CSCI 4273 Fall 2014 **Vince Coghlan**

Homework #1

1: first I tried "whois princeton.edu" and got:

Domain Name: PRINCETON.EDU

Registrant:
Princeton University
Office of Information Technology
701 Carnegie Center, Suite 302
Princeton, NJ 08540
UNITED STATES

Administrative Contact:
Princeton University
Contact for Internet name and number resources
Princeton University
Office of Information Technology
701 Carnegie Center, Suite 302
Princeton, NJ 08540
UNITED STATES
(609) 258-8700
netmaster@princeton.edu

Technical Contact:
Princeton University
Contact for Internet name and number resources
Princeton University
Office of Information Technology
701 Carnegie Center, Suite 302
Princeton, NJ 08540
UNITED STATES
(609) 258-8700
netmaster@princeton.edu

Name Servers:

DNS.PRINCETON.EDU	128.112.129.15
DIKAHBLE.PRINCETON.EDU	128.112.134.4
NS1.FAST.NET	
NS2.FAST.NET	
ADNS1.UCSC.EDU	
ADNS2.UCSC.EDU	

Domain record activated: 03−Apr−1987
Domain record last updated: 05−Oct−2010
Domain expires: 31−Jul−2015

Then for "whois princeton" I got a list of domains that the search feature found.

2:

(a) The total time to receive all bytes is the handshake, the transmit time, and time it takes to propegate. This is:

$$2 \cdot 100\text{ms} + 1000\text{KB}/1.5\text{Mbps} + \frac{1}{2}100\text{ms} = 5.71 \text{ seconds}$$

(b) We will be sending 1000 packets (1000KB/1KB). This means there will be an entire RTT for each packet, or:

$$5.71\text{seconds} + 1000 \cdot 100\text{ms} = 105.71 \text{ seconds}$$

(c) We will be sending data $1000/20 = 50$ times. This means:

$$\text{time} = 2 \cdot 100\text{ms} + 50 \cdot 100\text{ms} + \frac{1}{2}100\text{ms} = 5.25 \text{ seconds}$$

(d) To find out how many times we must send, we can look at the sum:

$$\sum_{k=1}^{n} 2^{k-1}$$

when n is 10, this number is 1023, the closest thing above 1000. This means that we

will need to send 10 times. Our time ends up being:

$$2 \cdot 100\text{ms} + 10 \cdot 100\text{ms} + \frac{1}{2}100\text{ms} = 1.25 \text{ seconds}$$

3: Multicast is more useful than unicast if you have many recipients. You only need to send the data once (or as much as the bandwidth will let you) instead of having to send data once for every target machine. It is better than broadcast in some circumstances if you dont want your data to go to anyone.

4: STDM is useful for the phone system since when you are listening to a voice, you want to be able to hear all the frequency ranges. You wouldnt want to be limited to what you could hear and have to listen to different people in different frequencies. FDM is good for broadcast networks since the channels are always on, and you know that each one will want to communicate at the same time. If you want to add a channel to television, you just allocate another slot. Neither of these methods are good for a general purpose network, since you would need to allocate a time or frequency unit to every single line, and you are not sure if they are going to send or not at any given time.

5:

(a)

$$\text{RTT} = \frac{2 \cdot \text{distance}}{\text{speed}} = \frac{2 \cdot 385000000}{3 \cdot 10^8} = 2.6 \text{ seconds}$$

(b)

$$\text{delay} \times \text{bandwidth} = 2.6 \cdot 1Gbps = 2.6Gb = 320.8 \text{ GB}$$

(c) This is the maximum amount of data that can be on the network at any given time.

(d)

$$\text{time} = \text{RTT} + \frac{\text{size}}{\text{bandwidth}} = 2.6 + (\frac{25 \cdot 8}{100}) = 4.6 \text{ seconds}$$

6:

(a) The effective bandwidth is 100 Mbps, since thats how fast the switches are. The switches are just sending the data down the line.

(b) The delay due to bandwidth is 12000 bits/100MBps, or 120μs. The total delay includes propegation delay of 10. With three switches, like in this setup, the delay will be:

$$120\mu\text{s} + 4 \cdot 10\mu\text{s} = 160\mu\text{s}$$

The acknowledgement will take half a microsecond per link to be sent. this means that the delay is:

$$4\mu\text{s} + 4 \cdot 10\mu\text{s} = 44\mu\text{s}$$

for a grand total of 204μs. 12000 bits in 204μs is 58.8Mbps.

(c)

$$\frac{100 \cdot 4.7}{12 \text{ hours}} = 93.46 \text{ Mbps}$$

7:

(a) Setting the transmission time equations together we find:

$$\frac{0.5\text{MB}}{t-1} = \frac{0.4\text{MB}}{t-2} \Rightarrow t = 6$$

meaning that the bandwidth is:

$$\frac{0.5\text{MB}}{5} = 0.84 \text{ Mbps}$$

(b) Since both messages are sent through the same channel, they have the same latency.

9: cs.princeton.edu would not respond to my pings, but princeton.edu had a ping of around 60ms and cisco.com had a ping of about 12ms. The reason for the descrepency can be seen if you type in traceroute. Princeton is sending the traffic through more than twice as many places as cisco is. Also cisco is a private corperation who can afford better equipment than an educational institution. The output is:

```
PING www.princeton.edu (128.112.132.86): 56 data bytes
64 bytes from 128.112.132.86: icmp_seq=0 ttl=47 time=60.996 ms
64 bytes from 128.112.132.86: icmp_seq=1 ttl=47 time=60.056 ms
64 bytes from 128.112.132.86: icmp_seq=2 ttl=47 time=67.742 ms
64 bytes from 128.112.132.86: icmp_seq=3 ttl=47 time=59.036 ms
^C
```

```
--- www.princeton.edu ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 59.036/61.957/67.742/3.411 ms


PING e144.dscb.akamaiedge.net (23.4.128.211): 56 data bytes
64 bytes from 23.4.128.211: icmp_seq=0 ttl=58 time=11.128 ms
64 bytes from 23.4.128.211: icmp_seq=1 ttl=58 time=11.017 ms
64 bytes from 23.4.128.211: icmp_seq=2 ttl=58 time=13.449 ms
^C
--- e144.dscb.akamaiedge.net ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 11.017/11.865/13.449/1.121 ms
```

I cant measure this at different times of the day since I am doing the assignment minutes before it is due.

10: We have already seen in the last problem how more servers means a higher RTT. We can see the output from some sites below.

```
traceroute: Warning: www.google.com has multiple addresses; using 74.125.225.
traceroute to www.google.com (74.125.225.211), 64 hops max, 52 byte packets
 1  192.168.0.1 (192.168.0.1)  2.016 ms  1.307 ms  0.981 ms
 2  c-71-196-136-1.hsd1.co.comcast.net (71.196.136.1)  16.265 ms  8.405 ms
10.199 ms
 3  te-8-2-ur01.boulder.co.denver.comcast.net (68.86.129.229)  11.615 ms
8.983 ms  10.040 ms
 4  te-7-4-ur02.boulder.co.denver.comcast.net (68.86.103.122)  10.329 ms
9.982 ms  10.027 ms
 5  xe-13-3-1-0-ar01.aurora.co.denver.comcast.net (68.86.179.81)  12.549 ms
8.700 ms  11.375 ms
 6  te-0-1-0-4-cr01.chicago.il.ibone.comcast.net (68.86.95.201)  14.244 ms
    he-3-10-0-0-cr01.denver.co.ibone.comcast.net (68.86.92.25)  12.609 ms
    68.86.87.89 (68.86.87.89)  15.816 ms
 7  xe-5-1-0-0-pe01.910fifteenth.co.ibone.comcast.net (68.86.82.206)
12.122 ms  10.825 ms  11.702 ms
 8  as15169-1-c.910fifteenth.co.ibone.comcast.net (23.30.206.106)
```

```
25.539 ms  25.615 ms  20.508 ms
 9  72.14.234.57 (72.14.234.57)  12.474 ms  12.747 ms  10.903 ms
10  209.85.251.111 (209.85.251.111)  12.827 ms  13.359 ms  17.532 ms
11  den03s06-in-f19.1e100.net (74.125.225.211)  9.317 ms  12.696 ms
12.408 ms

traceroute to www.colorado.edu (128.138.129.98), 64 hops max, 52 byte packets
 1  192.168.0.1 (192.168.0.1)  1.103 ms  1.300 ms  1.034 ms
 2  c-71-196-136-1.hsd1.co.comcast.net (71.196.136.1)  10.144 ms  9.291 ms
8.990 ms
 3  te-8-2-ur01.boulder.co.denver.comcast.net (68.86.129.229)  10.006 ms
8.746 ms  9.860 ms
 4  xe-13-3-1-0-ar01.denver.co.denver.comcast.net (68.86.103.157)
10.689 ms  10.992 ms  9.444 ms
 5  te-4-8-ur01.denver.co.denver.comcast.net (68.86.104.106)  12.347 ms
11.723 ms  10.915 ms
 6  68.86.128.18 (68.86.128.18)  10.155 ms  11.554 ms  10.180 ms
 7  ucb-i1-frgp.colorado.edu (198.59.55.9)  10.265 ms  10.653 ms  12.161 ms
 8  fw-juniper.colorado.edu (128.138.81.194)  12.178 ms *  12.508 ms
 9  hut-fw.colorado.edu (128.138.81.249)  14.506 ms  12.155 ms  11.904 ms
10  comp-hut.colorado.edu (128.138.81.11)  12.249 ms  11.552 ms  11.517 ms
11  www.colorado.edu (128.138.129.98)  11.603 ms  12.122 ms  11.366 ms

traceroute to thepiratebay.se (194.71.107.27), 64 hops max, 52 byte packets
 1  192.168.0.1 (192.168.0.1)  1.800 ms  1.687 ms  1.463 ms
 2  c-71-196-136-1.hsd1.co.comcast.net (71.196.136.1)  8.998 ms  9.312 ms
10.313 ms
 3  te-8-2-ur01.boulder.co.denver.comcast.net (68.86.129.229)  8.880 ms
9.171 ms  9.746 ms
 4  te-7-4-ur02.boulder.co.denver.comcast.net (68.86.103.122)  9.636 ms
9.087 ms  9.349 ms
 5  xe-13-3-2-0-ar01.aurora.co.denver.comcast.net (68.86.179.97)  11.518 ms
11.680 ms  10.332 ms
 6  he-3-4-0-0-cr01.denver.co.ibone.comcast.net (68.86.90.149)  15.268 ms
    he-3-9-0-0-cr01.denver.co.ibone.comcast.net (68.86.92.21)  13.843 ms
    68.86.87.89 (68.86.87.89)  15.150 ms
```

```
 7  be-16-pe04.ashburn.va.ibone.comcast.net (68.86.84.226)  28.342 ms
26.085 ms  28.247 ms
 8  he-0-12-0-0-pe03.1950stemmons.tx.ibone.comcast.net (68.86.86.178)
28.485 ms  25.271 ms  26.069 ms
 9  ae-19.r07.dllstx09.us.bb.gin.ntt.net (129.250.66.29)  26.523 ms
25.708 ms  25.435 ms
10  * * *
11  ae-3.r20.asbnva02.us.bb.gin.ntt.net (129.250.3.50)  59.842 ms
59.072 ms  58.726 ms
12  * ae-0.r21.asbnva02.us.bb.gin.ntt.net (129.250.4.5)  60.351 ms *
13  * * ae-2.r23.amstnl02.nl.bb.gin.ntt.net (129.250.2.145)  195.423 ms
14  ae-2.r02.amstnl02.nl.bb.gin.ntt.net (129.250.2.159)  270.771 ms
307.288 ms  194.190 ms
15  xe-4-1.r02.dsdfge01.de.bb.gin.ntt.net (129.250.2.65)  155.367 ms
147.905 ms  152.986 ms
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
(And it just keeps going like that, lol piratebay)

traceroute to localhost (127.0.0.1), 64 hops max, 52 byte packets
 1  localhost (127.0.0.1)  0.133 ms  0.029 ms  0.027 ms
```

Pinging myself took no time at all!