LAB 11

LAB REPORT

1. For ns-tcp.tcl:

a) What is the size of the TCP packet, what is the size of the ack?

A) Size of TCP packet is 1040 bytes.

Size of ack is 40 bytes.

b) What application feeds TCP?

A) FTP application feeds TCP.

c) What is the queue-size at the intermediate node?

A) Queue-size at the intermediate node is 10.

d) What is the data rate and propogation delay of the two links?

A) For link between n0 and n1:
    Data Rate : 1 Mb
    Propagation delay : 50 ms

For link between n1 and n2:
    Data Rate : 100 Kb
    Propagation delay : 5 ms

e) How many tcp data packets and how many ack packets were dropped?

A) Number of tcp packets dropped is 17.
    Command used : grep "^d" tcp.tr | grep "tcp"| wc -l

Number of ack packets dropped is 0.
    Command used : grep "^d" tcp.tr | grep "ack"|wc -l

2. For ns-simple.tcl

a) How many flows and of what type (distinguish based on transport protocol) have been set-up as part of the simulation?

A) 2 flows... types are tcp and udp

b) What application feeds the two flows?

A) TCP flow is fed by FTP

UDP flow is fed by CBR

c) By looking at the 'tcl' file, can you figure out at what rate the two flows are injecting data (source data-rate) into the network? Specify the rate also.

A) UDP-CBR flow : 0.1 MBps (1000B/0.01s)

Using the tcl file, we can't figure out the data-rate of the TCP-FTP flow

d) When do the flows end?

A) TCP-FTP flow ends at 4s

UDP-CBR flow ends at 4.5s

e) Determine the throughput (as observed at the receiver) of the twoflows? Throughput: The rate at which bits (unique not duplicate) are being received at the destination? Eg: If 10 packets of size 100 bytes were received in a duration of 100 seconds at receiver, we say the throughput is 10 * 100 /100 = 10Bps or 80bps or 0.08kbps.

A)CMD for UDP-CBR flow:

Get the number of packets received: grep "^r" simple.tr | grep "3 cbr" | wc -l = 433

Each packet is 1000B -> data received is 433000B

Determine start and end time of the flow (@receiver): grep "^r" simple.tr | grep "3 cbr" > temp; then look at the first and last line

Throughput is: 433000B/(4.538706-0.138706) s = 98.41 Kbps

CMD for TCP-FTP flow:

A common mistake people make is to calculate this as follows:

grep "^r" simple.tr | grep "3 tcp" |wc -l to get the number of packets = 283

The first packet is 40 bytes, rest are 1040. So total data is 293320B

Ans: 293320B/(4.051671-1.030348)s = 97.083 Kbps

However there could be duplicate packets delivered which should not be considered, since its double counting. So,the correct answer is

grep "^r" simple.tr | grep "3 tcp" > tmp

Look at the tmp file to determine the last succesfully received sequence number which is 271. Each packet is 1040B. So bytes recived is 271*1040 = 281840B

Ans: 281840B/(4.051671-1.030348)s = 93.283 Kbps

LAB REPORT

Plot the following graphs. Ensure the axis are properly labeled, with proper legend and correct units.

1. Offered Load vs percentage packet loss

Graph1: Offered Load(OL)(x) vs. Percentage packet loss(PPL) (y):

The PPL is 0% for OL = 40 and 80 Kbps. For OL = 120 Kbps, there is some packet loss (158/1000) and for OL =160Kbps, there is more packet loss (367/1000).

This is because our bottleneck bandwidth is 100Kbps. So as long as OL is less than 100 Kbps, no packets will be dropped and since queue size is finite(10), in OL=120 and 160 Kbps, packets get dropped with more packets being dropped in the later case.

For calculating throughput, use the below

No-pkts = grep "^r" trace-file | grep "2 cbr" | wc -l (or you could subtract the above calculated drops from 1001)

Throughput = No-pkt * pkt-size / (end-time - start-time)
(Use grep "^r" trace-file | grep "2 cbr" > temp. You can get the end-time/start-time from temp by reading the first and last line)

2. Offered load vs throughput

Graph2:Offered Load(OL)(x) vs. Throughput (TP)(y):

The TP increases and is equal to OL as long as OL is less than the bottleneck bandwidth(100Kbps in this case)

When OL > 100 Kbps, the TP increases to reach and is saturated at 100Kbps(the bottleneck bandwidth). This is because by definition, the maximum permissible bandwidth is 100Kbps and, therefore, it cannot cross that value and stays at that value if OL > 100Kbps

In the report, comment on what you observe in each graph and the reasons for the same.

Exercise 11.3: Additional Work(To be evaluated Separately)

This exercise is not compulsory but time permitting you can explore it. This does involve bash scripting or using C code.

This is a continuation of the previous exercise. If the source data rate (rate atwhich source injects data into the network) varies, how does the delay vary as a function of the source data rate. That is plot the Offered Load vs Average end-to-end delay.

Average end-to-end delay for a run: If t1 is the time the packet was generated at the source and t2 was the time the packet was received at the destination. Delay of a packet is t2-t1. Calculate average of these values for packets that reached the destination. Express it in ms. Then plot the Offered Load vs Average end-to-end delay for the different runs.

Graph3:Offered Load(OL)(x) vs. Average End-to-End Delay(AED)(y):

AED increases as OL increases because the traffic on the network increases as OL increases. But AED increases substantially, once the OL increases the bottleneck bandwidth(100Kbps), because then packets start getting queued and the delay increases.