1, encoder2, encoder3
    filename = filedialog.askopenfilename(initialdir="Dataset")
    dataset = pd.read_csv(filename)
    temp = dataset.values
    dataset['protocol_type'] = pd.Series(encoder1.transform(dataset['protocol_type'].astype(str)))
    dataset['service'] = pd.Series(encoder2.transform(dataset['service'].astype(str)))
    dataset['flag'] = pd.Series(encoder3.transform(dataset['flag'].astype(str)))
    data = dataset.values
    X = scaler.transform(data)
    for i in range(len(X)):
        values = X[i,0:X.shape[1]]
        cusum = np.cumsum(values)
        entropy = np.nansum(cusum * np.log2(cusum))
        if entropy < 0:
            text.insert(END,"Test Data = "+str(temp[i])+" Predicted As ===> Normal\n\n")
        else:
            text.insert(END,"Test Data = "+str(temp[i])+" Predicted As ===> Attack\n\n")

def close():
    main.destroy()

font = ('times', 15, 'bold')
```

Main.py - E:\venkat\Jan24\EntropyDDOS\Main.py (3.7.2)
File Edit Format Run Options Window Help
Ln: 142 Col: 73
Type here to search
26°C Haze
ENG
11:35
23-01-2024

In above screen read red colour comments to know about entropy based attack detection.

To implement this project we have designed following modules

1) Upload DDOS Dataset: using this module we will upload dataset to application and then plot graph of number of normal and attacks records

2) Pre-process Dataset: dataset contains both numeric and non-numeric values but CUSUM will accept only numeric values so will process dataset to convert non-numeric values to numeric values and then shuffle and replace missing values with 0

3) Normalize Training Features: using this module we will normalize all processed dataset values

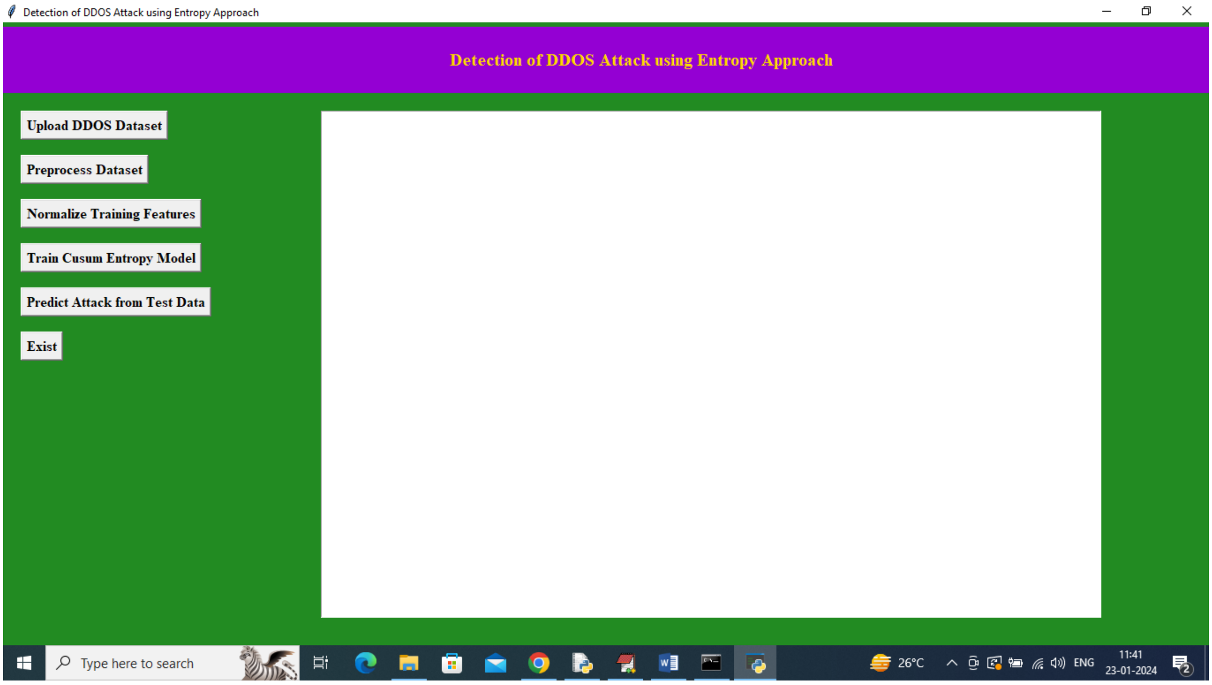
4) Train CUSUM Entropy Model: using this module will compute CUSUM entropy technique to detect attacks packets and then compare predicted labels with actual labels to calculate accuracy

5) Predict Attack from Test Data: using this module we will upload test data and then apply CUSUM entropy model to predict weather test data is normal or attack.

SCREEN SHOTS

To run project double click on 'run.bat' file to get below screen

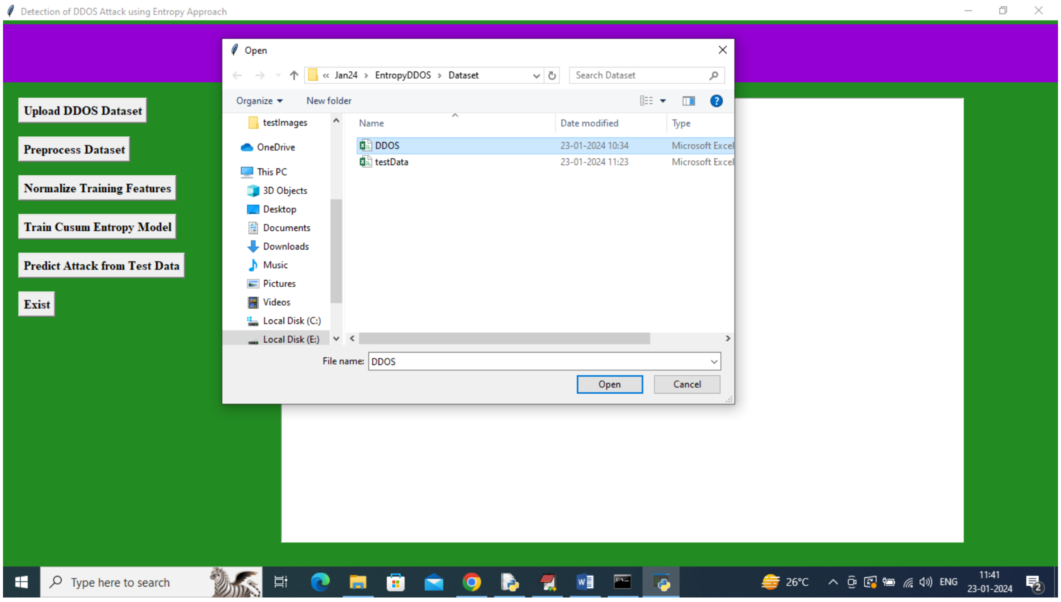
5.2.1 LOGIN PAGE:


Detection of DDOS Attack using Entropy Approach
Detection of DDOS Attack using Entropy Approach
Upload DDOS Dataset
Preprocess Dataset
Normalize Training Features
Train Cusum Entropy Model
Predict Attack from Test Data
Exist
Type here to search
26°C
ENG
11:41
23-01-2024

In above screen click on 'Upload DDOS Dataset' button to upload dataset and then will get below output

Fig 5.1:Login Page

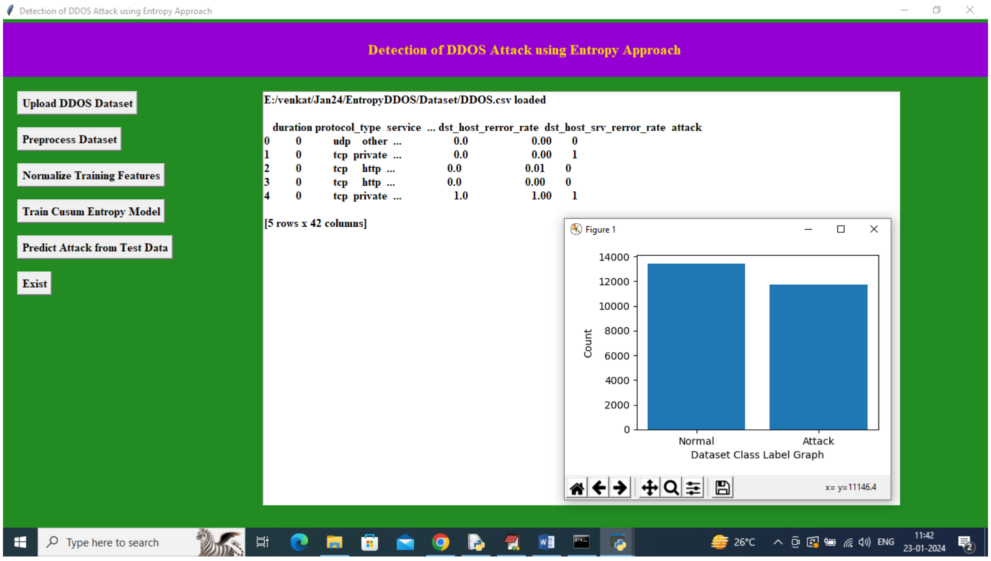
5.2.2 UPLOAD DDOS DATASET



In above screen selecting and uploading 'DDOS' dataset file and then click on 'Open' button to load dataset and then will get below output

Fig 5.2: Upload Ddos Dataset

5.2.3 PRE-PROCESS DATASET

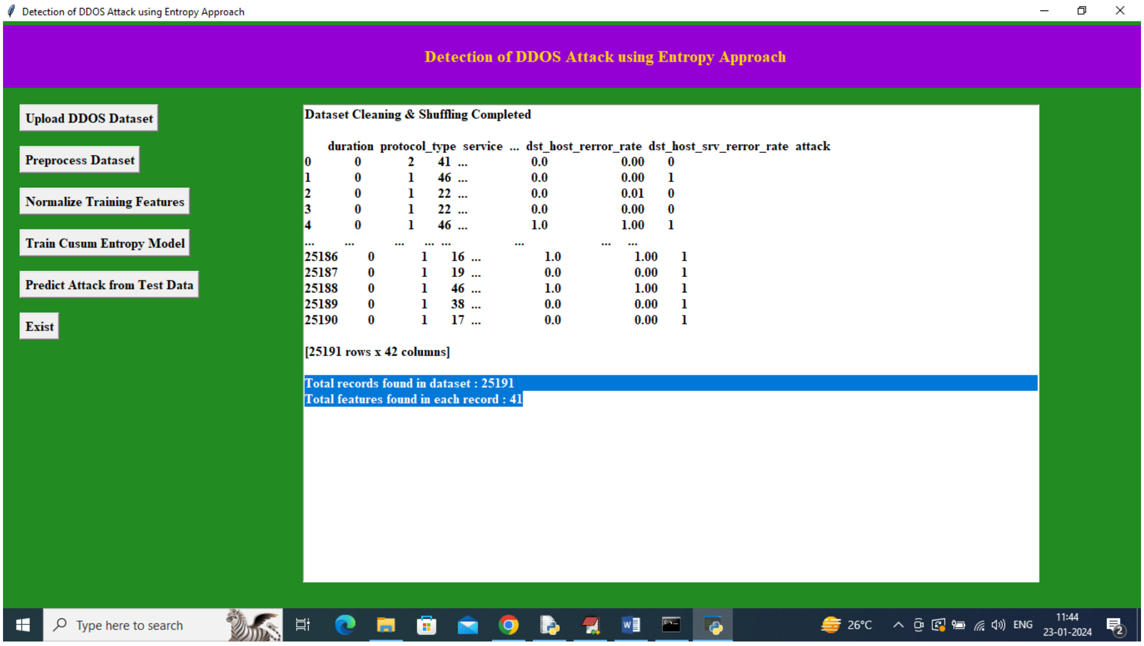

Detection of DDOS Attack using Entropy Approach
Detection of DDOS Attack using Entropy Approach
Upload DDOS Dataset
Preprocess Dataset
Normalize Training Features
Train Cusum Entropy Model
Predict Attack from Test Data
Exist
Figure 1
14000
12000
10000
8000
6000
4000
2000
0
Count
Normal
Attack
Dataset Class Label Graph
x= y=11146.4
Type here to search
26°C
ENG
11:42
23-01-2024

```
E:/venkat/Jan24/EntropyDDOS/Dataset/DDOS.csv loaded

  duration protocol_type  service  ... dst_host_rerror_rate  dst_host_srv_rerror_rate  attack
0        0           udp    other  ...                  0.0                      0.00       0
1        0           tcp  private  ...                  0.0                      0.00       1
2        0           tcp     http  ...                  0.0                      0.01       0
3        0           tcp     http  ...                  0.0                      0.00       0
4        0           tcp  private  ...                  1.0                      1.00       1

[5 rows x 42 columns]
```

In above screen dataset loaded and displaying few records from dataset and can see dataset contains both numeric and non-numeric values and by processing dataset will convert non-numeric to numeric values. In above graph x-axis represents class labels and y-axis represents number of records in that class label. Now close above graph and then click on 'Pre-process Dataset' button to process dataset and then will get below output

Fig 5.3: Pre-Process Dataset

5.2.4 NORMALIZE TEST DATA:

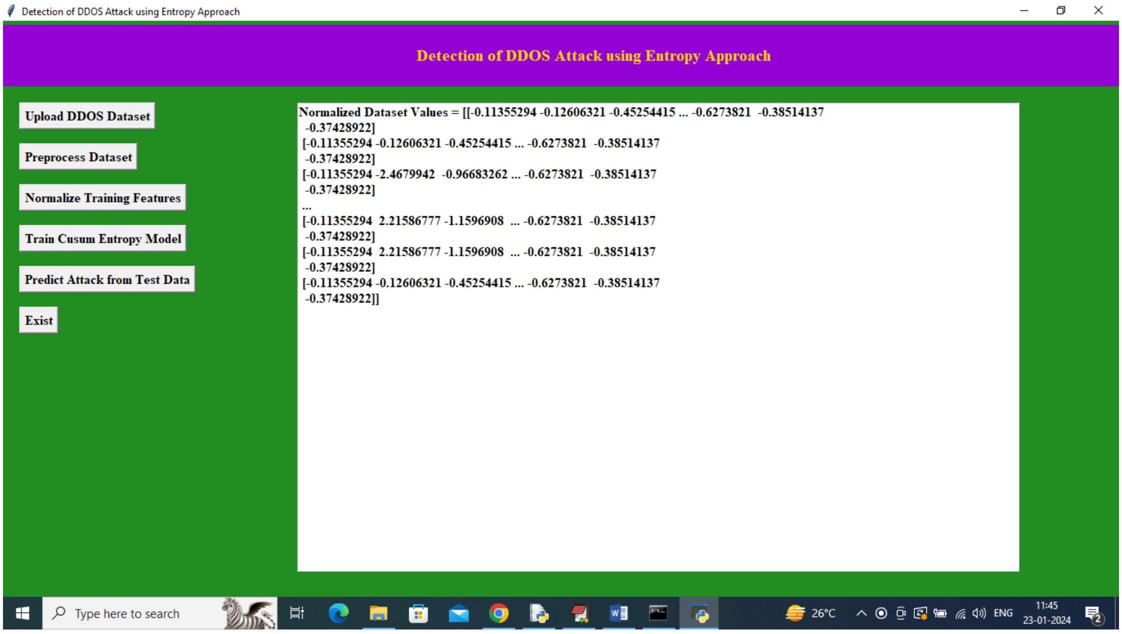

Detection of DDOS Attack using Entropy Approach
Detection of DDOS Attack using Entropy Approach
Upload DDOS Dataset
Preprocess Dataset
Normalize Training Features
Train Cusum Entropy Model
Predict Attack from Test Data
Exist
Type here to search
26°C
ENG
11:44
23-01-2024

```
Dataset Cleaning & Shuffling Completed

      duration  protocol_type  service  ...  dst_host_rerror_rate  dst_host_srv_rerror_rate  attack
0            0              2       41  ...                   0.0                      0.00       0
1            0              1       46  ...                   0.0                      0.00       1
2            0              1       22  ...                   0.0                      0.01       0
3            0              1       22  ...                   0.0                      0.00       0
4            0              1       46  ...                   1.0                      1.00       1
...        ...            ...      ...  ...                   ...                       ...     ...
25186        0              1       16  ...                   1.0                      1.00       1
25187        0              1       19  ...                   0.0                      0.00       1
25188        0              1       46  ...                   1.0                      1.00       1
25189        0              1       38  ...                   0.0                      0.00       1
25190        0              1       17  ...                   0.0                      0.00       1

[25191 rows x 42 columns]

Total records found in dataset : 25191
Total features found in each record : 41
```

In above screen can see all dataset values converted to numeric and can see total records and number of features available in each record and now click on 'Normalize test data' file button to get below output

Fig 5.4: Normalize Test Data

5.2.5 TRAIN CUSUM ENTROPY MODEL

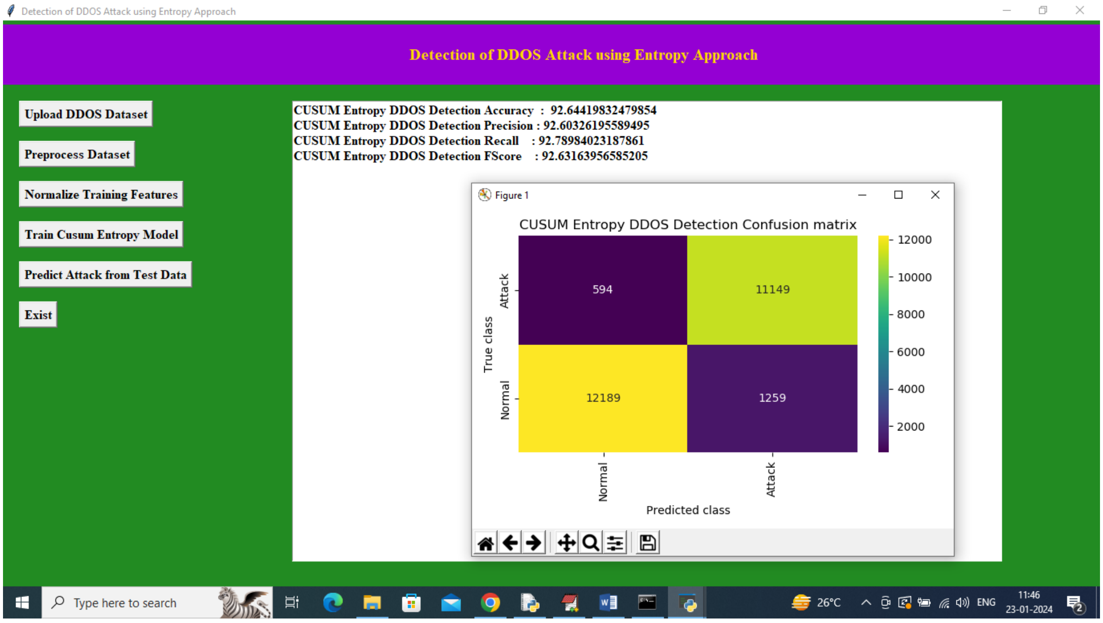

Detection of DDOS Attack using Entropy Approach
Detection of DDOS Attack using Entropy Approach
Upload DDOS Dataset
Preprocess Dataset
Normalize Training Features
Train Cusum Entropy Model
Predict Attack from Test Data
Exist
Type here to search
26°C
ENG
11:45
23-01-2024

```
Normalized Dataset Values = [[-0.11355294 -0.12606321 -0.45254415 ... -0.6273821  -0.38514137
  -0.37428922]
 [-0.11355294 -0.12606321 -0.45254415 ... -0.6273821  -0.38514137
  -0.37428922]
 [-0.11355294 -2.4679942  -0.96683262 ... -0.6273821  -0.38514137
  -0.37428922]
 ...
 [-0.11355294  2.21586777 -1.1596908  ... -0.6273821  -0.38514137
  -0.37428922]
 [-0.11355294  2.21586777 -1.1596908  ... -0.6273821  -0.38514137
  -0.37428922]
 [-0.11355294 -0.12606321 -0.45254415 ... -0.6273821  -0.38514137
  -0.37428922]]
```

In above screen all dataset values normalized and now click on 'Train CUSUM Entropy Model' button to train entropy and get below output

Fig 5.5: Train Cusum Entropy Model

5,2.6 PREDICT ATTACK FROM TEST DATA:

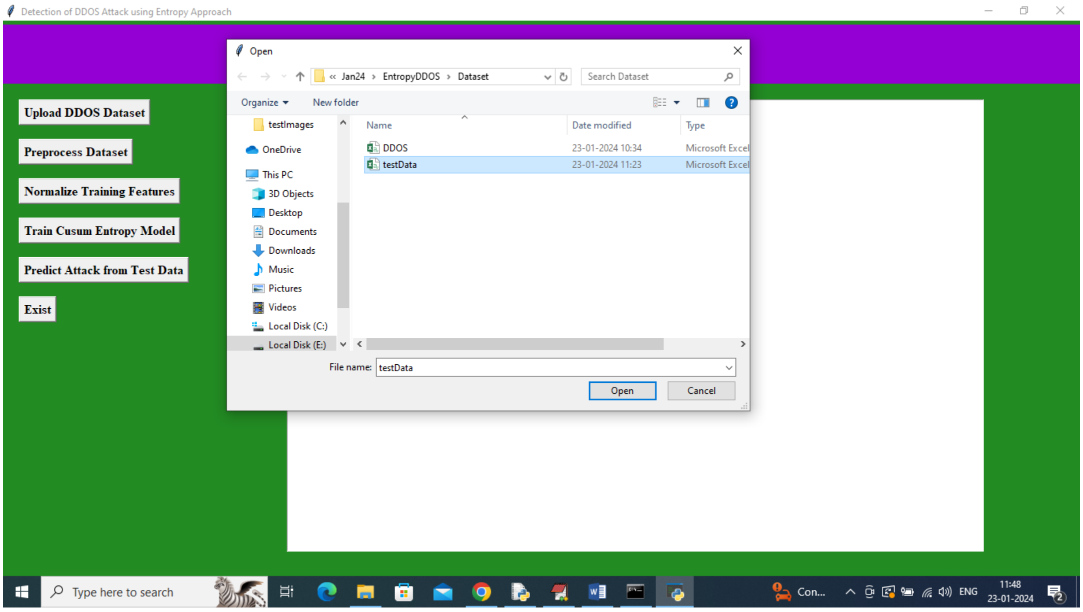

Detection of DDOS Attack using Entropy Approach
Detection of DDOS Attack using Entropy Approach
Upload DDOS Dataset
Preprocess Dataset
Normalize Training Features
Train Cusum Entropy Model
Predict Attack from Test Data
Exist
Figure 1
CUSUM Entropy DDOS Detection Confusion matrix
594
11149
12189
1259
True class
Attack
Normal
Predicted class
Normal
Attack
12000
10000
8000
6000
4000
2000
Type here to search
26°C
ENG
11:46
23-01-2024

```
CUSUM Entropy DDOS Detection Accuracy  :  92.64419832479854
CUSUM Entropy DDOS Detection Precision : 92.60326195589495
CUSUM Entropy DDOS Detection Recall    : 92.78984023187861
CUSUM Entropy DDOS Detection FScore    : 92.63163956585205
```

In above screen entropy DDOS attack detection got 92% accuracy and can see other metrics also and in confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels and all yellow and light green boxes contains correct prediction count and all blue boxes contains incorrect prediction count which are very few. Now close above graph and then click on 'Predict Attack from Test Data' button to upload test data and then will get below output

Fig 5.6: Predict Attack from Test Data

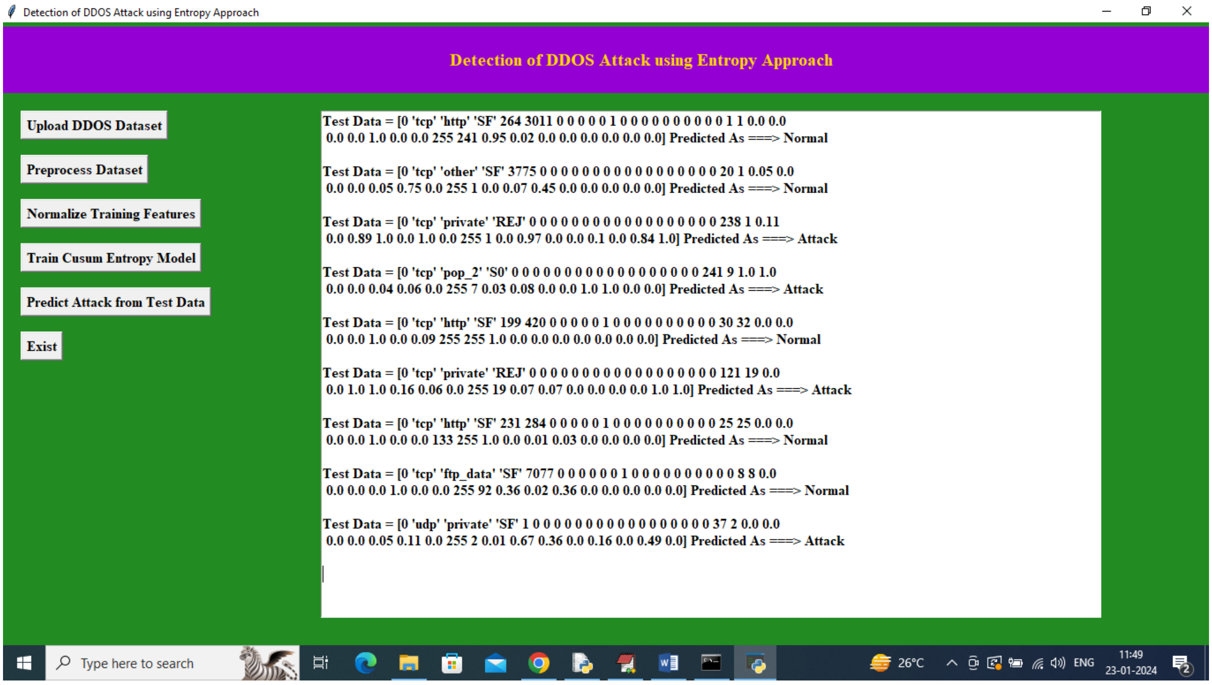
5.2.7 : TEST DATA FILE FOR PREDICTION:



In above screen selecting and uploading 'test data' file and then click on 'Open' button to get below output

Fig 5.7 : Test Data' File For Prediction

5.2.8 PREDICT THE DATA:


Detection of DDOS Attack using Entropy Approach
Detection of DDOS Attack using Entropy Approach
Upload DDOS Dataset
Preprocess Dataset
Normalize Training Features
Train Cusum Entropy Model
Predict Attack from Test Data
Exist
Type here to search
26°C
ENG
11:49
23-01-2024

```
Test Data = [0 'tcp' 'http' 'SF' 264 3011 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0.0 0.0
 0.0 0.0 1.0 0.0 0.0 255 241 0.95 0.02 0.0 0.0 0.0 0.0 0.0 0.0] Predicted As ===> Normal

Test Data = [0 'tcp' 'other' 'SF' 3775 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 20 1 0.05 0.0
 0.0 0.0 0.05 0.75 0.0 255 1 0.0 0.07 0.45 0.0 0.0 0.0 0.0 0.0] Predicted As ===> Normal

Test Data = [0 'tcp' 'private' 'REJ' 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 238 1 0.11
 0.0 0.89 1.0 0.0 1.0 0.0 255 1 0.0 0.97 0.0 0.0 0.1 0.0 0.84 1.0] Predicted As ===> Attack

Test Data = [0 'tcp' 'pop_2' 'S0' 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 241 9 1.0 1.0
 0.0 0.0 0.04 0.06 0.0 255 7 0.03 0.08 0.0 0.0 1.0 1.0 0.0 0.0] Predicted As ===> Attack

Test Data = [0 'tcp' 'http' 'SF' 199 420 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 30 32 0.0 0.0
 0.0 0.0 1.0 0.0 0.09 255 255 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0] Predicted As ===> Normal

Test Data = [0 'tcp' 'private' 'REJ' 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 121 19 0.0
 0.0 1.0 1.0 0.16 0.06 0.0 255 19 0.07 0.07 0.0 0.0 0.0 0.0 1.0 1.0] Predicted As ===> Attack

Test Data = [0 'tcp' 'http' 'SF' 231 284 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 25 25 0.0 0.0
 0.0 0.0 1.0 0.0 0.0 133 255 1.0 0.0 0.01 0.03 0.0 0.0 0.0 0.0] Predicted As ===> Normal

Test Data = [0 'tcp' 'ftp_data' 'SF' 7077 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.0
 0.0 0.0 0.0 1.0 0.0 0.0 255 92 0.36 0.02 0.36 0.0 0.0 0.0 0.0 0.0] Predicted As ===> Normal

Test Data = [0 'udp' 'private' 'SF' 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 37 2 0.0 0.0
 0.0 0.0 0.05 0.11 0.0 255 2 0.01 0.67 0.36 0.0 0.16 0.0 0.49 0.0] Predicted As ===> Attack
```

In above screen in square bracket can see TEST DATA values and after =➔ arrow symbol can see CUSUM predicted values as 'Normal or Attack'

Fig 5.8: Predict The Data

6. CONCLUSION AND FUTURE WORK

CONCLUSION

In conclusion, the detection of Distributed Denial of Service (DDoS) attacks using entropy-based approaches represents a promising avenue for improving network security in the face of evolving cyber threats. Throughout this study, we have explored the theoretical foundations, practical implementations, and potential benefits of leveraging entropy analysis for DDoS attack detection.

Entropy-based methods offer several advantages over traditional detection techniques, including their ability to capture the randomness and complexity of network traffic patterns. By analyzing entropy measures such as packet entropy, flow entropy, and information entropy, it becomes possible to identify anomalous behavior indicative of DDoS attacks with high accuracy and low false positive rates.

Furthermore, entropy-based approaches exhibit adaptability and scalability, making them suitable for deployment in dynamic network environments and capable of handling large volumes of traffic. Their integration with machine learning algorithms enables the detection system to learn and adapt to emerging DDoS attack techniques, enhancing its effectiveness over time.

7. REFRENCES

1 Douligeris, C., & Mitrokotsa, A. (2004). DDoS attacks and defense mechanisms: Classification and state-of-the-art. Computer Networks, 44(5), 643-666.

Ghosal, D., Swami, A., & Yu, D. (2014). Entropy-based DDoS attack detection and source identification using random neural networks. IEEE Transactions on Information Forensics and Security, 9(2), 307-318.

Hossain, M. S., Muhammad, G., & Amin, R. (2016). A survey of DDoS attacks and defense mechanisms in Cloud computing. Journal of Network and Computer Applications, 67, 235-254.

Kamhoua, C. A., Kwiat, K. A., Njilla, L. L., & Mohan, S. (2015). Entropy-based detection of DDoS attacks in software-defined networks. In Proceedings of the 10th International Conference on Availability, Reliability and Security (ARES) (pp. 81-90). IEEE.

Karap, A., & Sekeroglu, B. (2015). Entropy-based DDoS attack detection method in cloud computing. International Journal of Communication Systems, 28(17), 1879-1898.

Mirkovic, J., & Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, 34(2), 39-53.

Naseer, M. H., & Saeed, A. (2017). A survey on entropy-based techniques for DDoS detection and mitigation. Computers & Security, 66, 126-143.

Rajagopal, R., & Wong, W. C. (2010). Entropy-based DDoS detection model. In Proceedings of the International Conference on Advances in Computing, Control, and Telecommunication Technologies (ACT) (pp. 71-75). IEEE.

Razaque, A., & McGlynn, T. (2012). Entropy based collaborative detection of DDoS attacks on community clouds. In Proceedings of the IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom) (pp. 1895-1900). IEEE.

Teimouri, M., Fathy, M., & Naghibzadeh, M. (2019). Entropy-based detection and filtering of DDoS attacks in software-defined networks. Journal of Network and Computer Applications, 133, 45-54.