Henry Hoekel

14207985

HW1

3.a) 1000kb file, RTT = 50ms, packet size = 1kb

initial 2x RTT

bandwidth = 1.5mbps

# of packets = 1000kb / 1kb = 1 packet

total = 2RTT (initial handshake) + time to transmit 100kb+ propagation delay

= 2\* 0.05 + 1000\*(2^10 \* 8/(1.5\*10^6))+ 0.05/2

=0.1 + 5.4613333… + 0.025 = 5.5863333… seconds

b) After each packet, have to wait one RTT.

Total = transmission of a packet + one RTT for first 999 \* RTT

=5.58633333… + 999\*0.05 = 55.536333333… seconds

c) # of RTTs for 1000 packets = 1000/20 = 50 RTTs

last packet, we only consider one way 🡪 50-1 = 49

initial handshake = 2 RTT

last transmission is ½ RTT for propagation delay

= 2RTT + 49RTT + ½ RTT = 51.5 RTT

=51.5 RTT / (1000/50) = 2.575 sec

d) 1+2+4+…+2^t (around)= (2^(t+1)) -1

(2^(t+1)) – 1 >=1000

calculating the max t to be 9

t = 9+ 2 (initial RTT) + ½ (propagation delay) = 11.5RTT

=11.5RTT/ (1000/50) = 0.575 sec

10. (Network architecture -1 slide 8)

STDM (synchronous time- division multiplexing) and FDM (frequency-division multiplexing) are intended for continuous connection and communication. FDM is useful for TV because communication is split among “channels,” of frequencies, i.e. creating specific TV channels to watch. This doesn’t work for computer communication because the maximum number of flows must be fixed ahead of time. The large amount of computers on a network and the changing of users and the number or amount of traffic makes this very inconvenient in this network. STDM is useful in a continuous connection, specifically 1-1. This is cost effective when talking on the phone because this connection is very desirable. On the other hand, in a computer network, if the host has no data to send, then these resources are wasted.

16.a) 100mbps Ethernet

packet size = 12000bits = 12 kb

propagation delay = 10uS

throughput = transfer size / transfer time

12000bits / (100mbps\* 1000000 bits/MB) = 1.2\*10^-4s

=0.00012 = 12ms

0.12ms + 0.01ms (propagation delay) = 0.13ms

2(round trip) \* 0.13ms = 0.26ms

b) 4\* 0.13ms = 0.52ms

c) first 200 bits: 200bits / (100mbps\* 1000000 bits/mb) = 2\*10^-6 seconds

= 0.000002 seconds = 0.002ms

total = 0.002ms (first 200) + 0.01ms (first send) + 0.12ms (entire packet) + 0.01ms (for finish)

=0.142ms

18.a) The effective bandwidth is 100mbps. The switches allow data to steadily stream data. This is assuming the switches are adequate.

b) 100MBps network

each bit = 1/ (10^8) = 10ns (propagation delay)

10ns \* 12000 bits = 120us

delay = 4\*120us + 4\* 10 us = 520 us

4 us delay

4\* 4us + 4\*10us = 56 us

total RTT = 56 us + 520 us = 576us

c) (100\*4.7\*10^9bytes) / (12 hours **(60min)** (60seconds)) = 10.9 MBps

=87mbps

26.a) 640x480, 3bytes / pixel, 30 frames/ second

640\*480\*3 bytes/ pixel \* 30 frames/ second

=27648000 bytes / second

=27.648 MB/second

b) 160\*120, 1 byte/pixel, 5 frames/second

160\*120\* 1 byte/pixel\* 5 frames/second = 96000 bytes / second

=96 kb/ sec

c) 650MB and lasts 75 min

650MB/ 75 Min = 8.66 (around)=8.7mb/min\*60sec/min

=0.145MB/sec

d) 8\* 10-inch b/w image, 72 pixels per inch

14.4 kbps

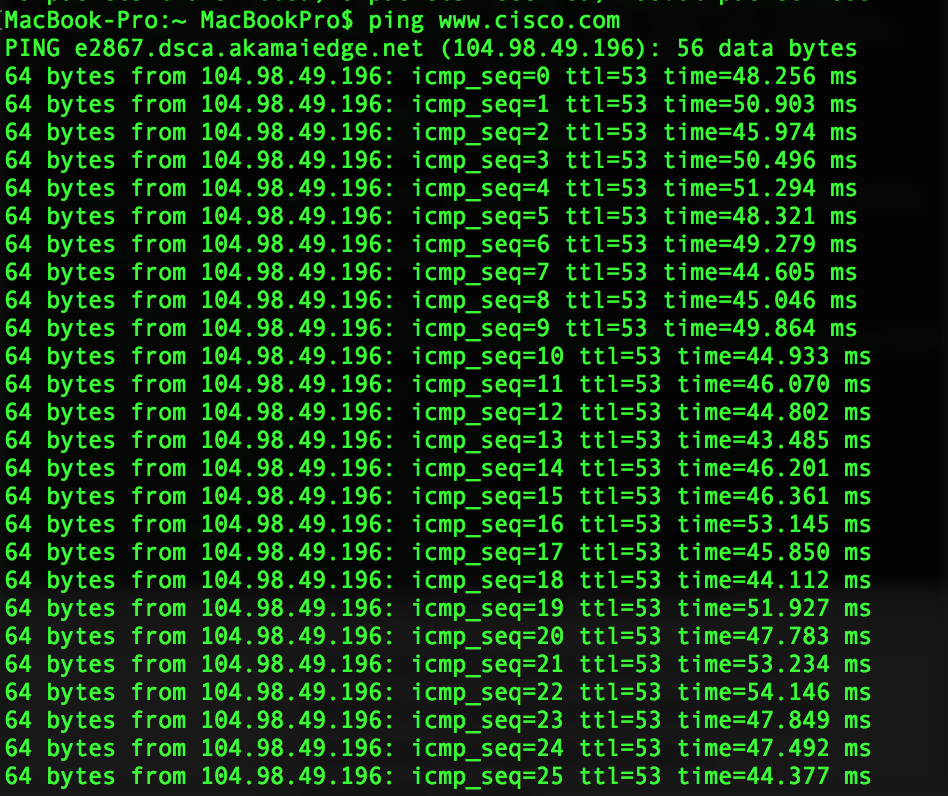
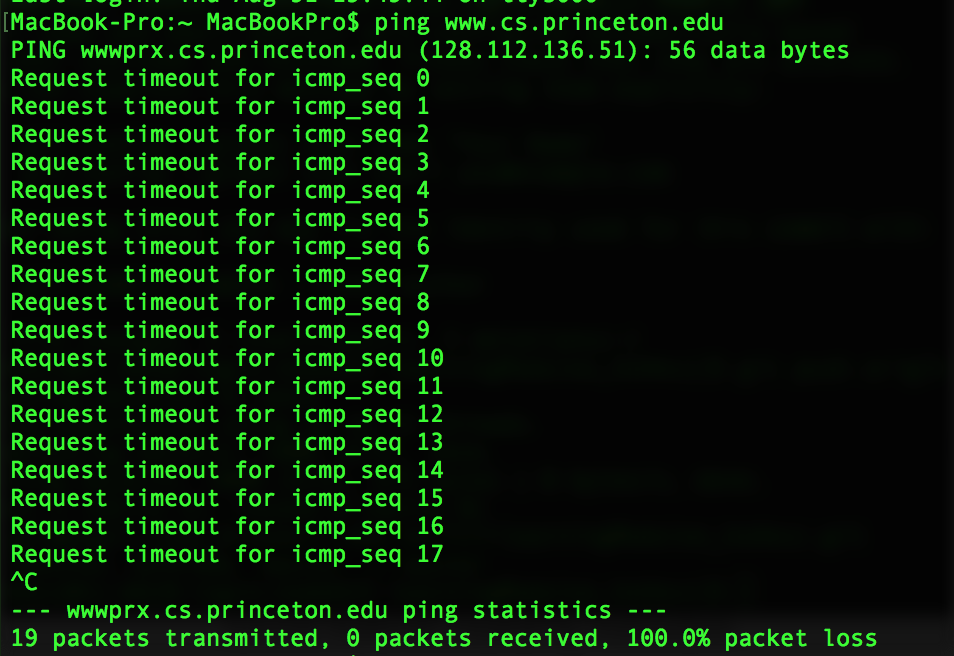
8\*10\*72\*72 = 414720bits / 14.4 kbps\* (1kb/1000bits) = 28.8 seconds

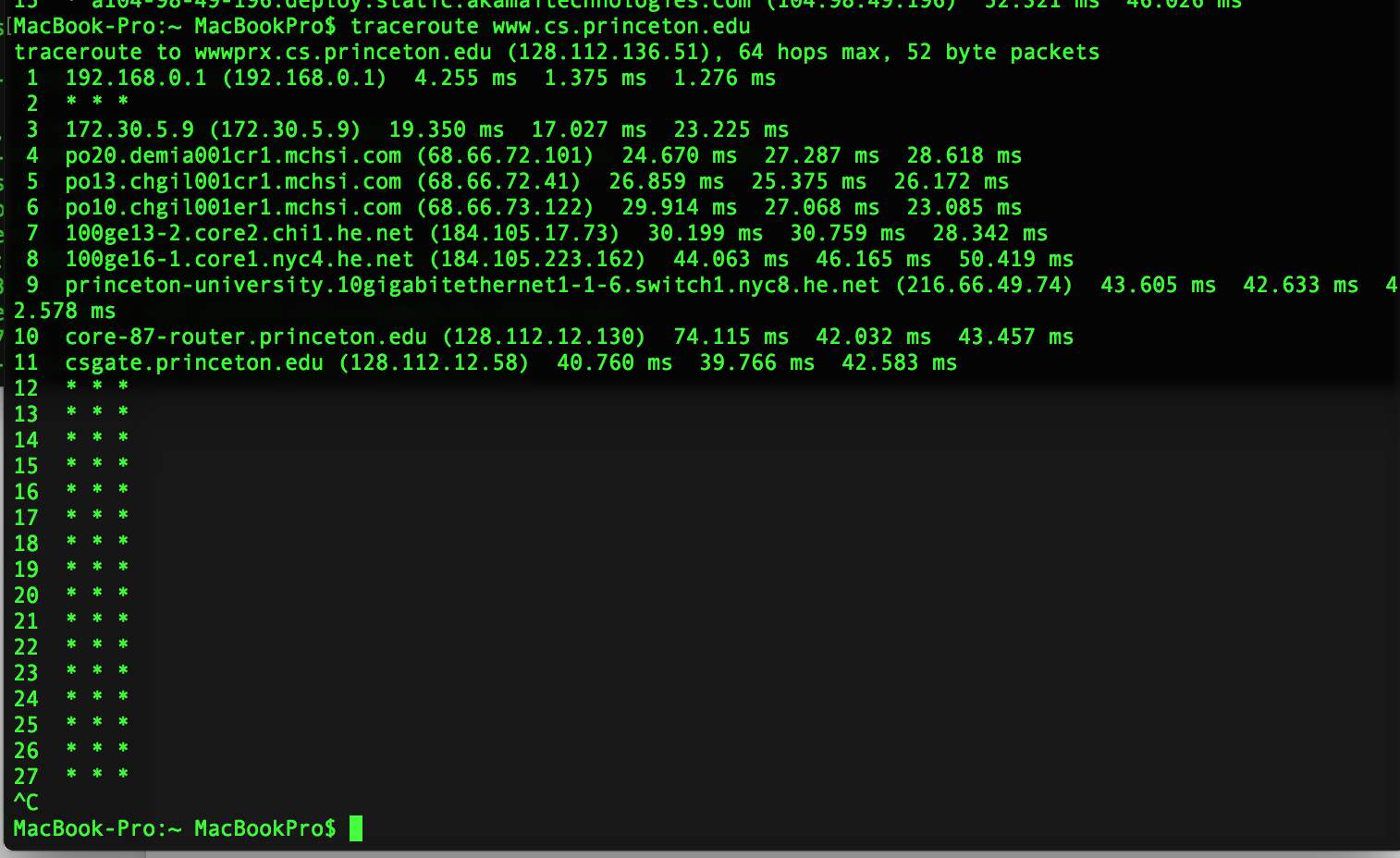
36. The differences could be due to different amounts of traffic for each host. For example, Princeton could have a higher delay during the day due to higher traffic, whereas Cisco retained a fairly consistent ping. Cisco needs to operate through all hours of the day/ night because of its global service. This contrasts with Princeton. Also, when I pinged Princeton, the network seemed to be down, whereas Cisco was pinging consistently around 50ms.

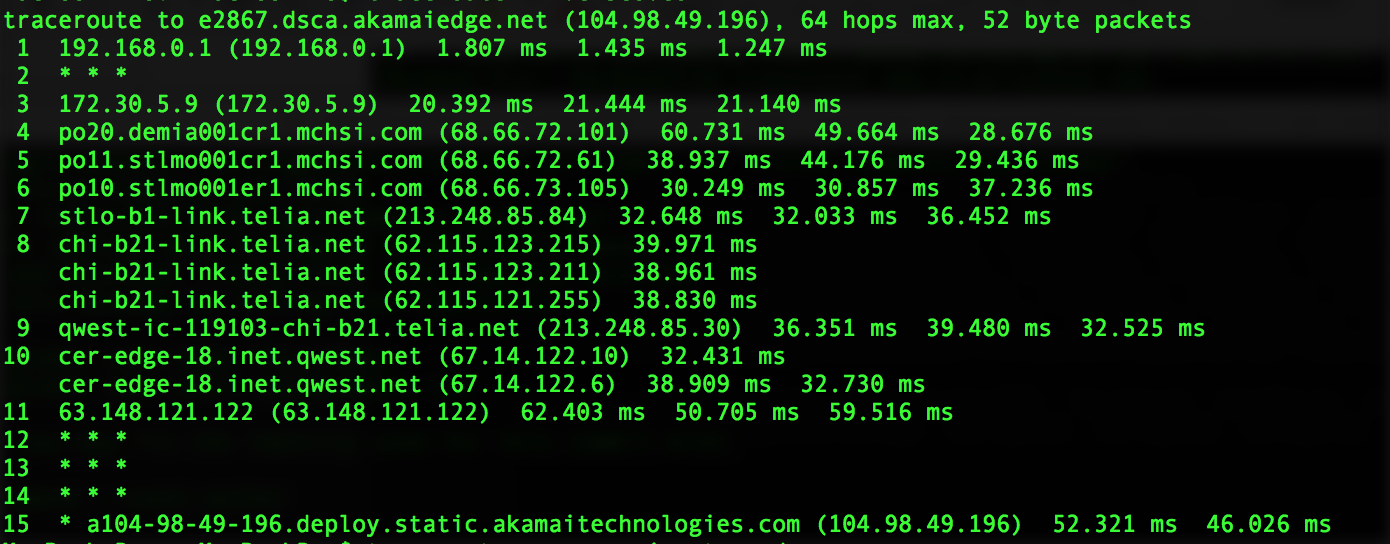
(screenshots below for ping and trace route)

37. The lesser the number of hops, the lower the ms response was. I am not sure how the geographical distance affects the number of hops because Princeton’s servers timed out during ping and traceroute.

(screenshots below for ping and trace route)







BONUS

Question: Consider a point-to-point link 50 km in length. At what bandwidth would propagation delay (at a speed of 2×10^8m/s) equal transmit delay for 100-byte packets? What about 512-byte packets?

100 Byte packets:

Propagation delay = Distance / propagation speed = 50000m / (2\*(10^8)) m/s = 25\*10^-5 s

Bandwidth = size of packet / transmit delay = (100\* 8) bits / 25\*10^-5 s = 3200000

3200000/ 10^6 = 3.2MBPS

The propagation delay equation is the distance / propagation speed. I converted km to m, and then computed the size of the packet by multiplying 100 bytes \* 8 bits in 1 byte. This gave the total. I then used the bandwidth equation to calculate the speed per second. I also converted bits to MBs by dividing by 10^6.

512 Byte packets:

Bandwidth = size of packet / transmit delay = (512 \*8) bits / 25\*10^-5 s = 16384000

16384000/ 10^6 = 16.384MBPS

I did the same thing as before using the same propagation delay computation. The answer is different because the packet size is now 512 bytes. The same calculation occurred but when I converted bits to bytes we came to a total of 16.384MBPS. This is much faster than the smaller packet size.