COSC 370 Programming 1: Socket Programming: ICMP Pinger

In this program assignment, you will gain a better understanding of Internet Control Message Protocol (ICMP). You will learn to implement a Ping application using ICMP request and reply messages.

Ping is a computer network application used to test whether a particular host is reachable across an IP network. It is also used to self-test the network interface card of the computer or as a latency test. It works by sending ICMP “echo reply” packets to the target host and listening for ICMP “echo reply” replies. The "echo reply" is sometimes called a pong. Ping measures the round-trip time, records packet loss, and prints a statistical summary of the echo reply packets received (the minimum, maximum, and the mean of the round-trip times and in some versions the standard deviation of the mean).

Your task is to develop your own Ping application. Your application will use ICMP but, in order to keep it simple, will not exactly follow the official specification in RFC 1739. Note that you will only need to write the client side of the program, as the functionality needed on the server side is built into almost all operating systems.

You should complete the Ping application so that it sends ping requests to a specified host separated by approximately one second. Each message contains a payload of data that includes a timestamp. After sending each packet, the application waits up to one second to receive a reply. If one second goes by without a reply from the server, then the client assumes that either the ping packet or the pong packet was lost in the network (or that the server is down).

Notes:

- You can implement your pinger in any language (eg. C, C++, Java, Python) you are familiar with.
- Your program 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).
- Your program can detect timeouts in receiving ICMP echo responses.
- This assignment requires the use of raw sockets. In some operating systems, you may need administrator/root privileges to be able to run your Pinger program.
- See the end of this programming assignment for more information on ICMP in Appendix.

Tasks:

1. Complete the client code

2. Test your client by sending packets to localhost, that is, 127.0.0.1.

3. See how your Pinger application communicates across the network by pinging servers in different continents and screenshots of your Pinger output for four target hosts, each on a different continent.

Extra credits:

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.

Requirements for Submission:

This is a group program assignment with 2-3 students in a group. You must hand in
1. Design/pseudocode/hierarchy chart/description of your program
2. A printed copy of your source code.
3. Your program running environment and executing commands
4. Sample runs of your program
5. Program report:
    a. One group report: (1) state clearly if your group has finished all required tasks for the program. If your group doesn't finish the program or some part of that doesn't work, the report must state clearly what the problem is. If you get any help from other resources (eg. Internet), you need to give the reference links. (2) Your group needs to records the group meeting times and hours. (3) You may also tell the instructor in your report any interesting feature you have implemented.
    b. One individual report for each student: (1) what your individual task and accomplishment, (2) how many hours and efforts each individual put on the program, (3) what most difficult things you ever encounter and how you conquer them, (4) and what do you learn from the program.

Each group needs to submit one copy of your source code through MyClasses @ SU

Grading Rubric:

Documentation of your program 20
Comments/readability of program 10
Workable Program 50
Demo of your program 20

Bonus Points: 10
Your program implements extra credits or optional features.

Appendix: Internet Control Message Protocol (ICMP)

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

Bits	160-167	168-175	176-183	184-191
160	Type	Code	Checksum	
192	ID		Sequence	

• **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.