Introduction

In this coding project you will augment a raw socket implementation of ICMP's ping to enhance ping accuracy and create trace route functionality.

Traceroute is a computer networking diagnostic tool which allows a user to trace the route from a host running the traceroute program to any other host in the world. Traceroute is implemented with ICMP messages. It works by sending ICMP echo (ICMP type ‘8’) messages to the same destination with increasing value of the time-to-live (TTL) field. The routers along the traceroute path return ICMP Time Exceeded (ICMP type ‘11’ ) when the TTL field become zero. The final destination sends an ICMP reply (ICMP type ’0’ ) messages on receiving the ICMP echo request. The IP addresses of the routers which send replies can be extracted from the received packets. The round-trip time between the sending host and a router is determined by setting a timer at the sending host.

Your task is to develop your own Traceroute application in python using the skeleton code provided as the base. Your application will use ICMP but, in order to keep it simple, will not exactly follow the official specification in RFC 1739.

Instructions

Below you will find the skeleton code for the client. You are to update the skeleton code to achieve the following objectives.

Objectives:

* Update the \_\_validateIcmpReplyPacketWithOriginalPingData() function:
  + Confirm the following items received are the same as what was sent:
    - sequence number
    - packet identifier
    - raw data
  + Set the valid data variable in the IcmpPacket\_EchoReply class based the outcome of the data comparison.
  + Create variables within the IcmpPacket\_EchoReply class that identify whether each value that can be obtained from the class is valid. For example, the IcmpPacket\_EchoReply class has an IcmpIdentifier. Create a variable, such as IcmpIdentifier\_isValid, along with a getter function, such as getIcmpIdentifier\_isValid(), and setting function, such as setIcmpIdentifier\_isValid(), so you can easily track and identify which data points within the echo reply are valid. Note: There are similar examples within the current skeleton code.
  + Create debug messages that show the expected and the actual values along with the result of the comparison.
* Update the printResultToConsole() function:
  + Identify if the echo response is valid and report the error information details. For example, if the raw data is different, print to the console what the expected value and the actual value.
* Currently, the program calculates the round-trip time for each packet and prints it out individually. Modify the code to correspond to the way the standard ping program works. You will need to report the minimum, maximum, and average RTTs at the end of all pings from the client. In addition, calculate the packet loss rate (in percentage). It is recommended to create an output that is easily readable with the amount of data used for a trace route since a ping is the foundation for such functionality.
* Your program can only detect timeouts in receiving ICMP echo responses. Modify the Pinger program to parse the ICMP response error codes and display the corresponding error results to the user. Examples of ICMP response error codes are 0: Destination Network Unreachable, 1: Destination Host Unreachable.
  + Ref: [https://www.iana.org/assignments/icmp-parameters/icmp-parameters.xhtml (Links to an external site.)](https://www.iana.org/assignments/icmp-parameters/icmp-parameters.xhtml)
* The skeleton code currently has a placeholder for performing a trace route function. It starts with the traceRoute() function and uses private functions to carry out the implementation. Update the \_\_sendIcmpTraceRoute() function to perform this task.

Skeleton code: [IcmpHelperLibrary.py](https://canvas.oregonstate.edu/courses/1884977/files/91314598?wrap=1)[Download IcmpHelperLibrary.py](https://canvas.oregonstate.edu/courses/1884977/files/91314598/download?download_frd=1)

Additional Notes

* This lab requires the use of raw sockets. In some operating systems, you may need administrator/root privileges to be able to run your Traceroute program.
* See below for more information on ICMP header.
* This will not work for websites that block ICMP traffic.
* You may have to turn your firewall or antivirus software off to allow the messages to be sent and received.
* After an initial run of 10 or so replies in a trace route, you may start to get timeouts. Let the program run for a minute or so and then terminate with ctrl-c.
* It is recommended you use the ping and trace route programs with your operating system to get more familiar with what to expect. Just note, we are not looking to replicate the output since our goal is to explore the ICMP protocol.

ICMP Header

The ICMP header starts after bit 160 of the IP header (unless IP options are used). 

* **Type**- ICMP type.
* **Code**- Subtype to the given ICMP type.
* **Checksum**- Error checking data calculated from the ICMP header + data, with value 0 for this field.
* **ID**- An ID value, should be returned in the case of echo reply.
* **Sequence**- A sequence value, should be returned in the case of echo reply.

Echo Request

The echo request is an ICMP message whose data is expected to be received back in an echo reply ("pong"). The host must respond to all echo requests with an echo reply containing the exact data received in the request message.

* Type must be set to 8.
* Code must be set to 0.
* The Identifier and Sequence Number can be used by the client to match the reply with the request that caused the reply. In practice, most Linux systems use a unique identifier for every ping process, and sequence number is an increasing number within that process. Windows uses a fixed identifier, which varies between Windows versions, and a sequence number that is only reset at boot time.
* The data received by the echo request must be entirely included in the echo reply.

Echo Reply

The echo reply is an ICMP message generated in response to an echo request, and is mandatory for all hosts and routers.

* Type and code must be set to 0.
* The identifier and sequence number can be used by the client to determine which echo requests are associated with the echo replies.
* The data received in the echo request must be entirely included in the echo reply.

Before Starting This Assignment

Please update your operating system firewall rules to allow for additional ICMP messages.

Windows: [https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/create-an-inbound-icmp-rule (Links to an external site.)](https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/create-an-inbound-icmp-rule)

What to turn in

* In the Word doc:
  + Include instructions on how to run your programs. Are they python3?
  + Include screenshots of running your trace route code for four different hosts with at least two on different continents.
  + Include comments / questions (optional)
* In your code listings:
  + Include sources you used (web pages, tutorials, books, etc)
  + Comment your code

Export your doc or docx file as pdf and upload it on Canvas. Separately upload your code file(s).