

**Name: Umut kuruluk**

**No:210101118**

## **Cyber Security Data Analysis**

**Introduction:** In this project, based on the given dataset, I am comparing attack and normal network traffic. I analyze the dataset using Python and visualize the findings. I'm using ChatGPT for analysis and obtaining the codes.

**Problem Definition:** Creating the exact prompt to ensure an excellent analysis, and giving clear and detailed instructions for solving it all at once.

### **Experiment Questions:**

#### **1. Differences in Packet Lengths:**

- What are the average packet lengths (both forward and backward) for benign traffic compared to potentially malicious traffic?
- Are there significant variations in packet length statistics between the two types of traffic?

#### **2. Flag Counts Analysis:**

- How do the flag counts (e.g., SYN, RST, FIN) differ between benign and malicious traffic?
- Can we identify any specific flag patterns that are more prevalent in malicious traffic compared to benign traffic?

### 3. Flow Features Analysis:

- What are the typical flow characteristics, such as flow duration, flow bytes per second, and flow packets per second, observed in benign traffic compared to potentially malicious traffic?
- Do significant differences exist in flow feature statistics between benign and malicious traffic, indicating distinctive behavioral patterns?

## Prompts&Codes:

### Prompt - 1:

Index(['Dst Port', 'Protocol', 'Timestamp', 'Flow Duration', 'Tot Fwd Pkts',  
'Tot Bwd Pkts', 'TotLen Fwd Pkts', 'TotLen Bwd Pkts', 'Fwd Pkt Len  
Max',  
'Fwd Pkt Len Min', 'Fwd Pkt Len Mean', 'Fwd Pkt Len Std',  
'Bwd Pkt Len Max', 'Bwd Pkt Len Min', 'Bwd Pkt Len Mean',  
'Bwd Pkt Len Std', 'Flow Byts/s', 'Flow Pkts/s', 'Flow IAT Mean',  
'Flow IAT Std', 'Flow IAT Max', 'Flow IAT Min', 'Fwd IAT Tot',  
'Fwd IAT Mean', 'Fwd IAT Std', 'Fwd IAT Max', 'Fwd IAT Min',  
'Bwd IAT Tot', 'Bwd IAT Mean', 'Bwd IAT Std', 'Bwd IAT Max',  
'Bwd IAT Min', 'Fwd PSH Flags', 'Bwd PSH Flags', 'Fwd URG Flags',  
'Bwd URG Flags', 'Fwd Header Len', 'Bwd Header Len', 'Fwd  
Pkts/s',  
'Bwd Pkts/s', 'Pkt Len Min', 'Pkt Len Max', 'Pkt Len Mean',  
'Pkt Len Std', 'Pkt Len Var', 'FIN Flag Cnt', 'SYN Flag Cnt',  
'RST Flag Cnt', 'PSH Flag Cnt', 'ACK Flag Cnt', 'URG Flag Cnt',  
'CWE Flag Count', 'ECE Flag Cnt', 'Down/Up Ratio', 'Pkt Size Avg',  
'Fwd Seg Size Avg', 'Bwd Seg Size Avg', 'Fwd Byts/b Avg',  
'Fwd Pkts/b Avg', 'Fwd Blk Rate Avg', 'Bwd Byts/b Avg',  
'Bwd Pkts/b Avg', 'Bwd Blk Rate Avg', 'Subflow Fwd Pkts',

'Subflow Fwd Byts', 'Subflow Bwd Pkts', 'Subflow Bwd Byts',  
'Init Fwd Win Byts', 'Init Bwd Win Byts', 'Fwd Act Data Pkts',  
'Fwd Seg Size Min', 'Active Mean', 'Active Std', 'Active Max',  
'Active Min', 'Idle Mean', 'Idle Std', 'Idle Max', 'Idle Min', 'Label'],  
dtype='object')

these are the labels at my dataset.csv file and

172.31.69.25-94.231.103.172-22-45498-  
6,94.231.103.172,45498,172.31.69.25,22,6,20/02/2018  
08:34:07,888751,11,11,1249,1969,736,0,113.5454545,220.8960677,97  
6,0,179,364.1864907,3620.811678,24.75383994,42321.47619,47851.7  
3578,101609,14,888751,88875.1,49295.73551,140273,14,788197,7881  
9.7,55863.18804,139285,72,0,0,0,0,360,360,12.37691997,12.37691997  
,0,976,139.9130435,290.633933,84468.083,0,0,0,1,0,0,0,0,1,146.27272  
73,113.5454545,179,0,0,0,0,0,0,11,1249,11,1969,14600,233,7,32,0,0,0,  
0,0,0,0,0,Benign

8.0.6.4-8.6.0.1-0-0-0,8.6.0.1,0,8.0.6.4,0,0,20/02/2018  
08:33:22,112642816,3,0,0,0,0,0,0,0,0,0,0,0,0.026632857,5.63E+07,7.  
071067812,5.63E+07,5.63E+07,1.13E+08,5.63E+07,7.071067812,5.63  
E+07,5.63E+07,0,0,0,0,0,0,0,0,0,0,0.026632857,0,0,0,0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,3,0,0,0,-1,-  
1,0,0,0,0,0,0,5.63E+07,7.071067812,5.63E+07,5.63E+07,Benign

8.0.6.4-8.6.0.1-0-0-0,8.6.0.1,0,8.0.6.4,0,0,20/02/2018  
08:36:11,112642712,3,0,0,0,0,0,0,0,0,0,0,0,0.026632881,5.63E+07,18  
.38477631,5.63E+07,5.63E+07,1.13E+08,5.63E+07,18.38477631,5.63E  
+07,5.63E+07,0,0,0,0,0,0,0,0,0,0,0.026632881,0,0,0,0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,3,0,0,0,-1,-  
1,0,0,0,0,0,0,5.63E+07,18.38477631,5.63E+07,5.63E+07,Benign

8.0.6.4-8.6.0.1-0-0-0,8.6.0.1,0,8.0.6.4,0,0,20/02/2018  
08:39:00,112642648,3,0,0,0,0,0,0,0,0,0,0,0,0.026632897,5.63E+07,5.  
656854249,5.63E+07,5.63E+07,1.13E+08,5.63E+07,5.656854249,5.63  
E+07,5.63E+07,0,0,0,0,0,0,0,0,0,0,0.026632897,0,0,0,0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,3,0,0,0,-1,-  
1,0,0,0,0,0,0,5.63E+07,5.656854249,5.63E+07,5.63E+07,Benign

8.0.6.4-8.6.0.1-0-0-0,8.6.0.1,0,8.0.6.4,0,0,20/02/2018  
08:41:49,112642702,3,0,0,0,0,0,0,0,0,0,0,0,0.026632884,5.63E+07,65  
.05382387,5.63E+07,5.63E+07,1.13E+08,5.63E+07,65.05382387,5.63E

172.31.69.25-91.185.191.213-22-39614-  
6,91.185.191.213,39614,172.31.69.25,22,6,20/02/2018  
08:45:21,1079159,10,11,1249,1969,736,0,124.9,229.4362414,976,0,17  
9,364.1864907,2981.951686,19.45959771,53957.95,59825.5431,12477  
0,17,1079159,119906.5556,47611.70351,162362,17,954962,95496.2,6  
7044.34139,163141,76,0,0,0,0,328,360,9.266475098,10.19312261,0,97  
6,146.2727273,295.8307402,87515.82684,0,0,0,1,0,0,0,0,1,153.238095  
2,124.9,179,0,0,0,0,0,10,1249,11,1969,65535,233,7,32,0,0,0,0,0,0,0  
,Benign

In this project, your task is to explore some insights about the dataset by querying your custom research questions (max 3 ).

The goal is to discover information about the difference between the attack and normal network traffic.

- What are the average packet lengths (both forward and backward) for benign traffic compared to potentially malicious traffic?
- Are there significant variations in packet length statistics between the two types of traffic?

**Answer - 1:**

```
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
```

```

df = pd.read_csv('dataset.csv')

# Filter the dataset for benign and potentially malicious traffic
benign_traffic = df[df['Label'] == 'Benign']
malicious_traffic = df[df['Label'] != 'Benign']

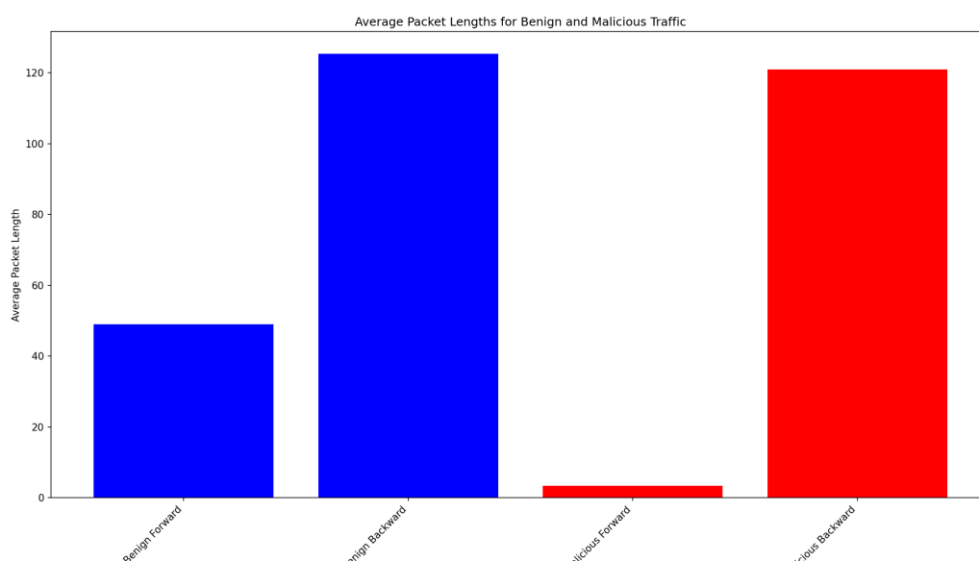
# Calculate average packet lengths for benign traffic
benign_avg_fwd_pkt_len = benign_traffic['Fwd Pkt Len Mean'].mean()
benign_avg_bwd_pkt_len = benign_traffic['Bwd Pkt Len Mean'].mean()

# Calculate average packet lengths for malicious traffic
malicious_avg_fwd_pkt_len = malicious_traffic['Fwd Pkt Len Mean'].mean()
malicious_avg_bwd_pkt_len = malicious_traffic['Bwd Pkt Len Mean'].mean()

# Create a bar plot
labels = ['Benign Forward', 'Benign Backward', 'Malicious Forward', 'Malicious Backward']
avg_lengths = [benign_avg_fwd_pkt_len, benign_avg_bwd_pkt_len,
malicious_avg_fwd_pkt_len, malicious_avg_bwd_pkt_len]

plt.bar(labels, avg_lengths, color=['blue', 'blue', 'red', 'red'])
plt.ylabel('Average Packet Length')
plt.title('Average Packet Lengths for Benign and Malicious Traffic')
plt.xticks(rotation=45, ha='right')
plt.show()

```



## Prompt - 2:

### 2. Flag Counts Analysis:

- How do the flag counts (e.g., SYN, RST, FIN) differ between benign and malicious traffic?

- Can we identify any specific flag patterns that are more prevalent in malicious traffic compared to benign traffic?

Make the second experiment with python and visualize it at code using pandas, numpy and matplotlib libraries.

## Answer - 2:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

# Load the dataset
df = pd.read_csv('dataset.csv')

# Filter the dataset for benign and potentially malicious traffic
benign_traffic = df[df['Label'] == 'Benign']
malicious_traffic = df[df['Label'] != 'Benign']

# Calculate flag counts for benign traffic
benign_syn_count = benign_traffic['SYN Flag Cnt'].sum()
benign_rst_count = benign_traffic['RST Flag Cnt'].sum()
benign_fin_count = benign_traffic['FIN Flag Cnt'].sum()

# Calculate flag counts for malicious traffic
malicious_syn_count = malicious_traffic['SYN Flag Cnt'].sum()
malicious_rst_count = malicious_traffic['RST Flag Cnt'].sum()
malicious_fin_count = malicious_traffic['FIN Flag Cnt'].sum()

# Create a bar plot
labels = ['SYN', 'RST', 'FIN']
benign_counts = [benign_syn_count, benign_rst_count, benign_fin_count]
malicious_counts = [malicious_syn_count, malicious_rst_count,
malicious_fin_count]

x = np.arange(len(labels))
width = 0.35
```

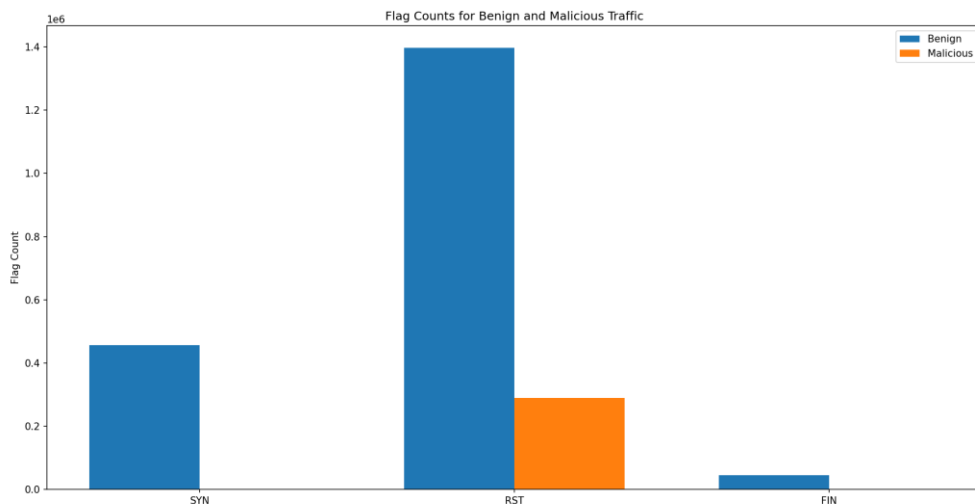
```

fig, ax = plt.subplots()
rects1 = ax.bar(x - width/2, benign_counts, width, label='Benign')
rects2 = ax.bar(x + width/2, malicious_counts, width, label='Malicious')

ax.set_ylabel('Flag Count')
ax.set_title('Flag Counts for Benign and Malicious Traffic')
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend()

plt.show()

```



### Prompt - 3:

#### 3. Flow Features Analysis:

- What are the typical flow characteristics, such as flow duration, flow bytes per second, and flow packets per second, observed in benign traffic compared to potentially malicious traffic?

- Do significant differences exist in flow feature statistics between benign and malicious traffic, indicating distinctive behavioral patterns?

Make the third experiment with python and visualize it at code using pandas, numpy and matplotlib libraries.

### Answer - 3:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv("dataset.csv")

# Filter benign and malicious traffic
benign_traffic = df[df['Label'] == 'Benign']
malicious_traffic = df[df['Label'] != 'Benign']

# Define the flow features to analyze
flow_features = ['Flow Duration', 'Flow Byts/s', 'Flow Pkts/s', 'Flow IAT
Mean', 'Flow IAT Std']

# Plot the comparison
plt.figure(figsize=(15, 10))

for i, feature in enumerate(flow_features, start=1):
    plt.subplot(2, 3, i)

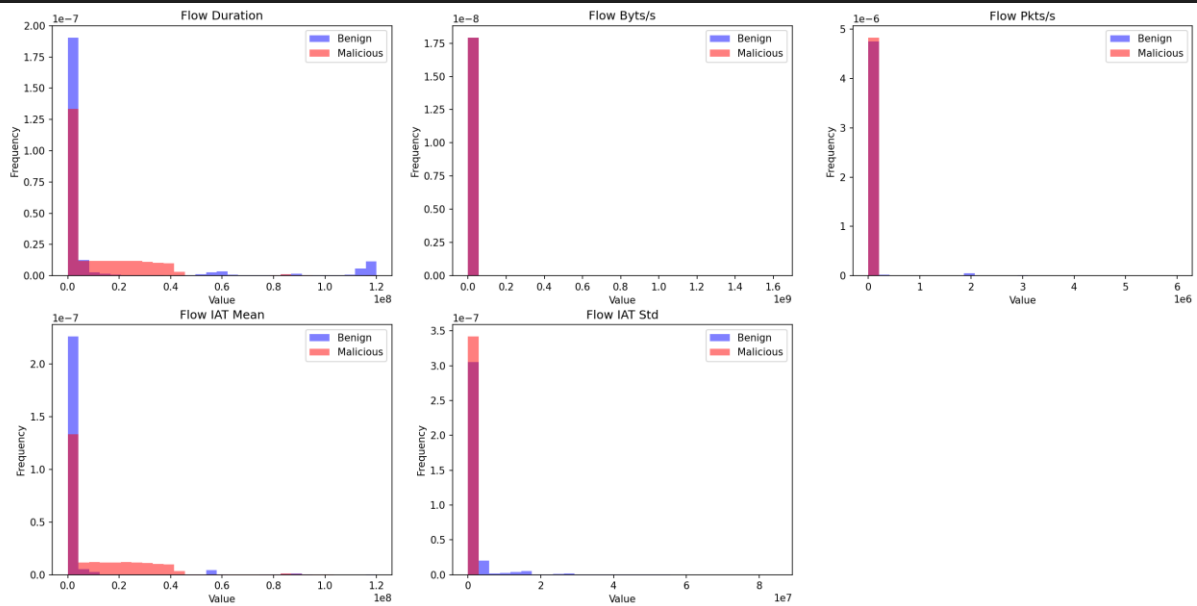
    # Filter out infinite or NaN values
    benign_values = benign_traffic[feature].replace([np.inf, -np.inf],
np.nan).dropna()
    malicious_values = malicious_traffic[feature].replace([np.inf, -np.inf],
np.nan).dropna()

    # Adjust the range of histogram bins
    max_value = max(benign_values.max(), malicious_values.max())
    bins = np.linspace(0, max_value, 30)

    # Plot histograms
    plt.hist(benign_values, bins=bins, alpha=0.5, color='blue',
label='Benign', density=True)
    plt.hist(malicious_values, bins=bins, alpha=0.5, color='red',
label='Malicious', density=True)
    plt.title(feature)
    plt.xlabel('Value')
    plt.ylabel('Frequency')
    plt.legend()
```



```
plt.tight_layout()
plt.show()
```



(Colors mix where they intersect)

## References:

### Dataset(02-20-2018.csv):

<https://www.kaggle.com/datasets/solarmainframe/ids-intrusion-csv>

### Artificial Intelligence:

<https://chatgpt.com/>