pingroute

Release 1.0

Caleb Naeger

Mar 01, 2025

CONTENTS:

CHAPTER

ONE

HOMEWORK2

1.1 my_ping module

my_ping.py A simple implementation of some of the functionality of the Unix ping utility.

Caleb Naeger - cmn4315@rit.edu Foundations of Computer Networks

usage: my_ping.py [-h] [-c C] [-i I] [-s S] [-t T] dst

positional arguments:
dst The destination hostname or IP address to ping

options:

-h, --help	show this help message and exit
-c C	The number of ping packets to send. Will send indefinitely if not provided
-i I	The interval to wait between each packet, defaults to 1s
-s S	The payload size in bytes, defaults to 56 bytes.
-t T	Timeout duration for a run. Will run until interrupt if not provided

`my_ping.`**`calculate_checksum`**`(`*data*`)`

Calculate the ICMP checksum for a given piece of data and return the calculated checksum. :param data: the data for which to calculate the checksum

`my_ping.`**`create_packet`**`(`*packetsize: int = 56*`)`

Create ICMP packet to be used and return the created packet. Packet creation code adapted from https://denizhalil.com/2024/04/06/sending-icmp-packets-with-python-socket-adventure-in-signaling/ :param packetsize: the number of data bytes to put in the packet

`my_ping.`**`main`**`(`*ip: str*, *timeout: int*, *packetsize: int*, *count: int*, *wait: int*`)`

The main function for my_ping.py, controlling the sending of pings, aggregation and output of results, and timeouts/end conditions. :param ip: the dst provided by the user. :param timeout: total time in seconds for which the program should run. :param packetsize: the number of data bytes to add to the ping packet. :param count: the total number of pings to send. :param wait: time in seconds to wait between each ping.

`my_ping.`**`ping`**`(`*packet: bytes*, *ip: str = '8.8.8.8'*`)`

Send a ping to a given IP, timing the response time. Returns a Tuple(bool, float) representing whether the response was lost and the total time in seconds for the ping. On loss, response time will be -1. :param packet: the packet to send. :param ip: the IP address to which to send the packet.

1.2 my_traceroute module

my_traceroute.py A simple implementation of a subset of the Unix traceroute functionality

Caleb Naeger - cmn4315@rit.edu Foundations of Computer Networks

Usage: my_traceroute.py [-h] [-n] [-q Q] [-S] dst

positional arguments:

dst The destination hostname or IP address to trace
the route for

options:

-h, --help	show this help message and exit
-n	Print hop addresses numerically rather than symbolically and numerically
-q Q	Set the number of probes to send per TTL
-S	Print a summary of unanswered probes per hop

`my_traceroute.`**`calculate_checksum`**`(`*data*`)`

Calculate the ICMP checksum for a given piece of data and return the calculated checksum. :param data: the data for which to calculate the checksum

`my_traceroute.`**`create_packet`**`(`*packetsize: int = 40*`)`

Create ICMP packet to be used and return the created packet. Packet creation code adapted from https://denizhalil.com/2024/04/06/sending-icmp-packets-with-python-socket-adventure-in-signaling/ :param packetsize: the number of data bytes to put in the packet

`my_traceroute.`**`get_hostname`**`(`*ip*`)`

Attempt to resolve the hostname for a given ip address, returning the ip address on failure and the hostname on success. :param ip: the IP address for which to resolve the hostname

`my_traceroute.`**`main`**`(`*ip: str*, *probes: int*, *n: bool*, *s: bool*, *hops: int = 64*`)`

Main function for traceroute. This function handles the control flow of the program to send each probe with increasing TTL and print the results as they are received. :param ip: the dst provided by the user to the program. :param probes: the number of probes to send per hop. :param n: flag determining whether hostnames will be shown (where possible) for each hop IP. :param s: flag denoting whether to show loss summaries for each hop. :param hops: maximum number of hops to attempt before prematurely ending the program.

`my_traceroute.`**`probe`**`(`*packet: bytes*, *ttl: int*, *ip: str = '8.8.8.8'*`)`

Send one traceroute probe to the destination address and return a Tuple(bool, float, str) representing whether a response was received, the total elapsed time to send and receive a response, and the source IP of the response packet respectively. On a failure, response time will be -1 and the response source IP will be “”. :param packet: the packet to send for the probe :param ttl: the Time To Live to be associated with the packet :param ip: the IP address to which to send the packet.

PYTHON MODULE INDEX

INDEX