**Installation:**

To use the script, you have to install:

Dpkt Library: pip install dpkt

Socket Library: pip install socket

Numpy Library: pip install numpy

Matplotlib Library: pip install -U matplotlib

**Run the script:**

python packetParser.py fileName

Inet\_to\_str(inet):

Convert inet object to a string

Args:

inet (inet struct): inet network address

Returns:

str: Printable/readable IP address

plot\_cdf\_packet\_length(lengths\_list):

Show a CDF plot of packet length

Args:

Length\_list(Struct List): a list with the lengths of every package

Returns:

plot\_cdf\_flow\_duration(dur\_list):

Show a CDF plot of flow duration

Args:

dur\_list(Struct List): a list with the duration of every package

Returns:

plot\_cdf\_flow\_length(lengths\_list):

Show a CDF plot of flow length

Args:

length\_list(Struct List): a list with the length of every package

Returns:

plot\_packet\_protocol(tcp, udp, icmp, arp, total):

Show a plot of the percentage of package’s protocol

Args:

tcp(Struct int): total packages with tcp protocol

udp(Struct int): total packages with udp protocol

icmp(Struct int): total packages with icmp protocol

arp(Struct int): total packages with arp protocol

total(Struct int): total packages

Returns:

analyze\_packets(pcap)

Analyze the packages from the pcap file. Keeps counters of how many packets of tcp, udp, icmp and arp are exists. Calculates the length and the duration of every flow and the length of every package.

Args:

pcap(Struct pcap): pcap file with the trace

Returns: