TCP_plaintext_RFC761

September 28, 2021

[ ]:
```
import pandas as pd
from numpy import arange
import matplotlib.pyplot as plt
%matplotlib inline
```

[ ]:
```
# Read from CSV

df = pd.read_csv('Outputs/packets-tcp-rfc761-plaintext-dinamico-2021.09.
↪27-processed')
```

[ ]:
```
# Setting global var

bytesize = 32
proto = 'TCP'
dstport = 8088
encoding = 'plaintext'
text = 'RFC761'
```

[ ]:
```
# Add a new column to the end called 'flow'

df['flow'] = df['srcip'] + ':' + df.srcport.map(str) + ' -> ' + df['dstip'] + ':
↪' + df.dstport.map(str)
# Read a specific location (R,C)
print('Example of flow {}'.format(df.iloc[5,10]))
```

```
Example of flow 127.0.0.1:1066 -> 127.0.0.1:8088
```

[ ]:
```
# Sort dataframe by an index (column) and show

df = df.sort_values(['payload_size','flow'])
print(df.iloc[:,6:11])
```

```
     payload_size   shannon      bien     tbien  \
0               1  1.000000  0.468917  0.759649
3               1  1.000000  0.468917  0.759649
316             1  1.000000  0.468917  0.759649
317             1  1.000000  0.468917  0.759649
248           217  0.287684  0.353241  0.482719
..            …         …         …         …
```

```
307          1024  0.290686  0.623590  0.972770
309          1024  0.296769  0.583578  0.912856
311          1024  0.291652  0.607224  0.973641
313          1024  0.284554  0.659996  0.873829
315          1024  0.291021  0.586671  0.919812

                                   flow
0    127.0.0.1:49792 -> 127.0.0.1:49791
3    127.0.0.1:49792 -> 127.0.0.1:49791
316  127.0.0.1:49792 -> 127.0.0.1:49791
317  127.0.0.1:49792 -> 127.0.0.1:49791
248    127.0.0.1:1066 -> 127.0.0.1:8088
..                                  …
307    127.0.0.1:8088 -> 127.0.0.1:1066
309    127.0.0.1:8088 -> 127.0.0.1:1066
311    127.0.0.1:8088 -> 127.0.0.1:1066
313    127.0.0.1:8088 -> 127.0.0.1:1066
315    127.0.0.1:8088 -> 127.0.0.1:1066

[318 rows x 5 columns]
```

[ ]:
```
# Filtering by port

is_port = df['dstport']==dstport
print(is_port.head())
df = df[is_port]
```

```
0      False
3      False
316    False
317    False
248     True
Name: dstport, dtype: bool
```

[ ]:
```
# Filtering by the number of packets of chosen size

is_bytes = df['payload_size']>2
print(is_bytes.head())
df = df[is_bytes]
```

```
248    True
2      True
5      True
7      True
9      True
Name: payload_size, dtype: bool
```

```
[ ]: # Minimize number of displayed columns

     # pd.set_option("display.max.columns", None)
     # df.head()
```

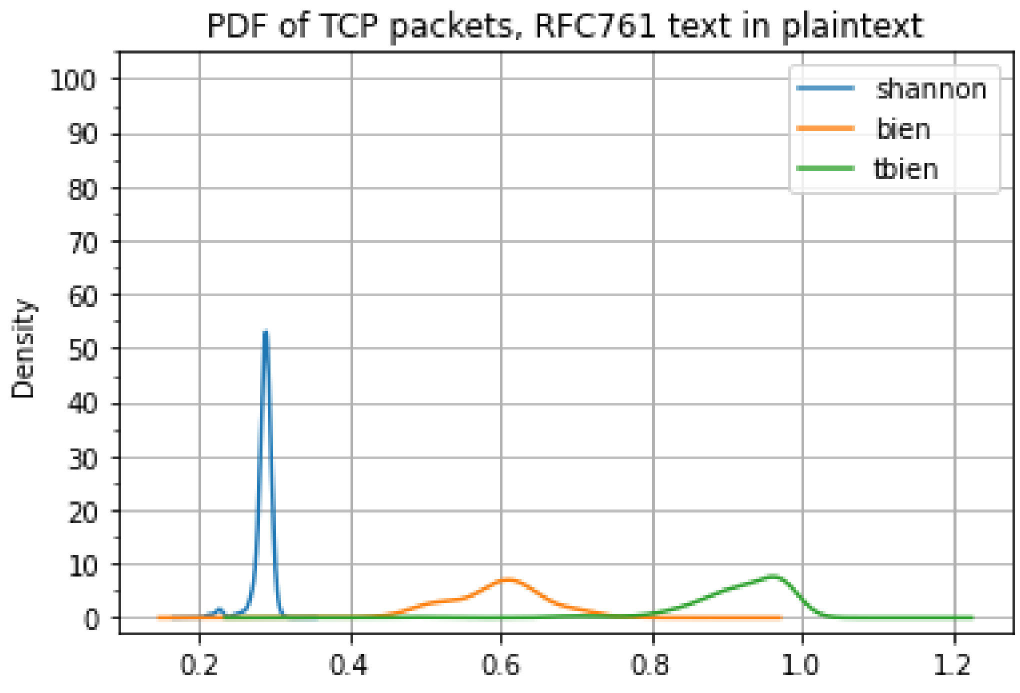
```
[ ]: # Aggregation by flow and each entropies mean

     df[['flow','shannon','bien','tbien','payload_size']].groupby('flow').mean().
      ↪sort_values('tbien', ascending=False)
```

```
[ ]:                                      shannon      bien    tbien  payload_size
     flow
     127.0.0.1:1066 -> 127.0.0.1:8088  0.285858  0.595526  0.91816   1018.859873
```

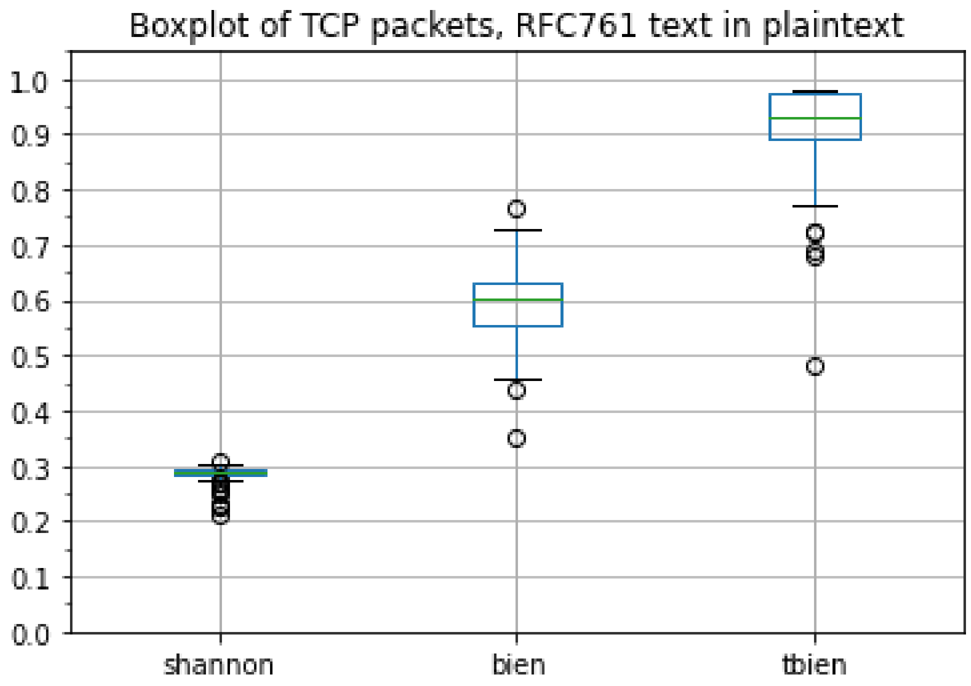
```
[ ]: # Plot 1

     title = 'PDF of {} packets, {} text in {}'.format(proto, text, encoding)
     ax = df.plot(x='payload_size',␣
      ↪y=['shannon','bien','tbien'],kind='density',title=title, grid=True)
     ax.xaxis.grid(True, which='major', linestyle='-', linewidth=1)
     ymajortick = arange(0,110,10)
     yminortick = arange(0,110,5)
     ax.set_yticks( ymajortick, minor=False )
     ax.set_yticks( yminortick, minor=True )
     ax.grid('on', which='both', axis='x' )
     plt.savefig('Plots/rfc761/{}{}{}density.png'.format(proto, encoding, text),␣
      ↪transparent=False)
```



```
[ ]: # Plot 2

     title = 'Boxplot of {} packets, {} text in {}'.format(proto, text, encoding)
     ax = df.plot(x='payload_size',␣
      ↪y=['shannon','bien','tbien'],kind='box',title=title, grid=True)
     ax.xaxis.grid(True, which='major', linestyle='-', linewidth=1)
     ymajortick = arange(0,1.1,0.1)
     yminortick = arange(0,1.1,0.05)
     ax.set_yticks( ymajortick, minor=False )
     ax.set_yticks( yminortick, minor=True )
     ax.grid('on', which='both', axis='x' )
     plt.savefig('Plots/rfc761/{}{}{}box.png'.format(proto, encoding, text),␣
      ↪transparent=False)
```



```
[ ]: # Table of data

     df = df.describe()
     print(df)
```

```
       srcport  dstport  payload_size     shannon        bien       tbien
count    157.0    157.0    157.000000  157.000000  157.000000  157.000000
mean    1066.0   8088.0   1018.859873    0.285858    0.595526    0.918160
```

```
std         0.0      0.0     64.405612    0.012732    0.063446    0.068418
min      1066.0   8088.0    217.000000    0.213344    0.353241    0.482719
25%      1066.0   8088.0   1024.000000    0.284272    0.556707    0.891808
50%      1066.0   8088.0   1024.000000    0.288051    0.601622    0.932076
75%      1066.0   8088.0   1024.000000    0.291652    0.633839    0.972473
max      1066.0   8088.0   1024.000000    0.307852    0.764830    0.977556
```

[ ]:
```
# Exporting new data

filename = 'Outputs/RFC761/{}{}{}data.csv'.format(proto, encoding, text)
df.to_csv(filename,',')
```