CSCI 651 - Project 2
(Name: Vinay Jain, vj9898)

PING EXECUTION README:-

NOTE: The assignment's pdf doesn't specify whether we have to build a program that is capable of handling multiple command line arguments or not.

THEREFORE, I HAVE MADE MY PROGRAM BASED ON HANDLING ONE COMMAND LINE ARGUMENT AT A TIME.

- Refer to the file “vj9898_ping.py”.
- Copy the python program “vj9898_ping.py” into any directory of your choice.
- Open the terminal and navigate to the folder where you copied the python file.
- Install “scapy” library in your system before proceeding.
    a) Using pip or pip3 - I had to use pip3
        i) pip3 install scapy
- Similarly make sure to install other necessary libraries like
    a) _socket
    b) sys, time, datetime
    c) socket
- By default, the destination address where the program would ping is “google.com”. If you want to change that and ping some other address:
    a) Navigate to “main()” function.
    b) The first line of the “main” function initializes a string variable:
        i) “whoToPing”: which address you want to ping
    c) Change this variable to the address (string format) that you want to ping.
- You might wanna put the “sudo” command in front of every command below.
    a) Avoid permission denied error.
- Use the following command if running from a terminal (I ran it via a mac terminal) using python or python 3 - python 3 is preferred
    a) To run it with no parameters, ping uninterrupted, ping after every 1 second:
        i) sudo python3 complete_path_to_the_pcap_vj9898_ping.py
    b) To run ping until ‘count’ number of packets pinged:
        i) sudo python3 complete_path_to_the_pcap_vj9898_ping.py -c count
    c) run ping after i seconds until interrupted:
        i) sudo python3 complete_path_to_the_pcap_vj9898_ping.py -i seconds
    d) run ping until timeout
        i) sudo python3 complete_path_to_the_pcap_vj9898_ping.py -t timeout
    e) run ping with the modified data size until interrupted
        i) sudo python3 complete_path_to_the_pcap_vj9898_ping.py -s sizeInBytes

- See the results of the program.

TRACEROUTE EXECUTION README:-

NOTE: The assignment's pdf doesn't specify whether we have to build a program that is capable of handling multiple command line arguments or not.

THEREFORE, I HAVE MADE MY PROGRAM BASED ON HANDLING ONE COMMAND LINE ARGUMENT AT A TIME.

ANOTHER NOTE: As per the assignment pdf, I couldn't really comprehend the proper meaning of using this program with '-n' parameter. "Print hop addresses numerically rather than symbolically and numerically."

THEREFORE, I JUST EXECUTED THE NORMAL TRACEROUTE PROGRAM WITH 3 REATTEMPTS FOR EVERY HOP FOR THIS COMMAND LINE ARGUMENT.

- Refer to the file “vj9898_traceroute.py”.
- Copy the python program “vj9898_traceroute.py” into any directory of your choice.
- Open the terminal and navigate to the folder where you copied the python file.
- Install “scapy” library in your system before proceeding.
    a) Using pip or pip3 - I had to use pip3
        i) pip3 install scapy
- Similarly make sure to install other necessary libraries like
    a) _socket
    b) Sys, time, datetime
    c) socket
- By default, the destination address where the program would traceroute is “google.com”. If you want to change that and traceroute some other address:
    a) Navigate to “main()” function.
    b) The first line of the “main” function initializes a string variable:
        i) “whoToTraceroute”: which address you want to traceroute
    c) Change this variable to the address (string format) that you want to traceroute.
- By default, the number of hops is set to 119
    a) Look for the ‘numberOfHops’ variable in the 6th line of the ‘main()’ function.
    b) Change it if you wish.
- By default, the number of reattempts for every hop is set to 3.

    a) Look for ‘numberOfReattempts’ in the 7th line of the ‘main()’ function.
    b) Change it if you wish.

- You might wanna put the “sudo” command in front of every command below.
    a) Avoid permission denied error.
- Use the following command if running from a terminal (I ran it via a mac terminal) using python or python 3 - python 3 is preferred
    a) To run it with no parameters:
        i) sudo python3 complete_path_to_the_pcap_vj9898_traceroute.py
    b) To run traceroute with ‘nqueries’ number of attempts for every hop:
        i) sudo python3 complete_path_to_the_pcap_vj9898_traceroute.py -q nqueries
    c) run traceroute, generate summary of non answered attempts for every hop:
        i) sudo python3 complete_path_to_the_pcap_vj9898_traceroute.py -S
    d) run traceroute (refer ANOTHER NOTE above)
        i) sudo python3 complete_path_to_the_pcap_vj9898_traceroute.py -n
- See the results of the program.