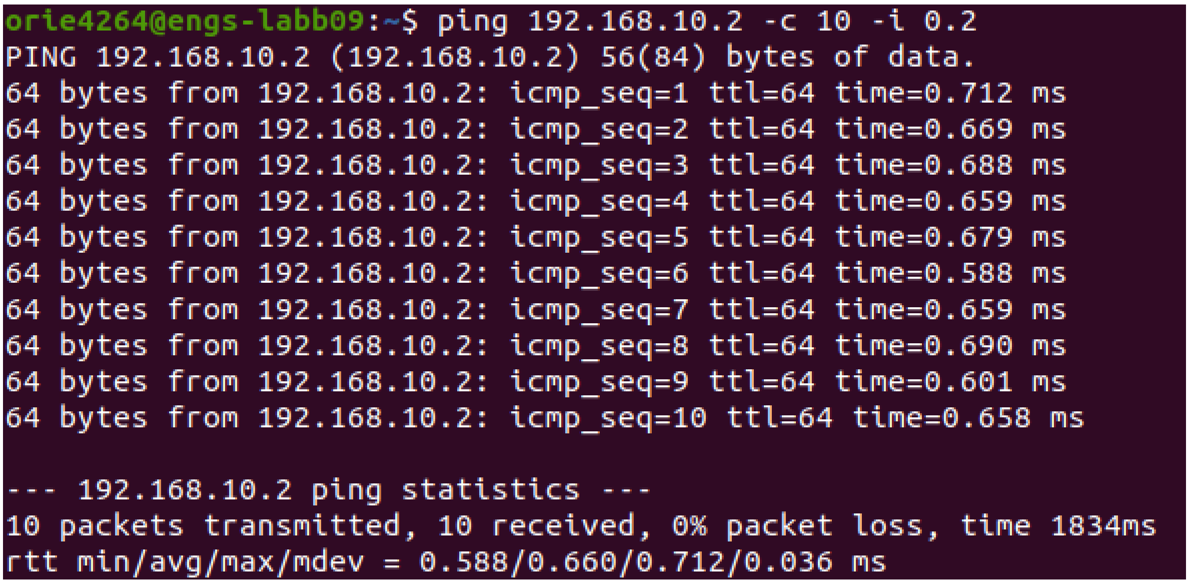
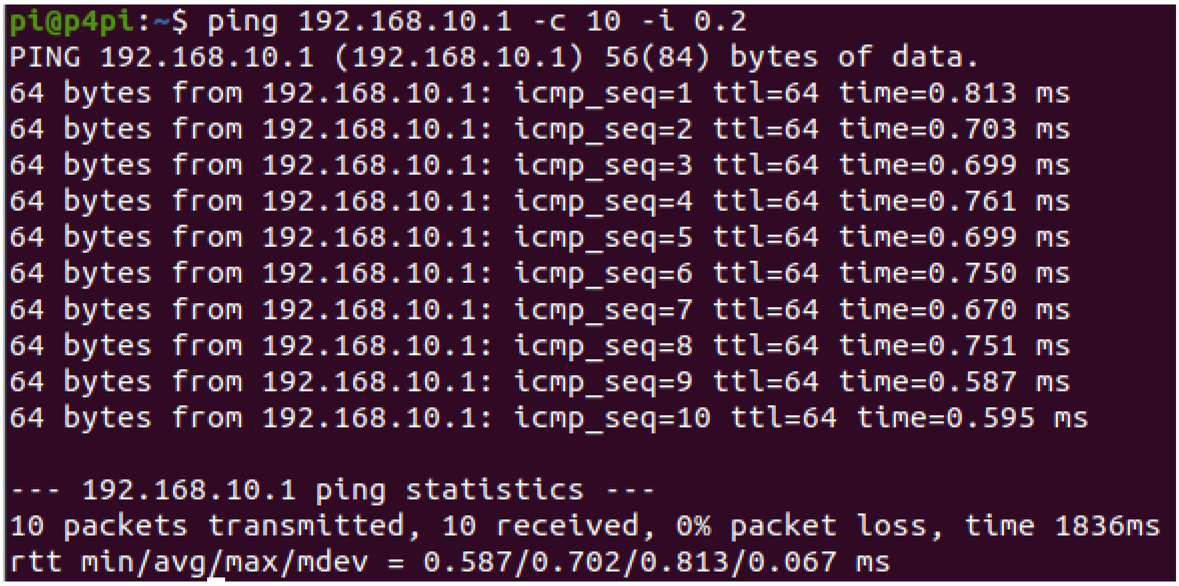
Lab machine pinging raspberry pi
10 counts, 0.2s interval
min/avg/max/mdev = 0.588/0.660/0.712/0.036ms



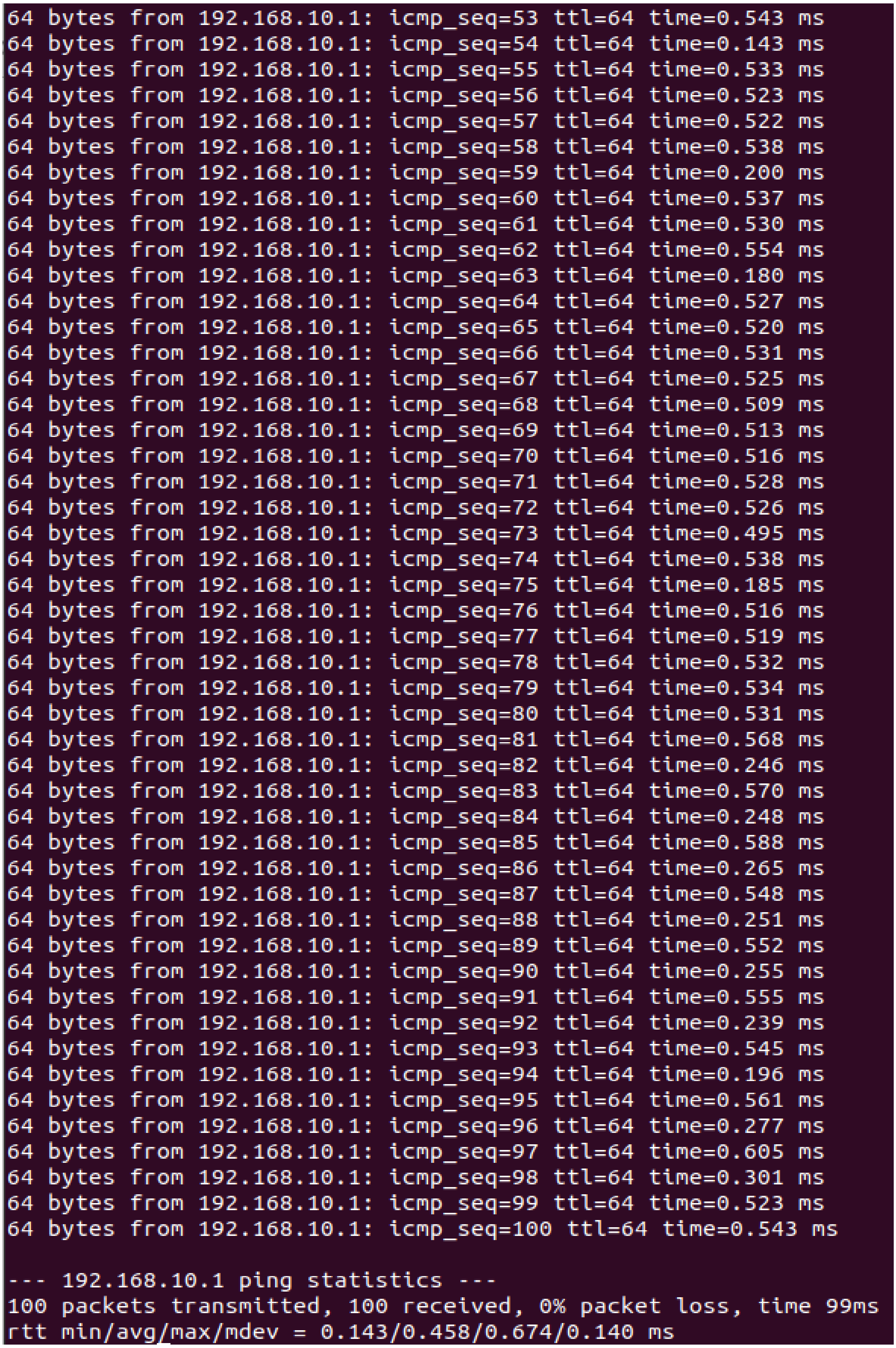
```
orie4264@engs-labb09:~$ ping 192.168.10.2 -c 10 -i 0.2
PING 192.168.10.2 (192.168.10.2) 56(84) bytes of data.
64 bytes from 192.168.10.2: icmp_seq=1 ttl=64 time=0.712 ms
64 bytes from 192.168.10.2: icmp_seq=2 ttl=64 time=0.669 ms
64 bytes from 192.168.10.2: icmp_seq=3 ttl=64 time=0.688 ms
64 bytes from 192.168.10.2: icmp_seq=4 ttl=64 time=0.659 ms
64 bytes from 192.168.10.2: icmp_seq=5 ttl=64 time=0.679 ms
64 bytes from 192.168.10.2: icmp_seq=6 ttl=64 time=0.588 ms
64 bytes from 192.168.10.2: icmp_seq=7 ttl=64 time=0.659 ms
64 bytes from 192.168.10.2: icmp_seq=8 ttl=64 time=0.690 ms
64 bytes from 192.168.10.2: icmp_seq=9 ttl=64 time=0.601 ms
64 bytes from 192.168.10.2: icmp_seq=10 ttl=64 time=0.658 ms

--- 192.168.10.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 1834ms
rtt min/avg/max/mdev = 0.588/0.660/0.712/0.036 ms
```

Raspberry pi pinging lab machine
10 counts, 0.2s interval
min/avg/max/mdev = 0.587/0.702/0.813/0.067ms



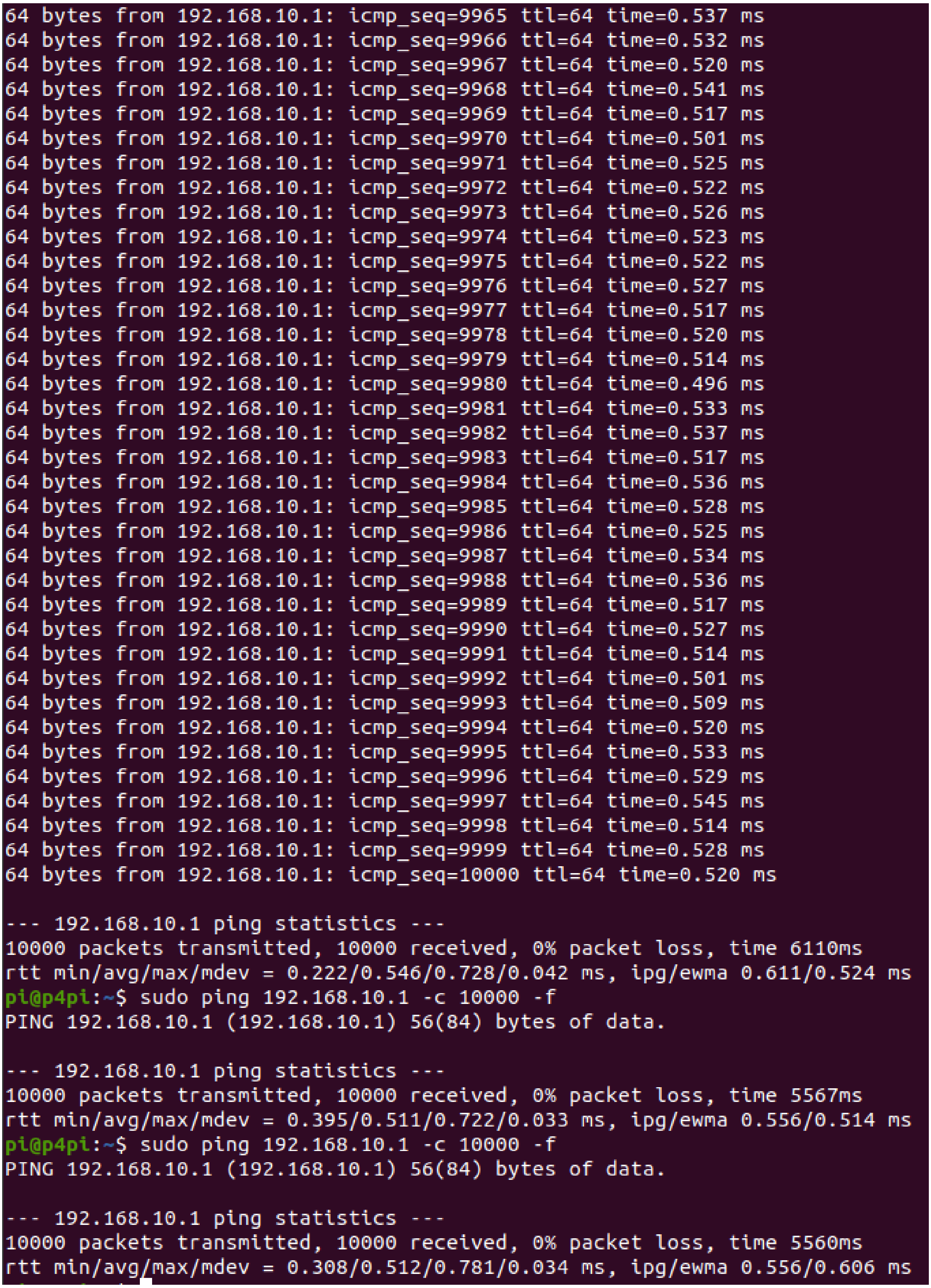
```
pi@p4pi:~$ ping 192.168.10.1 -c 10 -i 0.2
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=0.813 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=0.703 ms
64 bytes from 192.168.10.1: icmp_seq=3 ttl=64 time=0.699 ms
64 bytes from 192.168.10.1: icmp_seq=4 ttl=64 time=0.761 ms
64 bytes from 192.168.10.1: icmp_seq=5 ttl=64 time=0.699 ms
64 bytes from 192.168.10.1: icmp_seq=6 ttl=64 time=0.750 ms
64 bytes from 192.168.10.1: icmp_seq=7 ttl=64 time=0.670 ms
64 bytes from 192.168.10.1: icmp_seq=8 ttl=64 time=0.751 ms
64 bytes from 192.168.10.1: icmp_seq=9 ttl=64 time=0.587 ms
64 bytes from 192.168.10.1: icmp_seq=10 ttl=64 time=0.595 ms

--- 192.168.10.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 1836ms
rtt min/avg/max/mdev = 0.587/0.702/0.813/0.067 ms
```

Raspberry pi pinging lab machine
100 counts, 0.001s interval (sudo)
min/avg/max/mdev = 0.143/0.458/0.674/0.140ms



```
64 bytes from 192.168.10.1: icmp_seq=53 ttl=64 time=0.543 ms
64 bytes from 192.168.10.1: icmp_seq=54 ttl=64 time=0.143 ms
64 bytes from 192.168.10.1: icmp_seq=55 ttl=64 time=0.533 ms
64 bytes from 192.168.10.1: icmp_seq=56 ttl=64 time=0.523 ms
64 bytes from 192.168.10.1: icmp_seq=57 ttl=64 time=0.522 ms
64 bytes from 192.168.10.1: icmp_seq=58 ttl=64 time=0.538 ms
64 bytes from 192.168.10.1: icmp_seq=59 ttl=64 time=0.200 ms
64 bytes from 192.168.10.1: icmp_seq=60 ttl=64 time=0.537 ms
64 bytes from 192.168.10.1: icmp_seq=61 ttl=64 time=0.530 ms
64 bytes from 192.168.10.1: icmp_seq=62 ttl=64 time=0.554 ms
64 bytes from 192.168.10.1: icmp_seq=63 ttl=64 time=0.180 ms
64 bytes from 192.168.10.1: icmp_seq=64 ttl=64 time=0.527 ms
64 bytes from 192.168.10.1: icmp_seq=65 ttl=64 time=0.520 ms
64 bytes from 192.168.10.1: icmp_seq=66 ttl=64 time=0.531 ms
64 bytes from 192.168.10.1: icmp_seq=67 ttl=64 time=0.525 ms
64 bytes from 192.168.10.1: icmp_seq=68 ttl=64 time=0.509 ms
64 bytes from 192.168.10.1: icmp_seq=69 ttl=64 time=0.513 ms
64 bytes from 192.168.10.1: icmp_seq=70 ttl=64 time=0.516 ms
64 bytes from 192.168.10.1: icmp_seq=71 ttl=64 time=0.528 ms
64 bytes from 192.168.10.1: icmp_seq=72 ttl=64 time=0.526 ms
64 bytes from 192.168.10.1: icmp_seq=73 ttl=64 time=0.495 ms
64 bytes from 192.168.10.1: icmp_seq=74 ttl=64 time=0.538 ms
64 bytes from 192.168.10.1: icmp_seq=75 ttl=64 time=0.185 ms
64 bytes from 192.168.10.1: icmp_seq=76 ttl=64 time=0.516 ms
64 bytes from 192.168.10.1: icmp_seq=77 ttl=64 time=0.519 ms
64 bytes from 192.168.10.1: icmp_seq=78 ttl=64 time=0.532 ms
64 bytes from 192.168.10.1: icmp_seq=79 ttl=64 time=0.534 ms
64 bytes from 192.168.10.1: icmp_seq=80 ttl=64 time=0.531 ms
64 bytes from 192.168.10.1: icmp_seq=81 ttl=64 time=0.568 ms
64 bytes from 192.168.10.1: icmp_seq=82 ttl=64 time=0.246 ms
64 bytes from 192.168.10.1: icmp_seq=83 ttl=64 time=0.570 ms
64 bytes from 192.168.10.1: icmp_seq=84 ttl=64 time=0.248 ms
64 bytes from 192.168.10.1: icmp_seq=85 ttl=64 time=0.588 ms
64 bytes from 192.168.10.1: icmp_seq=86 ttl=64 time=0.265 ms
64 bytes from 192.168.10.1: icmp_seq=87 ttl=64 time=0.548 ms
64 bytes from 192.168.10.1: icmp_seq=88 ttl=64 time=0.251 ms
64 bytes from 192.168.10.1: icmp_seq=89 ttl=64 time=0.552 ms
64 bytes from 192.168.10.1: icmp_seq=90 ttl=64 time=0.255 ms
64 bytes from 192.168.10.1: icmp_seq=91 ttl=64 time=0.555 ms
64 bytes from 192.168.10.1: icmp_seq=92 ttl=64 time=0.239 ms
64 bytes from 192.168.10.1: icmp_seq=93 ttl=64 time=0.545 ms
64 bytes from 192.168.10.1: icmp_seq=94 ttl=64 time=0.196 ms
64 bytes from 192.168.10.1: icmp_seq=95 ttl=64 time=0.561 ms
64 bytes from 192.168.10.1: icmp_seq=96 ttl=64 time=0.277 ms
64 bytes from 192.168.10.1: icmp_seq=97 ttl=64 time=0.605 ms
64 bytes from 192.168.10.1: icmp_seq=98 ttl=64 time=0.301 ms
64 bytes from 192.168.10.1: icmp_seq=99 ttl=64 time=0.523 ms
64 bytes from 192.168.10.1: icmp_seq=100 ttl=64 time=0.543 ms

--- 192.168.10.1 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 99ms
rtt min/avg/max/mdev = 0.143/0.458/0.674/0.140 ms
```

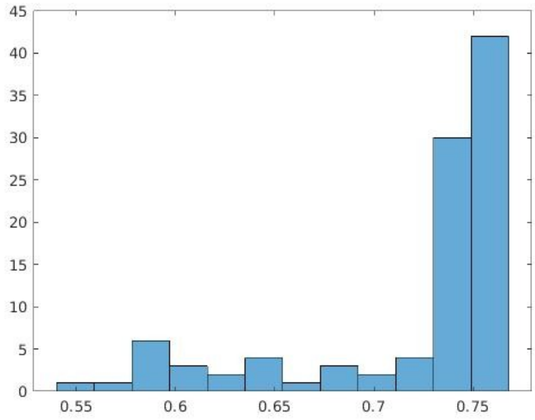
Raspberry pi pinging lab machine
10,000 counts, flooding (sudo)
Data below shows 3 different runs. Found out that not using -i will not display individual pings.

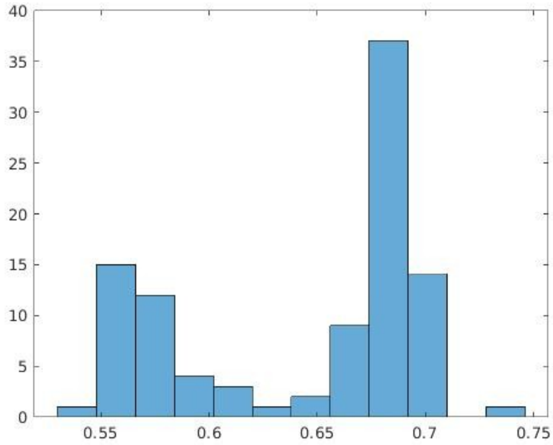


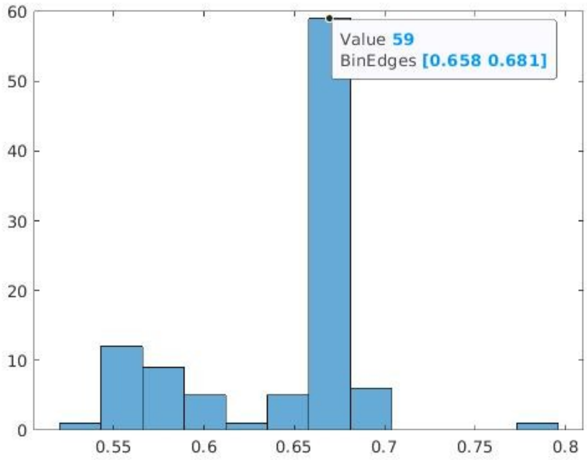
```
64 bytes from 192.168.10.1: icmp_seq=9965 ttl=64 time=0.537 ms
64 bytes from 192.168.10.1: icmp_seq=9966 ttl=64 time=0.532 ms
64 bytes from 192.168.10.1: icmp_seq=9967 ttl=64 time=0.520 ms
64 bytes from 192.168.10.1: icmp_seq=9968 ttl=64 time=0.541 ms
64 bytes from 192.168.10.1: icmp_seq=9969 ttl=64 time=0.517 ms
64 bytes from 192.168.10.1: icmp_seq=9970 ttl=64 time=0.501 ms
64 bytes from 192.168.10.1: icmp_seq=9971 ttl=64 time=0.525 ms
64 bytes from 192.168.10.1: icmp_seq=9972 ttl=64 time=0.522 ms
64 bytes from 192.168.10.1: icmp_seq=9973 ttl=64 time=0.526 ms
64 bytes from 192.168.10.1: icmp_seq=9974 ttl=64 time=0.523 ms
64 bytes from 192.168.10.1: icmp_seq=9975 ttl=64 time=0.522 ms
64 bytes from 192.168.10.1: icmp_seq=9976 ttl=64 time=0.527 ms
64 bytes from 192.168.10.1: icmp_seq=9977 ttl=64 time=0.517 ms
64 bytes from 192.168.10.1: icmp_seq=9978 ttl=64 time=0.520 ms
64 bytes from 192.168.10.1: icmp_seq=9979 ttl=64 time=0.514 ms
64 bytes from 192.168.10.1: icmp_seq=9980 ttl=64 time=0.496 ms
64 bytes from 192.168.10.1: icmp_seq=9981 ttl=64 time=0.533 ms
64 bytes from 192.168.10.1: icmp_seq=9982 ttl=64 time=0.537 ms
64 bytes from 192.168.10.1: icmp_seq=9983 ttl=64 time=0.517 ms
64 bytes from 192.168.10.1: icmp_seq=9984 ttl=64 time=0.536 ms
64 bytes from 192.168.10.1: icmp_seq=9985 ttl=64 time=0.528 ms
64 bytes from 192.168.10.1: icmp_seq=9986 ttl=64 time=0.525 ms
64 bytes from 192.168.10.1: icmp_seq=9987 ttl=64 time=0.534 ms
64 bytes from 192.168.10.1: icmp_seq=9988 ttl=64 time=0.536 ms
64 bytes from 192.168.10.1: icmp_seq=9989 ttl=64 time=0.517 ms
64 bytes from 192.168.10.1: icmp_seq=9990 ttl=64 time=0.527 ms
64 bytes from 192.168.10.1: icmp_seq=9991 ttl=64 time=0.514 ms
64 bytes from 192.168.10.1: icmp_seq=9992 ttl=64 time=0.501 ms
64 bytes from 192.168.10.1: icmp_seq=9993 ttl=64 time=0.509 ms
64 bytes from 192.168.10.1: icmp_seq=9994 ttl=64 time=0.520 ms
64 bytes from 192.168.10.1: icmp_seq=9995 ttl=64 time=0.533 ms
64 bytes from 192.168.10.1: icmp_seq=9996 ttl=64 time=0.529 ms
64 bytes from 192.168.10.1: icmp_seq=9997 ttl=64 time=0.545 ms
64 bytes from 192.168.10.1: icmp_seq=9998 ttl=64 time=0.514 ms
64 bytes from 192.168.10.1: icmp_seq=9999 ttl=64 time=0.528 ms
64 bytes from 192.168.10.1: icmp_seq=10000 ttl=64 time=0.520 ms

--- 192.168.10.1 ping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 6110ms
rtt min/avg/max/mdev = 0.222/0.546/0.728/0.042 ms, ipg/ewma 0.611/0.524 ms
pi@p4pi:~$ sudo ping 192.168.10.1 -c 10000 -f
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.

--- 192.168.10.1 ping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 5567ms
rtt min/avg/max/mdev = 0.395/0.511/0.722/0.033 ms, ipg/ewma 0.556/0.514 ms
pi@p4pi:~$ sudo ping 192.168.10.1 -c 10000 -f
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.

--- 192.168.10.1 ping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 5560ms
rtt min/avg/max/mdev = 0.308/0.512/0.781/0.034 ms, ipg/ewma 0.556/0.606 ms
```

5. Raspberry Pi pinging lab machine

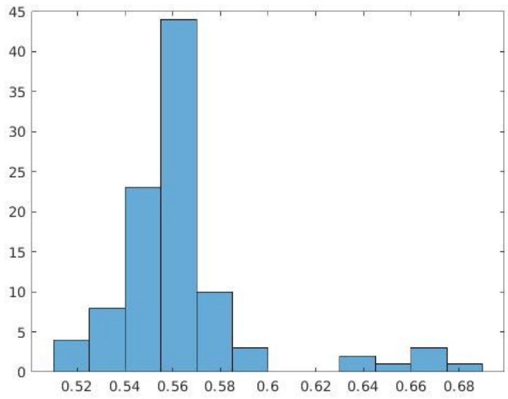
Initial 3 conditions
1000 measurements, intervals 0.01, 0.001 and 0.0001 respectively

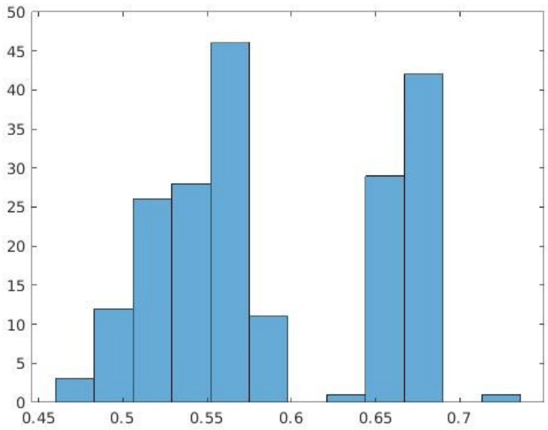






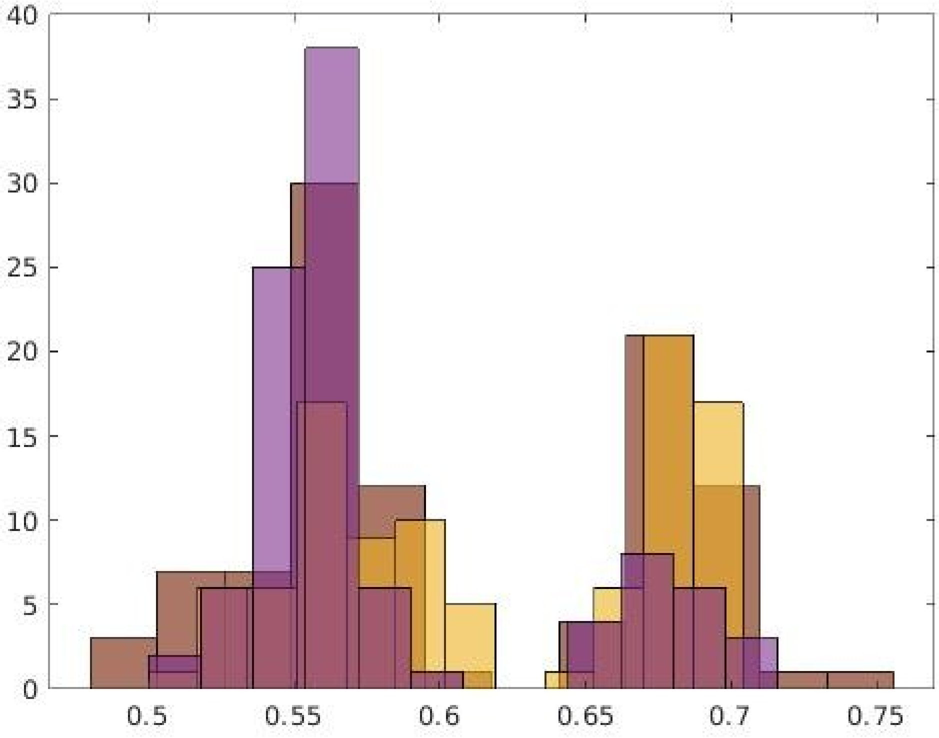
The trend shows that reducing intervals results in generally faster round trips
Rerunning the test with more runs as I reduce interval time (to match total run time, I.e 10,000 runs for 0.001s interval and 100,000 runs for 0.0001s interval respectively, left and right)



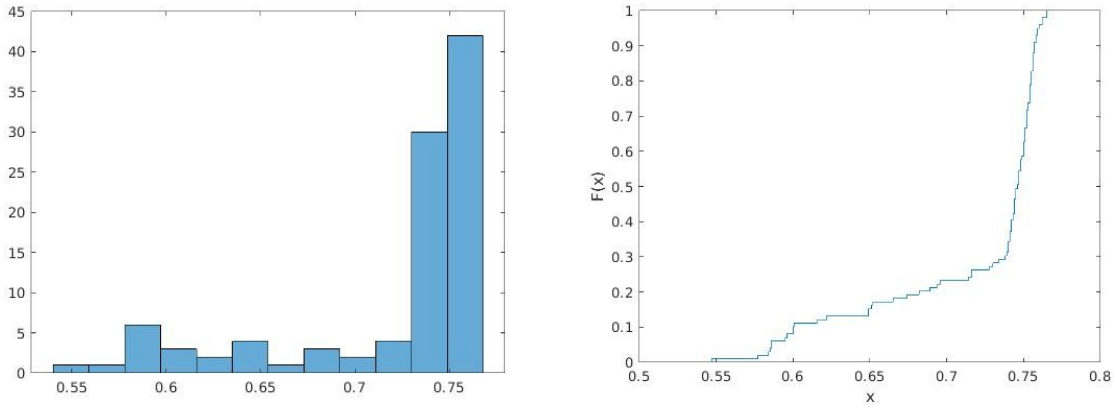


Increasing the total number of runs seemed to have shifted the graph towards the left compared to the original 1000 pings (reduced round trip time on average).

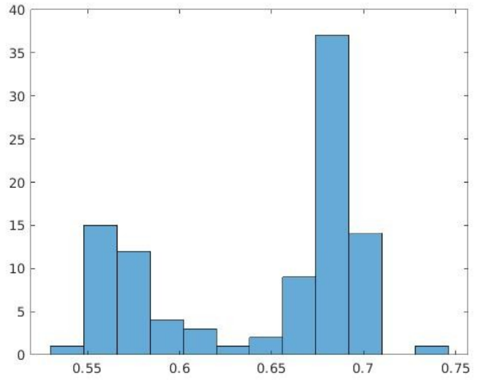
When running the same test three times (1000 round trips, 0.01s interval), there seems to be quite some variance in the tests itself, as seen in the picture below where 3 of the histograms are overlayed. Each histogram is run on the same conditions (1000 pings, 0.01s interval).

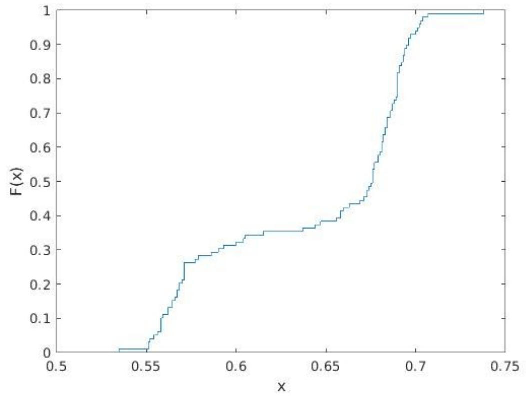


When put into CDF form, the graphs below are obtained. (Side by side with histogram)

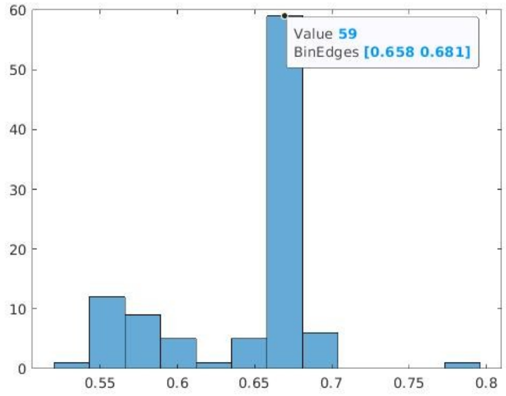


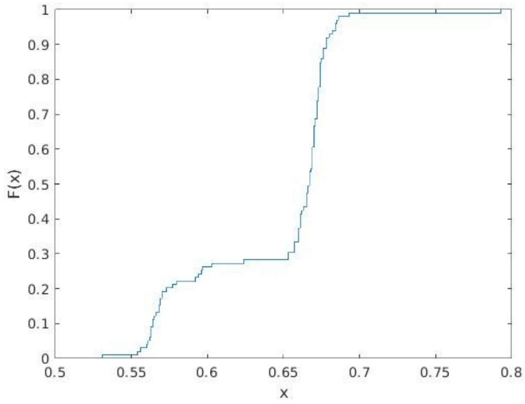
1000 pings, 0.01s interval. Large number of pings occur with round trip timings around 0.75ms
90% of round trip timings are below 0.76ms





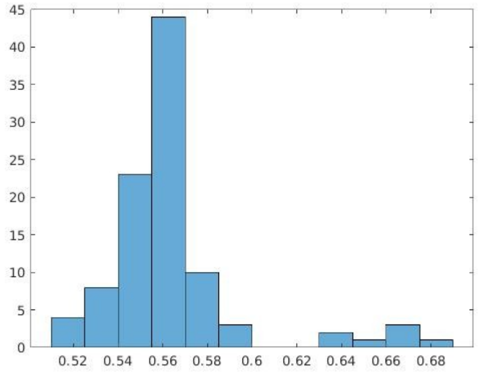
1000 pings, 0.001s interval. Round trip timings are mostly located around 0.675ms and 0.55ms. Overall on average it is faster than 0.01s intervals.
90% of round trip times are below 0.7ms

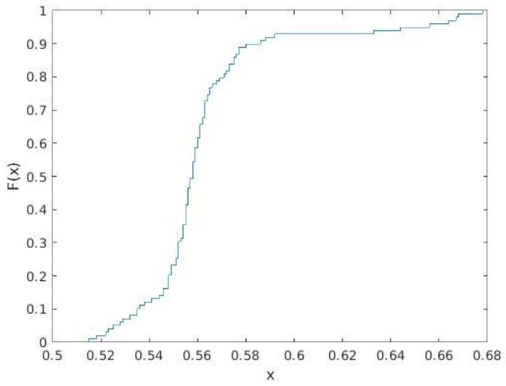




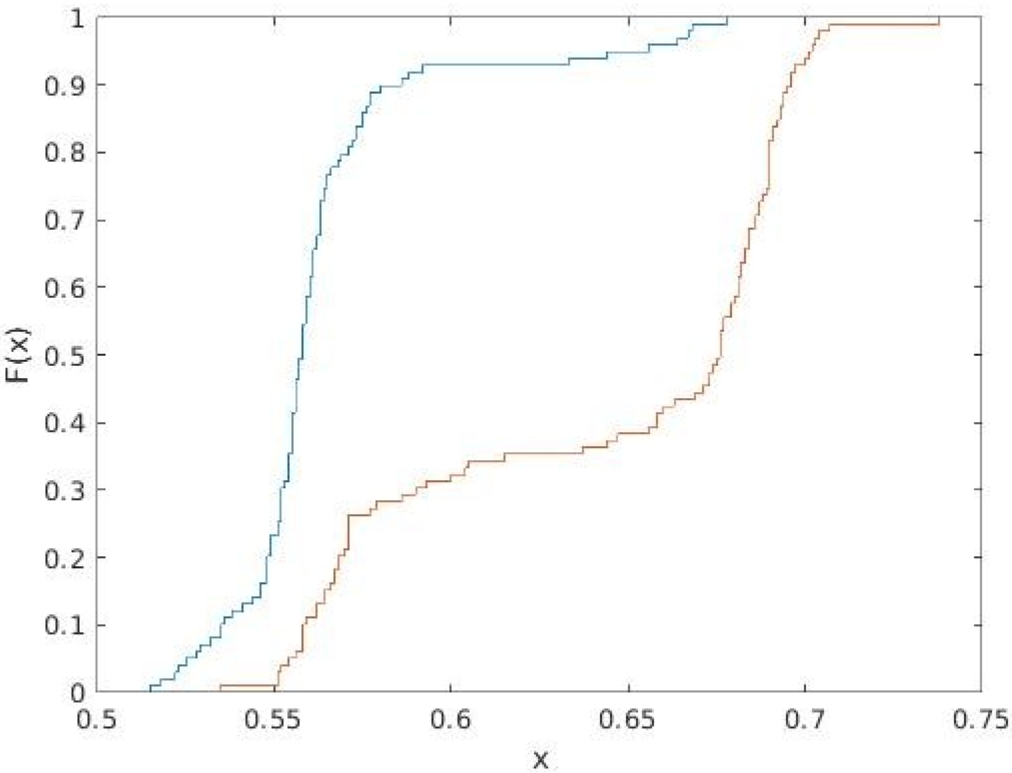
1000 pings, 0.0001s interval. Round trip timings is generally also skewed more towards the left than 0.001s interval.
90% of round trip times are below 0.67ms

Checking CDFs for greater ping numbers (10,000 pings 0.001s and 100,000 pings 0.0001s)

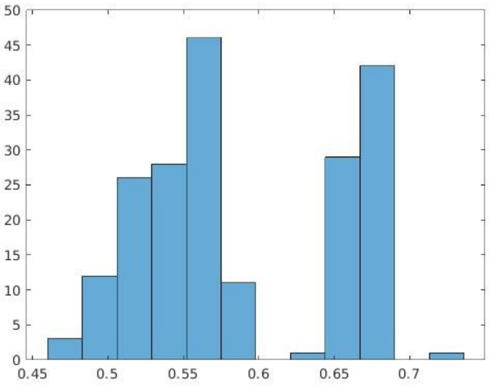


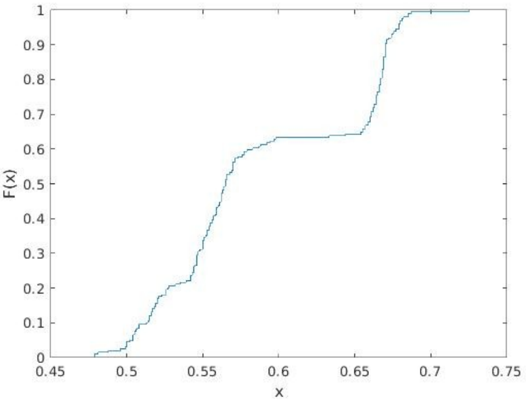


10,000 pings, 0.001s interval. 90% of round trip timings are below 0.59ms

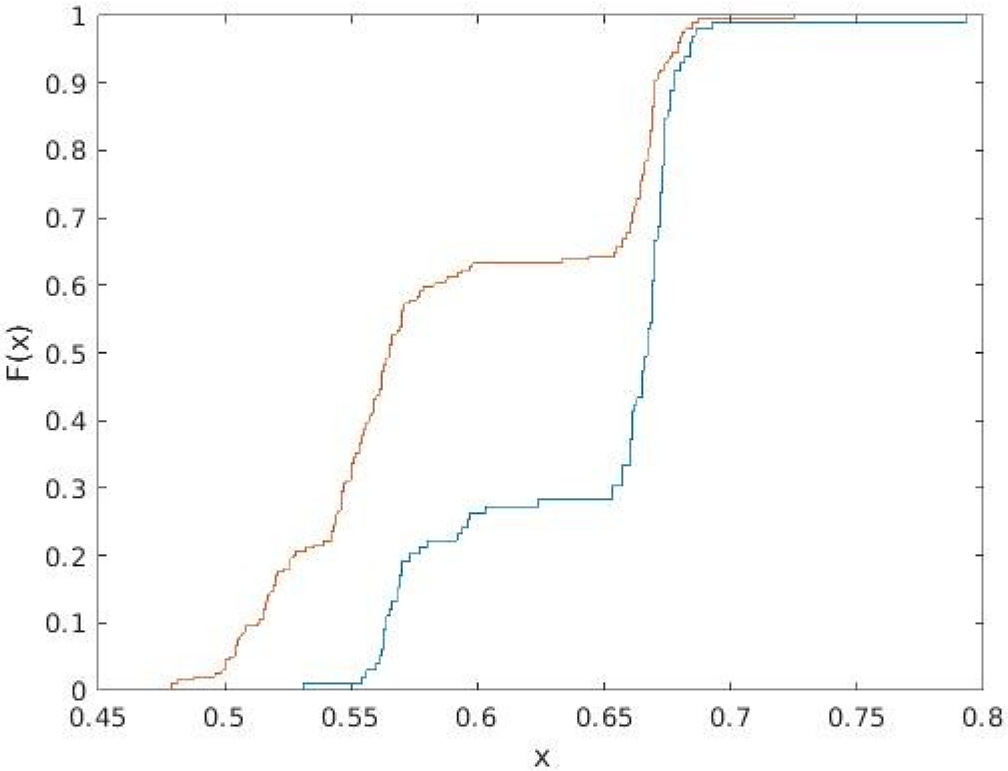


The comparison becomes obvious when plotted together. Blue lines = 10,000 pings, brown = 1000pings.





Same trends occurs for the 0.0001s interval, when raised to 100,000 pings (Graphs above) Comparison to 1000pings graph below, brown line = 100,000 pings. Blue line = 1000 pings, red-brown line = 100,000 pings.



Question: Why does different intervals lead to different round trip results
From the data, small intervals generally have lower round trip times (faster). One possibility might be due to the caching between intervals, which is fully utilised when the intervals are small and less when intervals are big (other processes might have cut in), optimising the switch paths that it can take between the 2 systems internally, assuming there are switches that need to be passed to record the pings within each system.

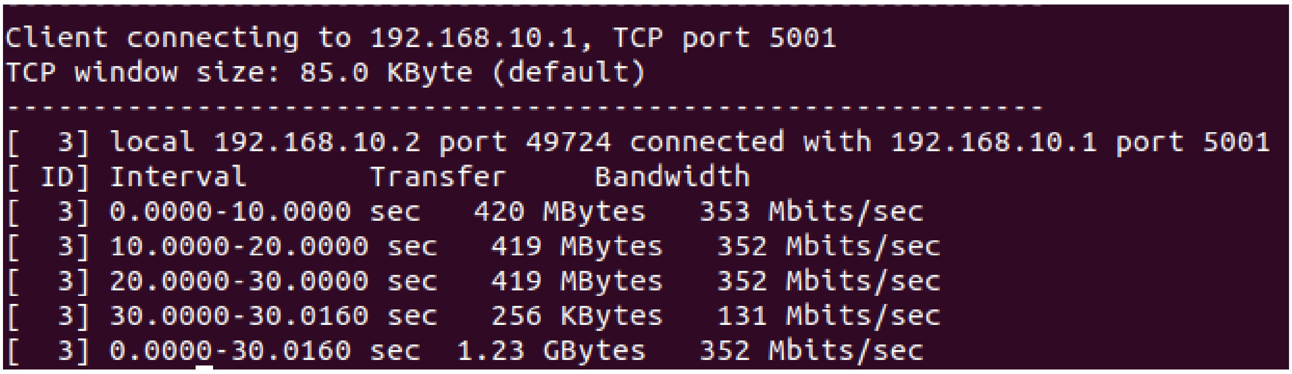
Another possibility might be due to lower interval pings may be sent in batches, and are just spaced out with a frame delay between them rather than having the processor send the pings individually, keeping a path constantly open just for the pings and allowing it to return faster as well. It may also reduce system latency factor if pings are recorded in batches since it only starts once per batch rather than multiple times.

Out of the parameters (min, max, mean etc.). I estimate that mode is the most accurate parameter to use to estimate system latency. This is due to some graphs having very skewed results which may not be reflective if we used mean. Min and Max also have too big of variances as delays can occur at the very start which will vary the numbers wildly across repeated experiments. Median shows where most of the round trip timings lie, which will more likely have shared characteristics which in this case would be the system latency at the lab machine before returning to the raspberry pi machine. As latency can vary somewhat, it would give a rough estimate of where the majority of the latency values were.

It will also give a slight overestimation since there are other overheads, giving us a slightly conservative estimate (larger latency).

Iperf section

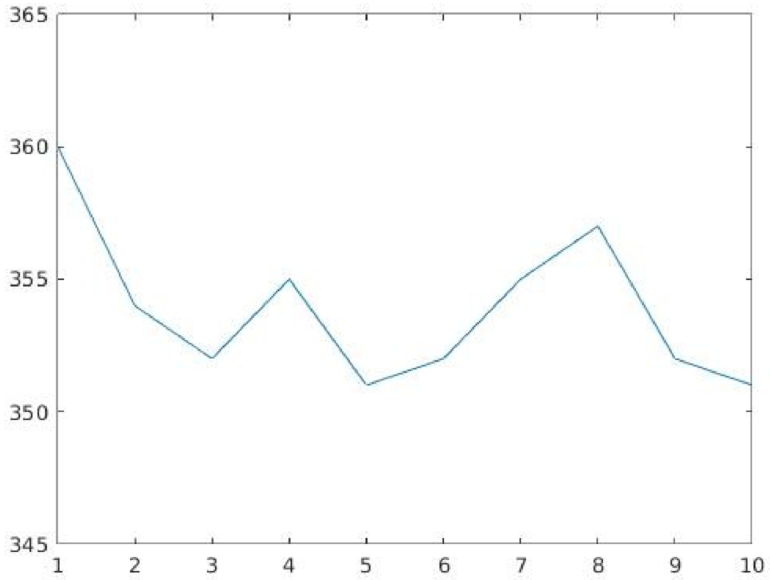
Setting lab machine as server, Raspberry Pi as client.
10 seconds for each experiment (30s to run 3 runs for reproducibility)



```
Client connecting to 192.168.10.1, TCP port 5001
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 192.168.10.2 port 49724 connected with 192.168.10.1 port 5001
[ ID] Interval        Transfer     Bandwidth
[  3] 0.0000-10.0000 sec   420 MBytes   353 Mbits/sec
[  3] 10.0000-20.0000 sec   419 MBytes   352 Mbits/sec
[  3] 20.0000-30.0000 sec   419 MBytes   352 Mbits/sec
[  3] 30.0000-30.0160 sec   256 KBytes   131 Mbits/sec
[  3] 0.0000-30.0160 sec  1.23 GBytes   352 Mbits/sec
```

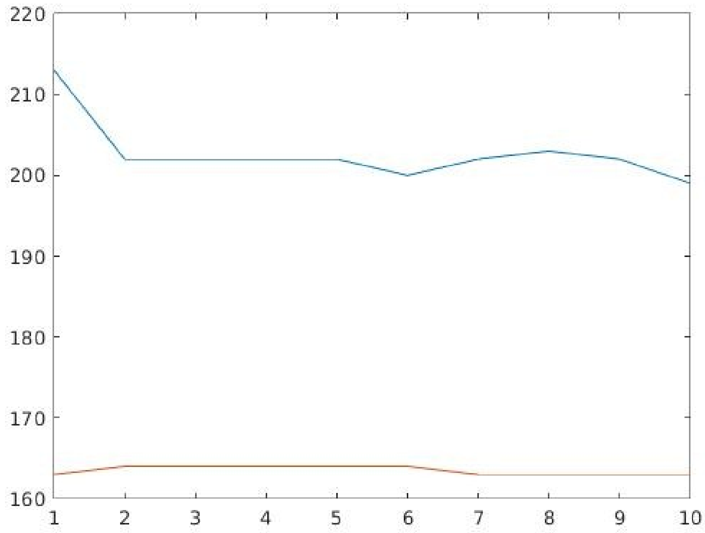
Effective bandwidth: Approximately 353 Mbits/sec.

Setting Raspberry Pi as server, lab machine as client



y-axis: Bandwidth in Mbits/s x-axis: time/s
Bandwidth generally hovered around 350-360Mbits/s for total bandwidth

Setting Raspberry pi as server, lab machine as client. Bi-directional iperf

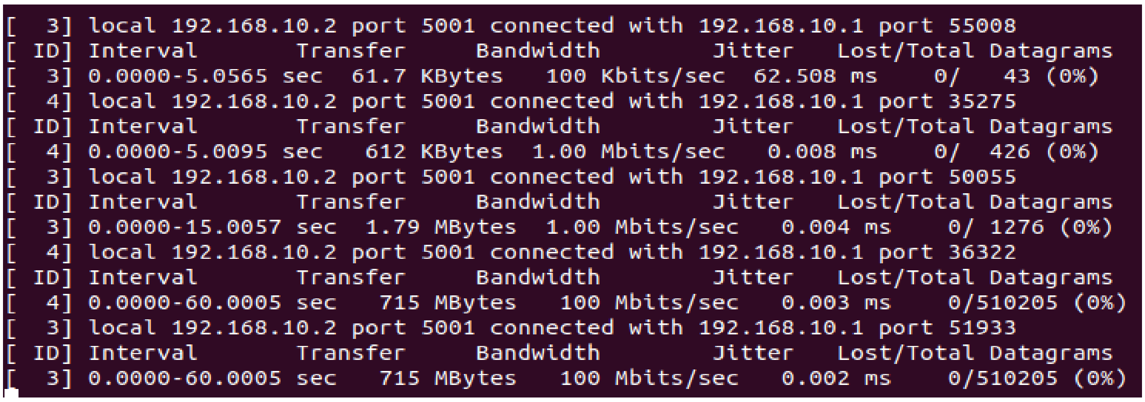


y-axis: Bandwidth in Mbits/s x-axis: time/s

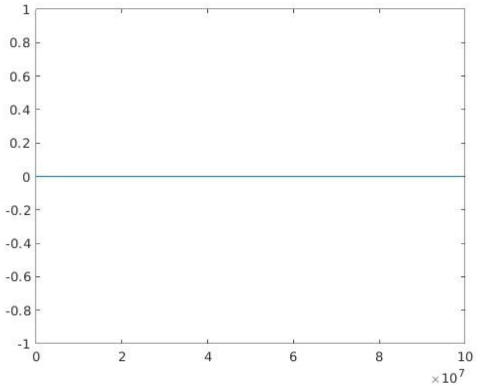
The blue line is the bandwidth from machine to raspberry pi, and orange line is bandwidth from raspberry pi to machine.

One way iperf UDP, lab machine to raspberry pi. 5 seconds long with varying bandwidth.
There are no packet losses (0%) over the varying bandwidths. After repeating each bandwidth test multiple times, there is still no packet losses. Packet losses only occur when pushing bit rate to 1000mb/s, where it saturates the channel capacity of the 1Gb/s ethernet cable. Losses then occur due to imperfections within the pathway itself.

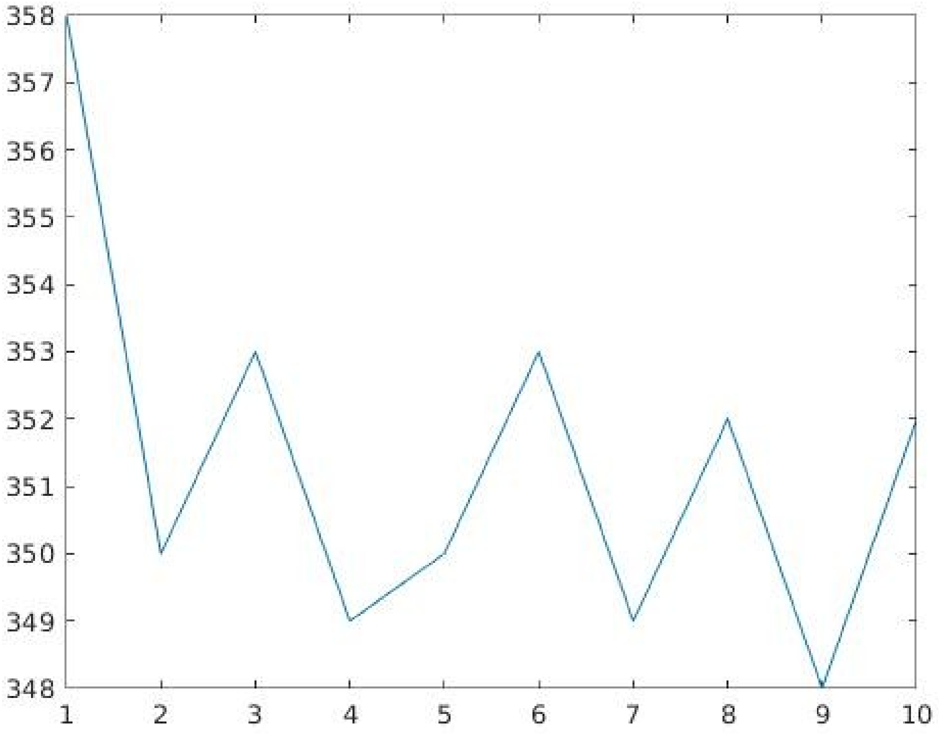
When changing time to a much longer duration (60seconds), there is still no packet losses. Perhaps the UDP protocol bandwidth is preventing any losses from occuring.



```
[  3] local 192.168.10.2 port 5001 connected with 192.168.10.1 port 55008
[ ID] Interval        Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[  3] 0.0000-5.0565 sec  61.7 KBytes   100 Kbits/sec  62.508 ms    0/   43 (0%)
[  4] local 192.168.10.2 port 5001 connected with 192.168.10.1 port 35275
[ ID] Interval        Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[  4] 0.0000-5.0095 sec   612 KBytes  1.00 Mbits/sec   0.008 ms    0/  426 (0%)
[  3] local 192.168.10.2 port 5001 connected with 192.168.10.1 port 50055
[ ID] Interval        Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[  3] 0.0000-15.0057 sec  1.79 MBytes  1.00 Mbits/sec   0.004 ms    0/ 1276 (0%)
[  4] local 192.168.10.2 port 5001 connected with 192.168.10.1 port 36322
[ ID] Interval        Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[  4] 0.0000-60.0005 sec   715 MBytes   100 Mbits/sec   0.003 ms    0/510205 (0%)
[  3] local 192.168.10.2 port 5001 connected with 192.168.10.1 port 51933
[ ID] Interval        Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[  3] 0.0000-60.0005 sec   715 MBytes   100 Mbits/sec   0.002 ms    0/510205 (0%)
```



Y-axis = packet loss %
X-axis = Bandwidth limit in bits (log scale)

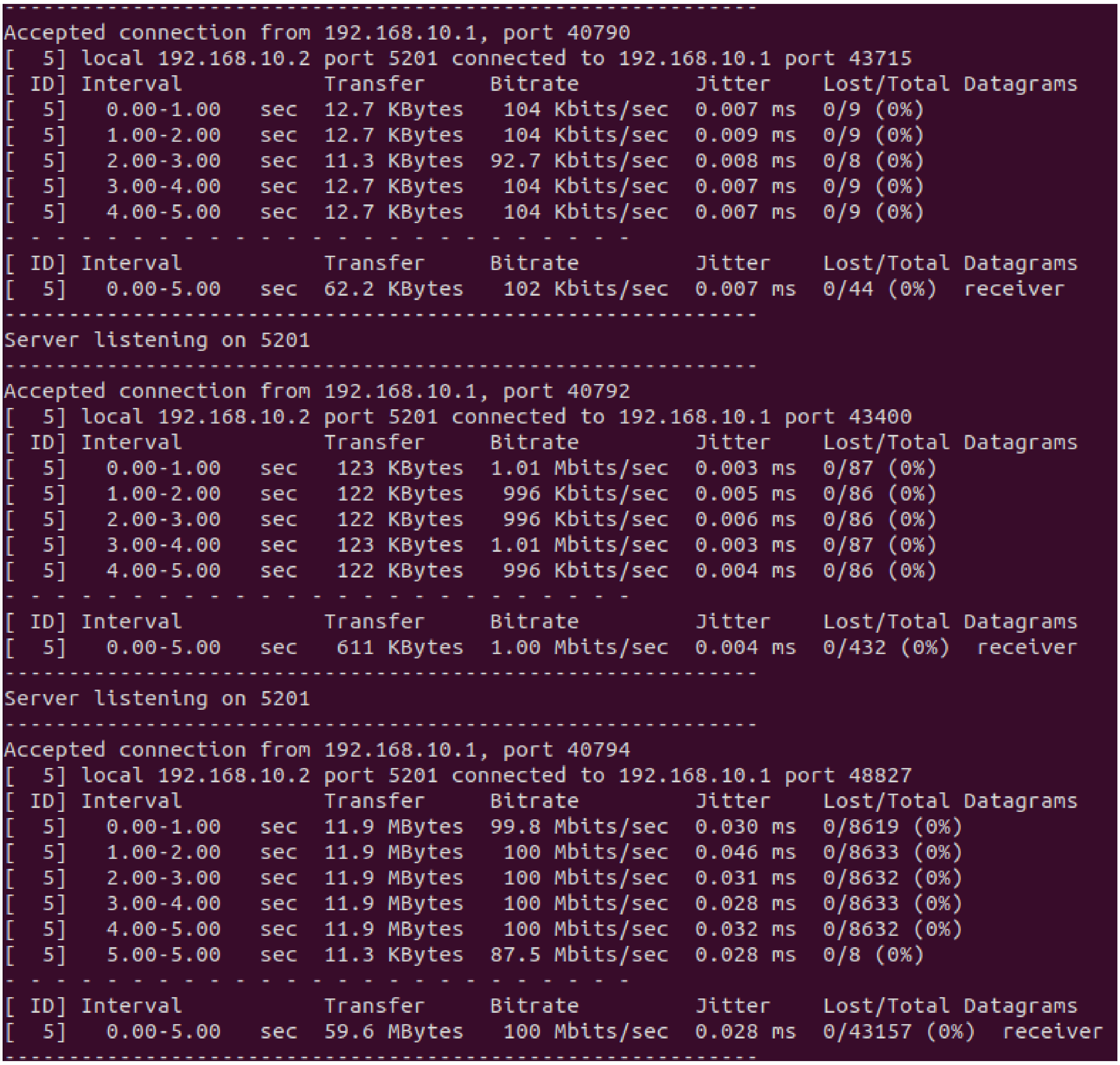


Y-axis = Bandwidth in Mbits/s X-axis = time/s

Set Raspberry Pi as iperf3 server, lab machine as client.
TCP,
10s long experiment, interval 1
Plot above shows bandwidth over each interval in Mbits/s

Running one way via UDP, with varying bandwidths. There is still no datagrams losses, unclear as to how is there any packet loss without exceeding cable bandwidth(bandwidth of machine to raspberry pi exceeds 100Mbits/sec.).

Graph will be the same as iperf packet losses. (0% throughout)



```
-----------------------------------------------------------
Accepted connection from 192.168.10.1, port 40790
[  5] local 192.168.10.2 port 5201 connected to 192.168.10.1 port 43715
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  12.7 KBytes   104 Kbits/sec  0.007 ms  0/9 (0%)
[  5]   1.00-2.00   sec  12.7 KBytes   104 Kbits/sec  0.009 ms  0/9 (0%)
[  5]   2.00-3.00   sec  11.3 KBytes  92.7 Kbits/sec  0.008 ms  0/8 (0%)
[  5]   3.00-4.00   sec  12.7 KBytes   104 Kbits/sec  0.007 ms  0/9 (0%)
[  5]   4.00-5.00   sec  12.7 KBytes   104 Kbits/sec  0.007 ms  0/9 (0%)
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-5.00   sec  62.2 KBytes   102 Kbits/sec  0.007 ms  0/44 (0%)  receiver
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 192.168.10.1, port 40792
[  5] local 192.168.10.2 port 5201 connected to 192.168.10.1 port 43400
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec   123 KBytes  1.01 Mbits/sec  0.003 ms  0/87 (0%)
[  5]   1.00-2.00   sec   122 KBytes   996 Kbits/sec  0.005 ms  0/86 (0%)
[  5]   2.00-3.00   sec   122 KBytes   996 Kbits/sec  0.006 ms  0/86 (0%)
[  5]   3.00-4.00   sec   123 KBytes  1.01 Mbits/sec  0.003 ms  0/87 (0%)
[  5]   4.00-5.00   sec   122 KBytes   996 Kbits/sec  0.004 ms  0/86 (0%)
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-5.00   sec   611 KBytes  1.00 Mbits/sec  0.004 ms  0/432 (0%)  receiver
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 192.168.10.1, port 40794
[  5] local 192.168.10.2 port 5201 connected to 192.168.10.1 port 48827
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  11.9 MBytes  99.8 Mbits/sec  0.030 ms  0/8619 (0%)
[  5]   1.00-2.00   sec  11.9 MBytes   100 Mbits/sec  0.046 ms  0/8633 (0%)
[  5]   2.00-3.00   sec  11.9 MBytes   100 Mbits/sec  0.031 ms  0/8632 (0%)
[  5]   3.00-4.00   sec  11.9 MBytes   100 Mbits/sec  0.028 ms  0/8633 (0%)
[  5]   4.00-5.00   sec  11.9 MBytes   100 Mbits/sec  0.032 ms  0/8632 (0%)
[  5]   5.00-5.00   sec  11.3 KBytes  87.5 Mbits/sec  0.028 ms  0/8 (0%)
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-5.00   sec  59.6 MBytes   100 Mbits/sec  0.028 ms  0/43157 (0%)  receiver
-----------------------------------------------------------
```

In terms of raw data, there isn’t really any observed differences between iperf and iperf3. The amount of data transferred in each test is the same, and no data loss occurred.

Simply put, 1Gb/s is 10 times greater than 100mb/s. There is far enough buffer that no packet loss should occur.