Ping

1. Ping from the lab machine to the Raspberry Pi, 10 times, interval 0.2 seconds. Report the results.

10 packets transmitted, 10 received, 0% packet loss, time 1834ms
rtt min/avg/max/mdev = 0.044/0.051/0.083/0.010 ms

2. Ping from the Raspberry Pi to the lab machine, 10 times, interval 0.2 seconds. Report the results.

10 packets transmitted, 10 received, 0% packet loss, time 1808ms
rtt min/avg/max/mdev = 0.026/0.048/0.076/0.018 ms

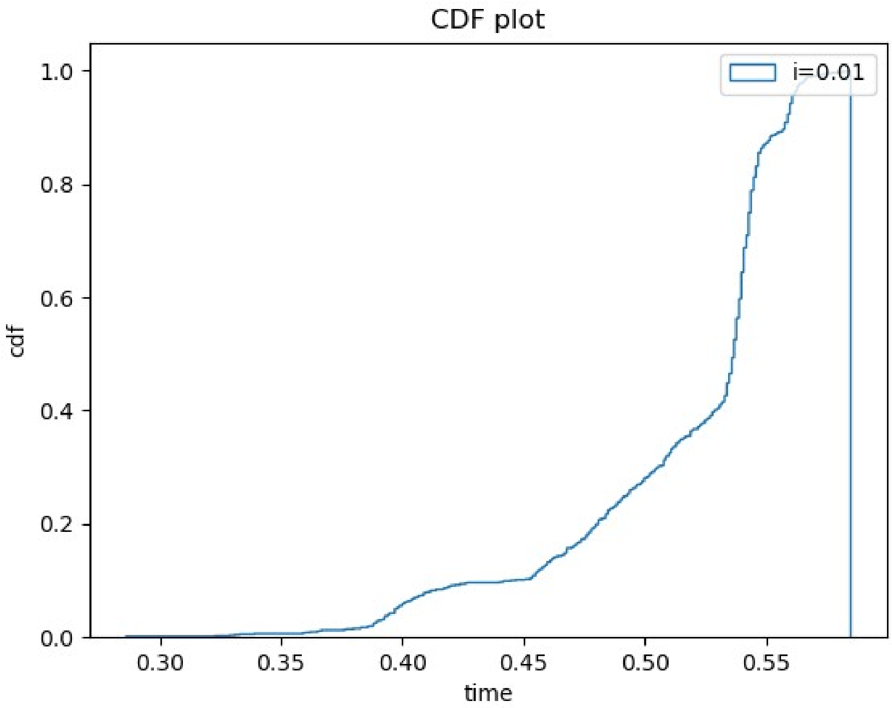
3. Ping from the Raspberry Pi to the lab machine, 100 times, interval 0.001 seconds (use sudo). Report the results, and discuss the differences between minimum, mean and maximum results.

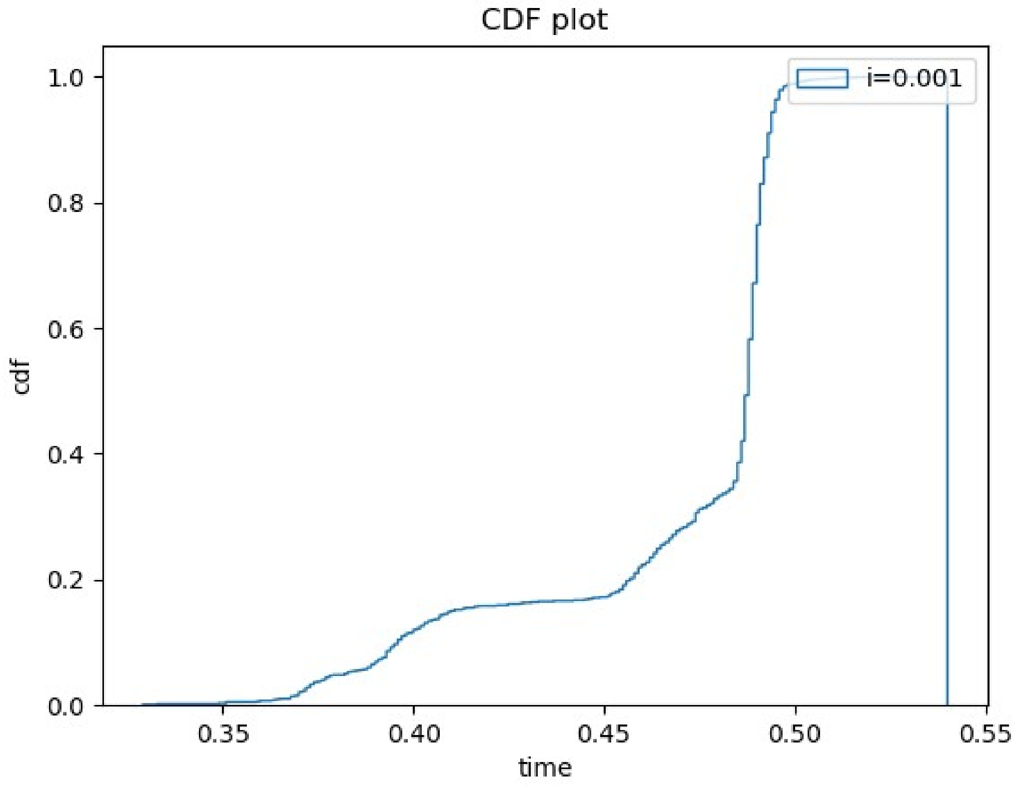
100 packets transmitted, 100 received, 0% packet loss, time 99ms
rtt min/avg/max/mdev = 0.017/0.018/0.072/0.005 ms
Minimum and mean times are very close together with the max being much larger. This suggests there are a lot more near the minimum value rather than the maximum.

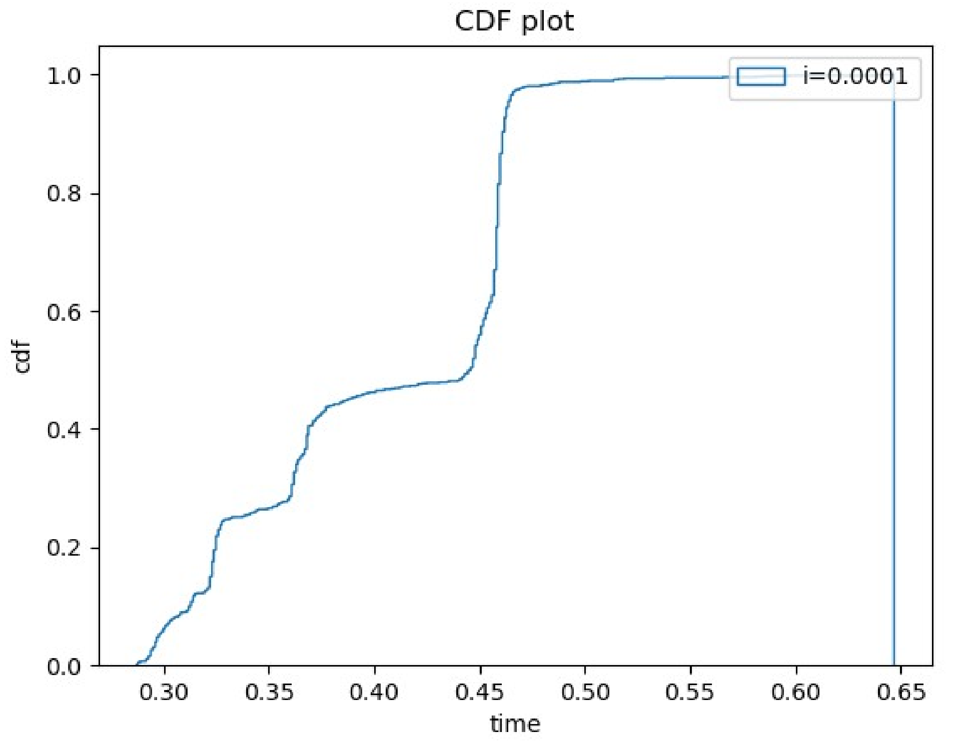
4.Ping from the Raspberry Pi to the lab machine, 10000 times using flooding (use sudo). Report the results.

10000 packets transmitted, 10000 received, 0% packet loss, time 660ms
rtt min/avg/max/mdev = 0.016/0.017/0.470/0.010 ms, ipg/ewma 0.065/0.017 ms

5. Ping from the Raspberry Pi to the lab machine. Run measurements with 3 different intervals (0.01, 0.001, 0.0001) and at least 1000 measurements, and draw a cdf of your measurements results (one graph per interval).







6. Can you speculate why different intervals lead to different round trip results? What do you estimate is the most accurate measured parameter (e.g., min, max, mean) that can be used to estimate propagation time between the two machines?

As more of the CPU is dedicated to it when there is a smaller interval making it faster. This will not happen for the first few runs as the CPU is still running other tasks creating the outliers that have much larger round trips. The most accurate for large volumes of traffic will be the mean as the CPU will be more dedicated to it. However, for smaller volume of traffic the propagation time would be much closer to the maximum value.

Iperf

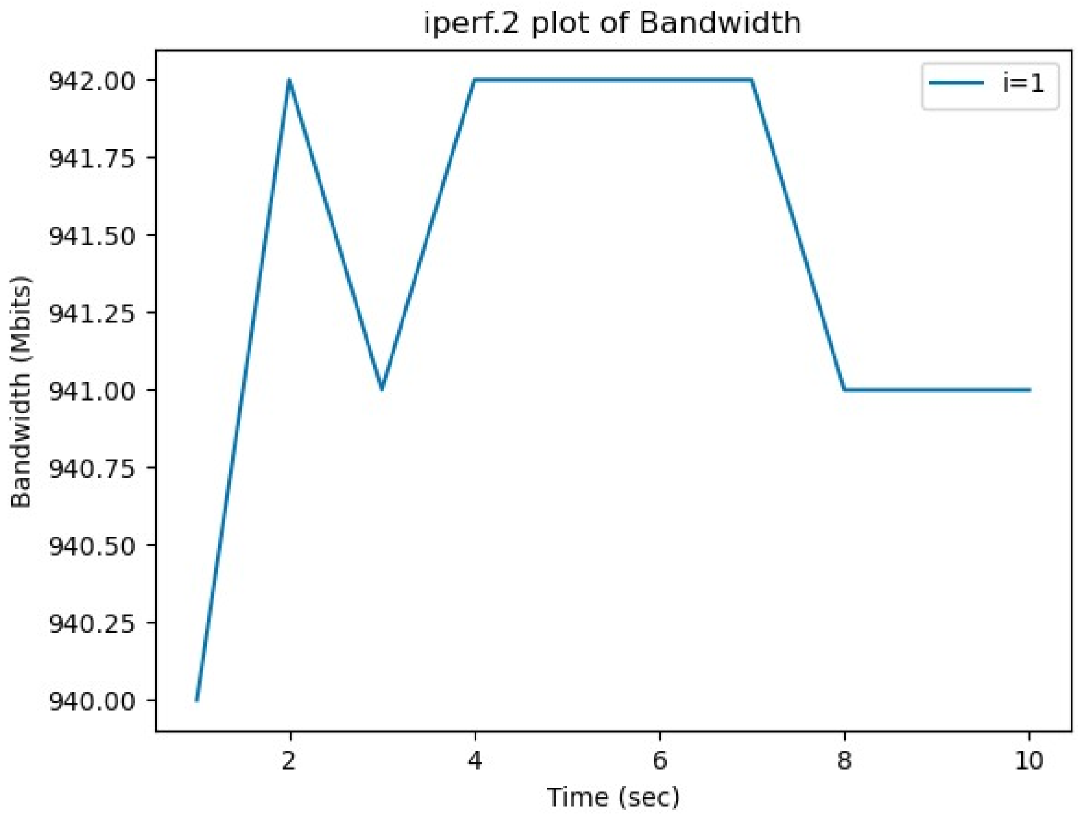
1. Set the lab machine as iperf server, and the Raspberry Pi as the client, and use iperf to measure the effective bandwidth between the two, using TCP and a 10 seconds long experiment. Report the results.

Lab Computer side
[1] local 192.168.10.1 port 5001 connected with 192.168.10.2 port 50048
[ID] Interval   Transfer   Bandwidth
[1] 0.0000-10.0290 sec 1.09 GBytes   936 Mbits/sec

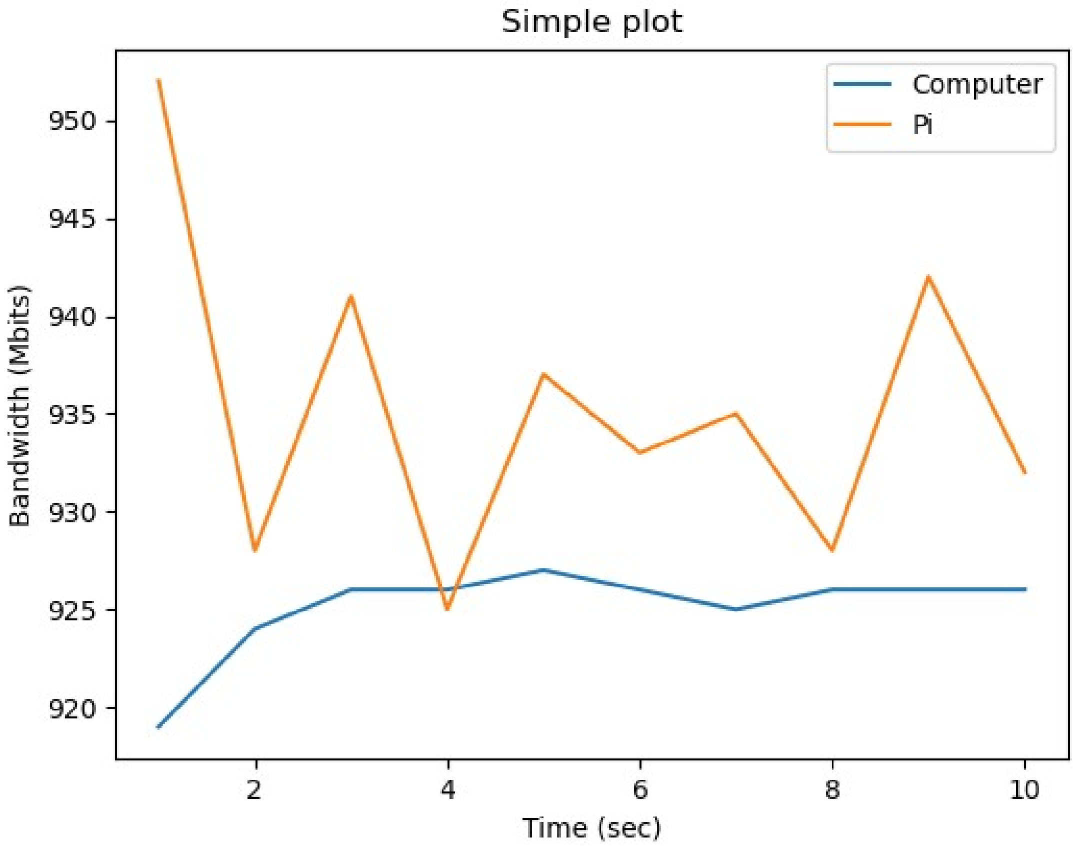
Raspberry Pi side
[3] local 192.168.10.2 port 50048 connected with 192.168.10.1 port 5001
[ID] Interval   Transfer   Bandwidth
[3] 0.0000-10.0085 sec 1.09 GBytes   938 Mbits/sec

The effective bandwidth is 936Mbits/sec

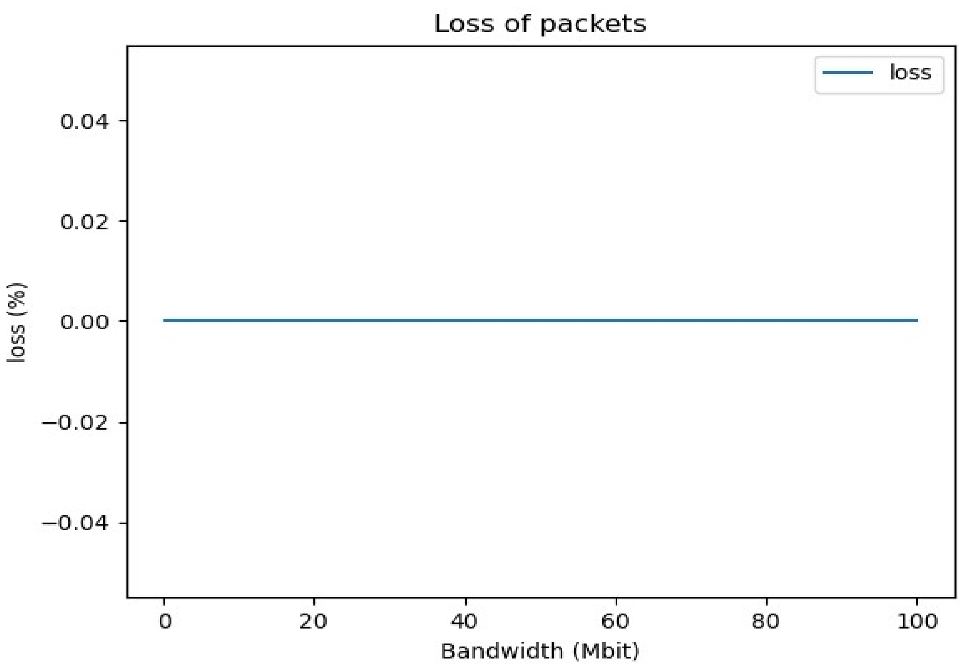
2. Set the Raspberry Pi as iperf server, and the lab machine as the client, and use iperf to measure the effective bandwidth between the two, using TCP, a 10 seconds long experiment, and 1 second interval. Plot the measured bandwidth.



3. Set the Raspberry Pi as iperf server, and the lab machine as the client, and use bi-directional iperf to measure the effective bandwidth between the two, using TCP, a 10 seconds long experiment, and 1 second interval.
Plot the measured bandwidth in each direction.



4. Run one way iperf using UDP, from the lab machine to the Raspberry Pi, 5 sec long, with varying bandwidth (100Kb/s, 1Mb/s, 100Mb/s).
a. Plot the percentage of packets dropped for each bandwidth.



b. Report the iperf results of each configuration.

100Kb/s:
[ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[1] 0.0000-5.1745 sec 66.0 KBytes 105 Kbits/sec 0.007 ms 0/46 (0%)

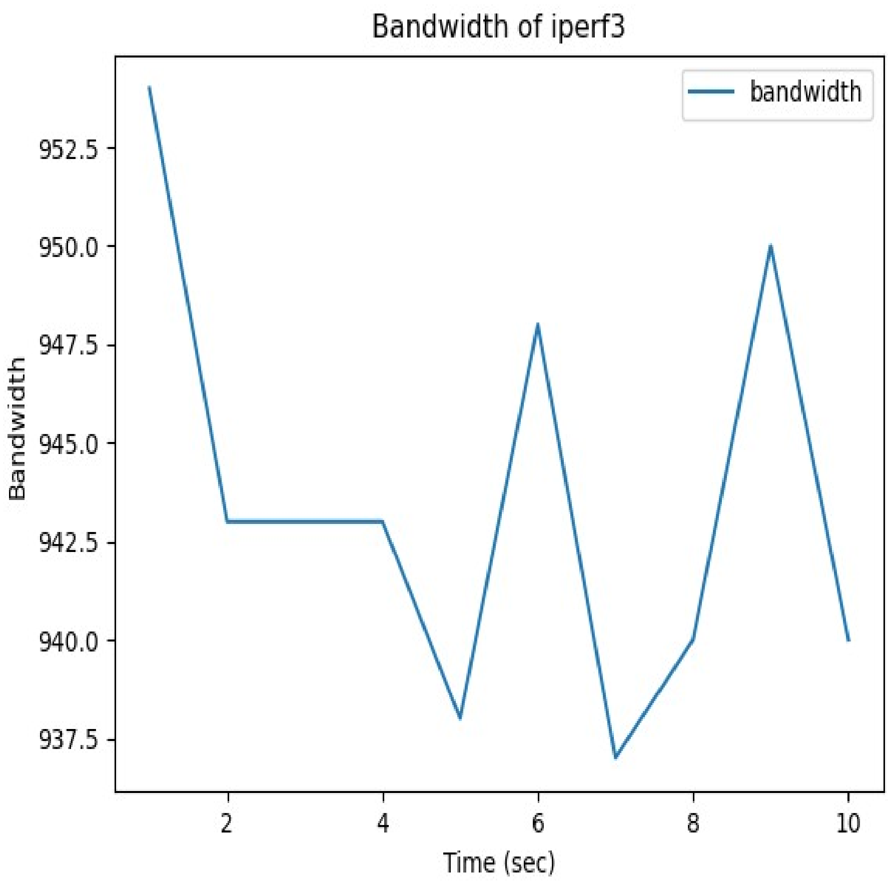
1Mb/s:
[ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[1] 0.0000-5.0216 sec 616 KBytes 1.00 Mbits/sec 0.004 ms 0/429 (0%)

100Mb/s:
[ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[1] 0.0000-4.9997 sec 59.6 MBytes 100 Mbits/sec 0.007 ms 0/42520 (0%)

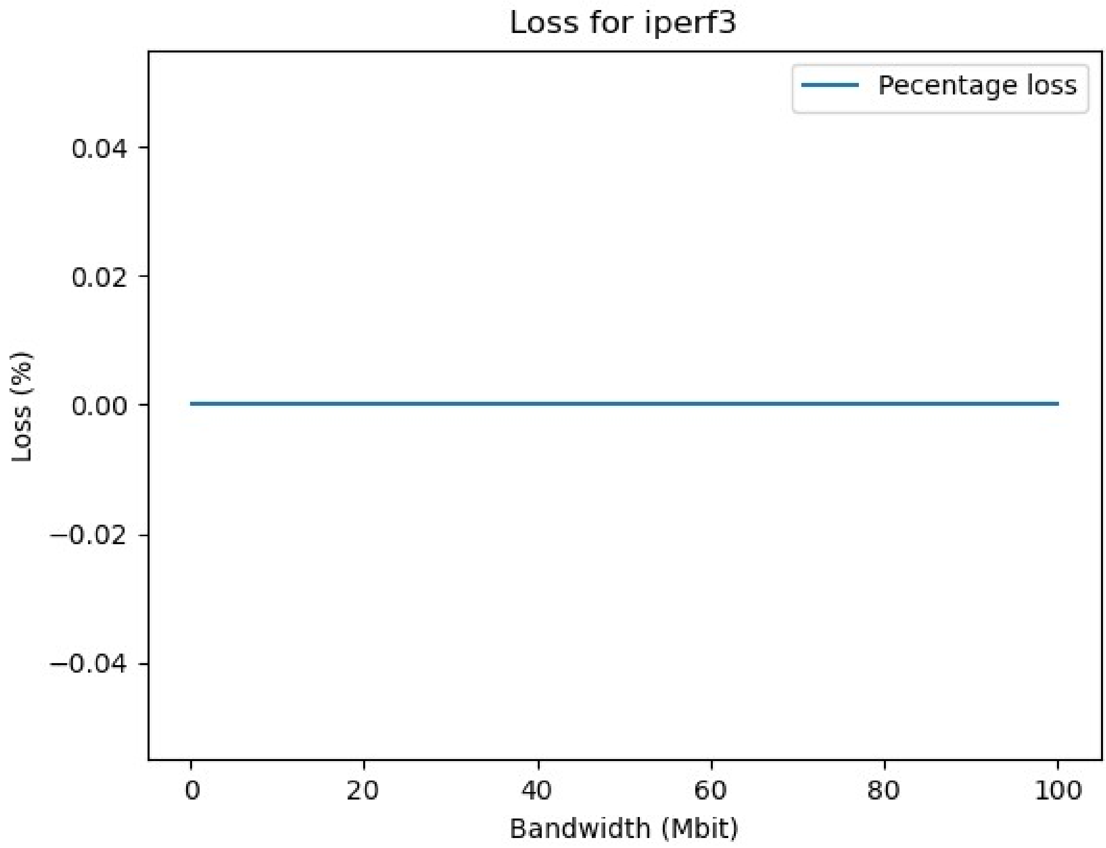
No loss through any bandwidth.

Iperf3

1. Set the Raspberry Pi as iperf3 server, and the lab machine as the client, and use iperf3 to measure the effective bandwidth between the two, using TCP, a 10 seconds long experiment, and 1 second interval.
Plot the measured bandwidth



2. Run one way iperf3 using UDP, from the lab machine to the Raspberry Pi, 5 sec long, with varying bandwidth (100Kb/s, 1Mb/s, 100Mb/s).
a. Plot the percentage of packets dropped for each bandwidth.



b. Report the iperf3 results of each configuration.
Bandwidth ~ Bitrate

100Kb/s:

[ID]	Interval		Transfer	Bitrate	Jitter	Lost/Total Datagrams	
[5]	0.00-5.00	sec	62.2 KBytes	102 Kbits/sec	0.000 ms	0/44 (0%)	sender
[5]	0.00-5.04	sec	62.2 KBytes	101 Kbits/sec	0.010 ms	0/44 (0%)	receiver

1Mb/s:

[ID]	Interval		Transfer	Bitrate	Jitter	Lost/Total Datagrams	
[5]	0.00-5.00	sec	611 KBytes	1.00 Mbits/sec	0.000 ms	0/432 (0%)	sender
[5]	0.00-5.04	sec	611 KBytes	993 Kbits/sec	0.005 ms	0/432 (0%)	receiver

100Mb/s:

[ID]	Interval		Transfer	Bitrate	Jitter	Lost/Total Datagrams	
[5]	0.00-5.00	sec	59.6 MBytes	100 Mbits/sec	0.000 ms	0/43158 (0%)	sender
[5]	0.00-5.04	sec	59.6 MBytes	99.2 Mbits/sec	0.018 ms	0/43158 (0%)	receiver

No loss through transfer.

3. Discuss any observed differences between iperf and iperf3 results.
Iperf3 measured both sender and receiver over the interval and measured to exact 5 seconds. Iperf only measured one way with less accurate timing for the test.