

DESIGN DOCUMENT

P1: RTT

Problem Statement - Write a C program to find RTT values for a list of IP addresses listed in a file as fast as possible:

1. Your program should take a text file with one IP(v4/v6)address per line as command line argument
2. Program should use a proper method to compute correct RTTvalues.
3. Program should use all possible methods to increase throughput (number of ips/time).

Usage

The C code can be compiled using gcc and run. It takes filename of the file containing list of ip addresses as an argument and must be run with elevated privileges using sudo:

```
gcc rtt.c -o rtt
sudo ./rtt filename
```

Design

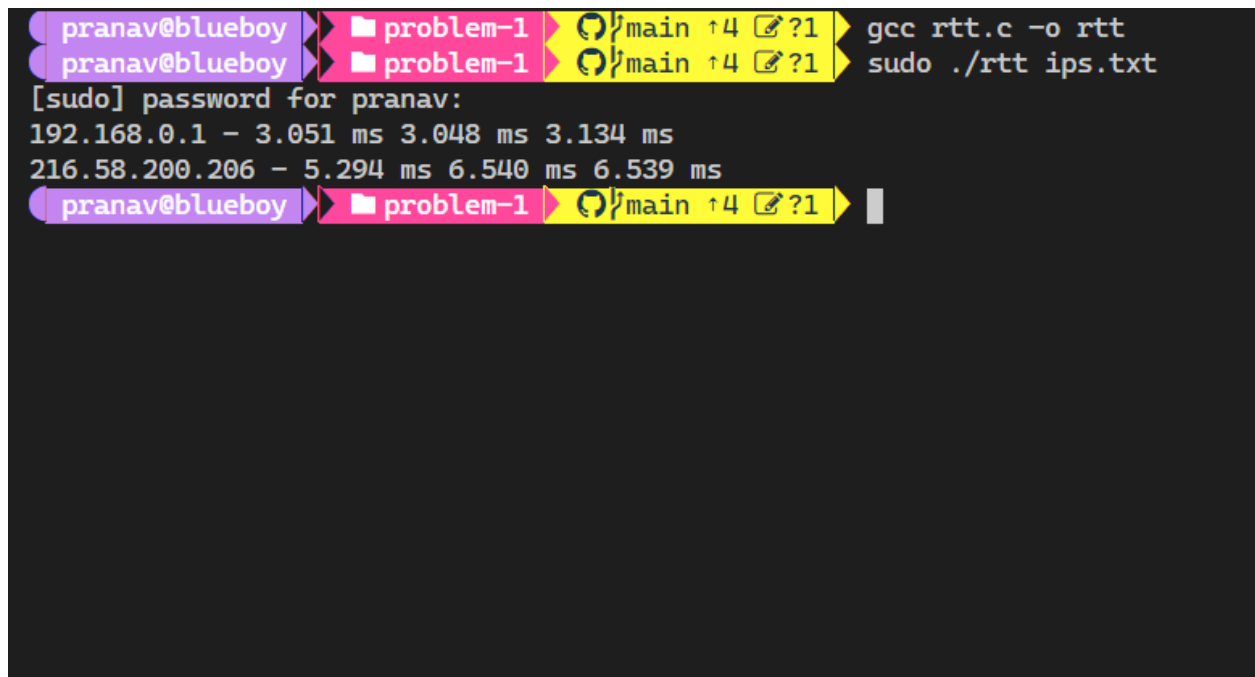
The file of IP addresses is read line by line. A raw socket is created for each IP address and connected to it. The RTT values are calculated using ICMP echo messages. For each IP address we send 3 echo request messages. Each message contains the time that it was sent at, in the body, the process id as identifier and the sequence number as one of 0, 1 or 2. The socket, address structure and other information is added to a structure and which is stored for future reference in an array. We use epoll to ensure the highest throughput in our application. An epoll instance is created and each IP

address is added to it to be monitored.

Then the program waits on the epoll instance and processes all incoming ICMP echo messages from the IP addresses. It calculates the RTT time from the time received in the body of the message and the current time, and stores these values. When all 3 messages have arrived, it prints the 3 RTT values with the IP address. This keeps going until RTT values have been displayed for all IP addresses.

To accommodate both IPv4 and IPv6, separate functions have been made for sending and receiving messages for both protocols. Using the information obtained from `getaddrinfo()`, the appropriate functions are selected.

Screenshot



```
pranav@blueboy > problem-1 > main ↑4 ?1 gcc rtt.c -o rtt
pranav@blueboy > problem-1 > main ↑4 ?1 sudo ./rtt ips.txt
[sudo] password for pranav:
192.168.0.1 - 3.051 ms 3.048 ms 3.134 ms
216.58.200.206 - 5.294 ms 6.540 ms 6.539 ms
pranav@blueboy > problem-1 > main ↑4 ?1
```