Introduction

A simple network topology with three hosts is created using Mininet. Two of these hosts act as normal host nodes while the third one is configured as a router which connects the other two hosts. The ultimate goal is to measure the Completion Time and Retransmission Time in a TCP file transfer from a host to the other host over this network. This is done in two phases-

i) TLP Enabled
ii) TLP Disabled

Completion Time is the time elapsed between first and last observed packet of the TCP connection. And by Retransmission Time, we refer to the time elapsed between dropping of first packet and beginning of the first packet retransmission.

Overview

The execution starts with config_measurement.py which is developed to measure the Completion Time and Retransmission Time in the network with different link configurations in terms of bandwidth and latency, and also the length of TCP segments transferred. Measurements are taken by enabling and disabling Tail Loss Probe one by one for each configuration. So in all there are 72 different cases which are analyzed in this framework. In the end, results are studied in the form of Box and Whisker Plots which give clear comparison of the behavior between TLP Enabled and TLP Disabled scenarios.

Configuration Description

We have collected various samples on the basis of the following characteristics:

Lengths of the TCP segments used are:

i) Long (256)
ii) Medium (128)
iii) Short (64)

The **configurations of the links** in the network topology are as follows:

Configuration	Link 1		Link 2	
	Bandwidth	Latency	Bandwidth	Latency
Fast	1 Mbit/s	5 ms	1 Mbit/s	5 ms
Moderate	500 Kbit/s	10 ms	500 Kbit/s	10 ms
Slow	100 Kbit/s	20 ms	100 Kbit/s	20 ms

The emulation was done for the following number of packets dropped:

i) One
ii) Two
iii) Four
iv) Eight

Results

Few cases among 18 cases are shown here with the readings which are as follows:

Case i): Fast and Short

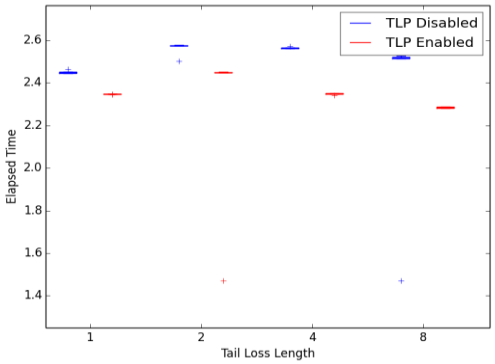

Fig 1.Completion time

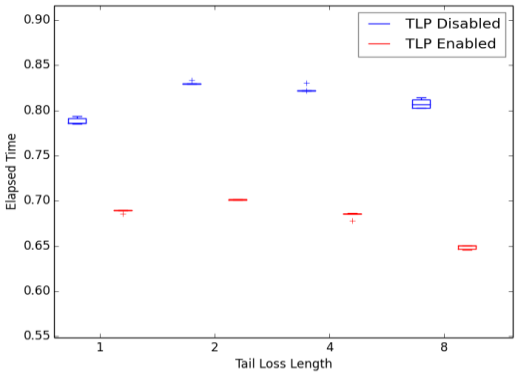

Fig.2 Retransmission time

No. of packets lost	Fast and Short					
	Completion Time			Retransmission Time		
	TLP Enabled	TLP Disabled	%age Improvement	TLP Enabled	TLP Disabled	%age Improvement
1	2.346615696	2.450330632	4.23	0.68883224	0.78839159	12.63
2	2.253173304	2.560421705	12.00	0.70128846	0.830515862	15.56
4	2.347658443	2.565178871	8.48	0.68441143	0.823593569	16.90
8	2.284595346	4.588064814	50.21	0.64870358	0.768411827	15.58

Case ii) Moderate and Short

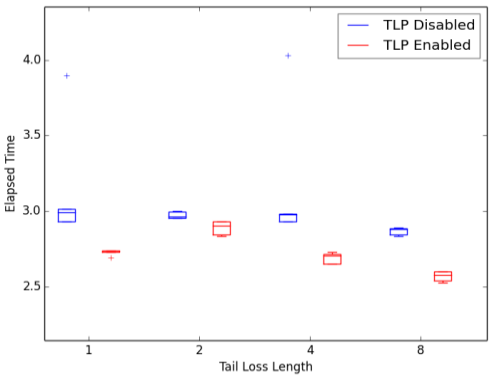

Fig 3 Completion time

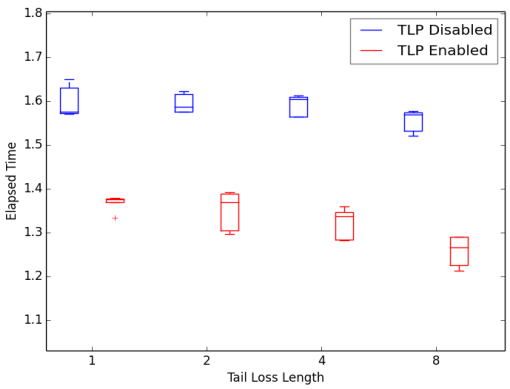

Fig.4 Retransmission time

No. of packets lost	Moderate and Short					
	Completion Time			Retransmission Time		
	TLP Enabled	TLP Disabled	%age Improvement	TLP Enabled	TLP Disabled	%age Improvement
1	2.729123211	3.154491854	13.48	1.36773953	1.600731564	14.56
2	2.890166044	2.973709154	2.81	1.35082488	1.595660543	15.34
4	2.692448425	3.172083998	15.12	1.32237182	1.591579247	16.91
8	2.571008205	2.868571806	10.37	1.2572638	1.555249834	19.16

Case iii) Slow and Medium

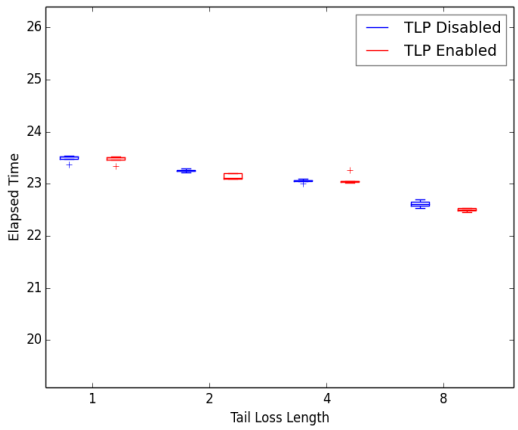

Fig 5 Completion time

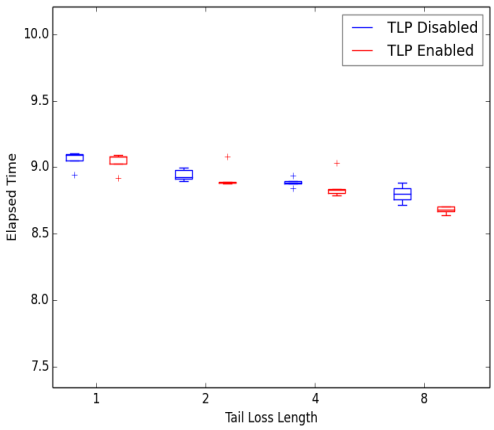

Fig.6 Retransmission time

No. of packets lost	Slow and Medium					
	Completion Time			Retransmission Time		
	TLP Enabled	TLP Disabled	%age Improvement	TLP Enabled	TLP Disabled	%age Improvement
1	23.4679976	23.48760281	0.08	9.03940067	9.057588816	0.20
2	23.14331946	23.25451102	0.48	8.9214169	8.941370985	0.22
4	23.08449221	23.05964179	-0.11	8.85805345	8.875452841	0.20
8	22.50307684	22.61740303	0.51	8.6791656	8.79709053	1.34

Case iv) Fast and Long

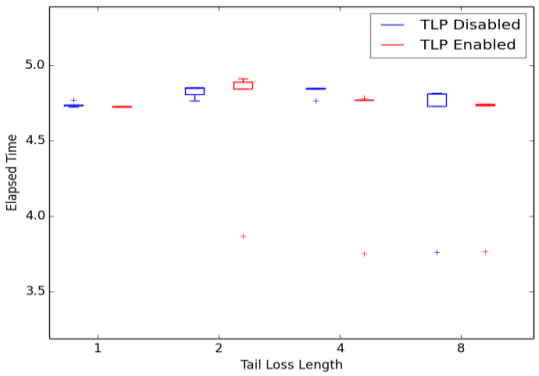

Fig 7 Completion time

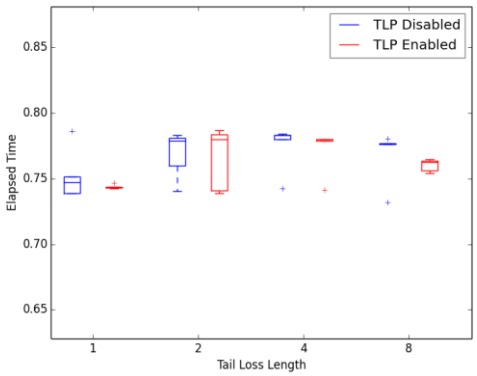

Fig.8 Retransmission time

No. of packets lost	Fast and Long					
	Completion Time			Retransmission Time		
	TLP Enabled	TLP Disabled	%age Improvement	TLP Enabled	TLP Disabled	%age Improvement
1	4.728798008	4.741371012	0.27	0.74389477	0.752494812	1.14
2	4.68159399	4.825225671	2.98	0.76593156	0.767366648	0.19
4	4.570185852	4.834159613	5.46	0.77193475	0.774491596	0.33
8	4.546319818	4.588064814	0.91	0.76028874	0.768411827	1.06

Explanation

With TLP enabled, it is found that it takes less completion and retransmission time compared to TLP disabled. This is because-

Firstly, the sender waits for the Retransmission Timeout(RTO) after the tail loss occurs in the case of TLP Disabled.

And secondly, Tail Loss Probe (TLP) enables sending of probe segments to trigger duplicate ACKs with the intent of invoking fast recovery more quickly than an RTO at the end of a transaction.

This is true irrespective of the number of packets dropped. In the above figures (Fig 1 and Fig 2), it is also observed that the performance of TLP increases with the increase in the number of packets dropped. The increase in difference between the elapsed times for TLP enable and disable cases can be attributed to that.

Conclusion

We can conclude the following:

1. TLP Enabled improves the performance in most of the tested cases.
2. More the bandwidth and lesser the size of the segment can give us better performance enhancement in case of TLP Enabled.
3. More the packets dropped better is the performance of TLP.

Feedback

We invested 5 hours a week to reach the milestones. The project required active research in the area of TLP and python libraries. The learning curve was pretty steep throughout the project, which was the actual bonus apart from the tangible grades. We would say that the work load was just right.

There are many aspects of the project which were quite interesting. The emulation of TLP and checking out the TCP dumps in Wireshark were exciting. The insights on the packet loss and retransmission were engaging and the overall knowledge obtained is worth the time spent.

The frequent hiccups in the Mininet which involved restarting the VM as the only remedy were a little frustrating. This slowed our progress and the ignorance of the existing bugs was something which we did not see coming.

Now, as we know the libraries which are involved in the project, we can delve more into code efficiency. Additionally we can add those minute details like sleep or avoiding the promiscuous mode which can make a difference in tackling the anomalies of the Mininet.

The message to the people who are yet to do the assignment would be to be well versed in Python and its networking libraries. The on-the-job learning is best part of this project so one should not miss the opportunity in trying this out.