Shenliang Wang

Project2

(a) Based on the execution results of the Awk commands, what is the total traffic of each flows? What is the overall packet loss rate of the two TCP flows? What is the ratio of throughput between the two TCP flows?

**Answer**:

The traffic of TCP1: 95720

The traffic of TCP2: 95720

The total traffic of both: 191440

Dropped Packets: 0

The overall packet loss rate is: 0

The ratio of TCP1: 0.5

The ratio of TCP2: 0.5

The ratio of throughput between the two TCP flows is: 50-50

(b) Revise the sample code "task1\_sample.tcl" so that the window size of the second TCP flow(tcp2) changes from 1 to 25, and save it a new file “task1\_b.tcl”. Use the Awk commands to compute the total traffic of each flow. How does the packet loss rate change? Which flow uses

more bandwidth in the bottleneck from node 3 to node 4? What is the reason?

**Answer**:

The traffic of TCP1: 82200

The traffic of TCP2: 433720

The total traffic of both: 515920

Dropped Packets: 14560

The packet loss rate increase from 0 to 0.0282

The ratio of TCP1: 0.1593

The ratio of TCP2: 0.8407

TCP2 use more bandwidth because the buffer size from node 3 to node 4 is 4 packets, when TCP2 windows size dramatically increase node 3 can’t handle that much traffic flow many packets have dropped causing packet loss.

(c) Revise the sample code "task1\_sample.tcl" so that the window size of both the TCP flows are set to 5, and save it as a new file “task1\_c.tcl”. Use the Awk commands to compute the total traffic of each flows. How does the packet loss rate change? Do both flows receive a "fair share" of the available bandwidth of the bottleneck link?

**Answer**:

The traffic of TCP1: 383800

The traffic of TCP2: 217400

The total traffic of both: 601200

Dropped Packets: 11440

The packet loss rate increase from 0 to 0.0190

The ratio of TCP1: 0.6384

The ratio of TCP2: 0.3616

comparing to case (b) but increase comparing to case (a). TCP1 get more bandwidth than TCP2.

(d) Revise the sample code "task1\_sample.tcl" so that the window size of both the TCP flows are set to 5 and the queue buffer size for the link from node 3 to node 4 is set to 40 packets. Save the new file as “task1\_d.tcl”. Use the Awk commands to compute the total traffic of each flows. How does the packet loss rate change? Do both flows receive a "fair share" of the available bandwidth of the bottleneck link?

**Answer**:

The traffic of TCP1: 305800

The traffic of TCP2: 303720

The total traffic of both: 609520

Dropped Packets: 0

The packet loss rate decrease to 0

(e) Revise the sample code “task1\_sample.tcl” and add node 5 to the network configuration. It is connected to node 3. The link between node 5 and node 3 has a speed of 2 Mbps and a propagation delay of 10 ms. The new network topology is shown in Figure 2. One more

TCP/FTP flow with a window size of 5 is generated from node 5 to node 4. Save the new file as “task1\_e.tcl”. Run the simulation. Revise the sample Awk commands to compute the total bytes of TCP for each flow. What are the Awk commands? Do all the TCP flows receive a fair "share" of the bandwidth? Why or why not?

**Answer**:

NO, because:

When queuing TCP traffic into a queue, competing TCP flows can cause a great amount of data packets queued at the device, resulting in high buffer requirement and large packet latency. This is caused by the mismatch between the increasing TCP window and the bottleneck bandwidth. Moreover, the TCP flows may not fairly share the class bandwidth, especially when their round-trip times (RTTs) are different. Thus many vendors apply specific algorithms for regulating TCP traffic.

Unlike TCP, UDP (User Datagram Protocol) only provides port number multiplexing and is commonly used to carry real-time applications such as Voice over IP (VoIP). Such applications often have fixed bandwidth and the video/voice quality heavily depends on the loss rate, delay, and delay jitter. The packet scheduler must precisely allocate enough bandwidth for real-time UDP traffic to minimize packet losses, delay, and delay jitter at the controlling device.