Homework 3

1 DNS

Suppose you have a Host C, a local name server L, and authoritative name servers A_{root}, A_{com}, and $A_{google.com}$, where the naming convention $A_{\mathbf{x}}$ means that the name server knows about the name zone **x**. $A_{\mathbf{x}}$ is a variable and **NOT** a hostname. A_{root} is a root name server known to L, with IP address 198.41.0.4. Assume that all name servers initially have nothing in their caches.

a) Using the resource records below, provide the hostnames and IP addresses for A_{com} and $A_{google.com}$

b) List the sequence of DNS queries and corresponding resource records that are exchanged when C wants to lookup the address for www.google.com. When a name server $A_{\mathbf{x}}$ contacts $A_{\mathbf{y}}$, please specify the IP address for $A_{\mathbf{y}}$. The following represents the level of detail we are looking for:
i) $A_{\mathbf{x}}$ contacts $A_{\mathbf{y}}$ at 1.2.3.4 to resolve www.foobar.com
ii) $A_{\mathbf{y}}$ returns resource records $R_1, R_2, \ldots, R_n$.

c) Client C performs a lookup for mail.google.com immediately after the previous request. Assume all records have long TTLs. List the sequence of DNS queries and corresponding resource records exchanged during the lookup.

Name Server Variable	Resource Record
A_{root}	{com, a.gtld-servers.net, NS, IN}
A_{root}	{a.gtld-servers.net, 192.5.6.30, A, IN}
A_{com}	{google.com, ns1.google.com, NS, IN}
A_{com}	{ns1.google.com, 216.239.32.10, A, IN}
$A_{google.com}$	{www.google.com, 66.102.7.104, A, IN}
$A_{google.com}$	{mail.google.com, 66.102.7.83, A, IN}

Answer

a)

A_{com}: hostname=a.gtld-servers.net, IP address=192.5.6.30
$A_{google.com}$: hostname=ns1.google.com, IP address=216.239.32.10

Notes: Many people missed these because they assumed that the hostnames and IP addresses merely corresponded to the resource record listed next to the variable.

b)

1) C queries L to resolve www.google.com
2) L queries A_{root} at 198.41.0.4 to resolve www.google.com.
3) A_{root} returns {com, a.gtld-servers.net, NS, IN} and {a.gtld-servers.net, 192.5.6.30, A, IN}.
4) L queries A_{com} at 192.5.6.30 to resolve www.google.com
5) A_{com} returns {google.com, ns1.google.com, NS, IN} and {ns1.google.com, 216.239.32.10, A, IN}
6) L queries $A_{google.com}$ at 216.239.32.10 to resolve www.google.com
7) $A_{google.com}$ returns {www.google.com, 66.102.7.83, A, IN}
8) L returns 66.102.7.83 to C.

c)

1) C queries L to resolve mail.google.com
2) L queries $A_{google.com}$ at 216.239.32.10
3) $A_{google.com}$ returns {mail.google.com, 66.102.7.104, A, IN}
4) L returns 66.102.7.104 to C.

2 TCP-like Design

NASA has built a series of satellites to orbit the Earth, except they lack the requisite knowledge to design a communication protocol. They task you with designing a reliable, sliding window, bytestream protcol similar to TCP for communication over a link with a bandwidth of 1 Gbps, an RTT of 384 ms, and a maximum segment lifetime of

15 sec. The protocol will not include a window scaling option.

a) How many bits wide should you make the *AdvertisedWindow* and *SequenceNum* fields? Explain why you select your values

b) The person in charge of the program scoffs at your suggestions, and instead wishes to use an *AdvertisedWindow* of 16 bits. Explain why this is problematic. c) Suppose you settle on using TCP as your communication protocol. Two satellites, S_1 and S_2 want to communicate with a third satellite. Both S_1 and S_2 are in the Additive Increase, Multiplicative Decrease (AIMD) phases of their transmissions, but S_2 has a much higher RTT than S_1. As such, you observe that S_2 is not receiving its fair share of the bandwidth. Explain how, under AIMD, the TCP congestion window grows with respect to the RTT. Using that intuition, explain why the bandwidth is not being shared fairly between S_1 and S_2.

Answer

a)

To fully utilize the network, the *AdvertisedWindow* needs to be larger than the *Delay*Bandwidth*.

$$\begin{aligned} AdvertisedWindow \geq & Delay * Bandiwidth \\ & = 384\text{ms} * 1\text{Gbps} \\ & = 48\text{MB} \\ & = 50,331,648 \text{ bytes} \end{aligned}$$

Since $2^{25} = 33,554,432$ and $2^{26} = 67,108,864$, the *AdvertisedWindow* must be 26 bits.

SequenceNum must not wrap around before any delayed segments have left the network. Presumably, this occurs within the maximum segment lifetime. Thus, the *SequenceNum* must be larger than the *MSS * Bandwidth*.

$$\begin{aligned} SequenceNum \geq & MSS * Bandiwidth \\ & = 15\text{sec} * 1\text{Gbps} \\ & = 1920\text{MB} \\ & = 2,013,265,920 \text{ bytes} \end{aligned}$$

Since $2^{30} = 1,073,741,824$ and $2^{31} = 2,147,483,648$, the *SequenceNum* must be 31 bits.

Notes: Many people got the correct justification, but forgot to convert the bits to bytes.

b)

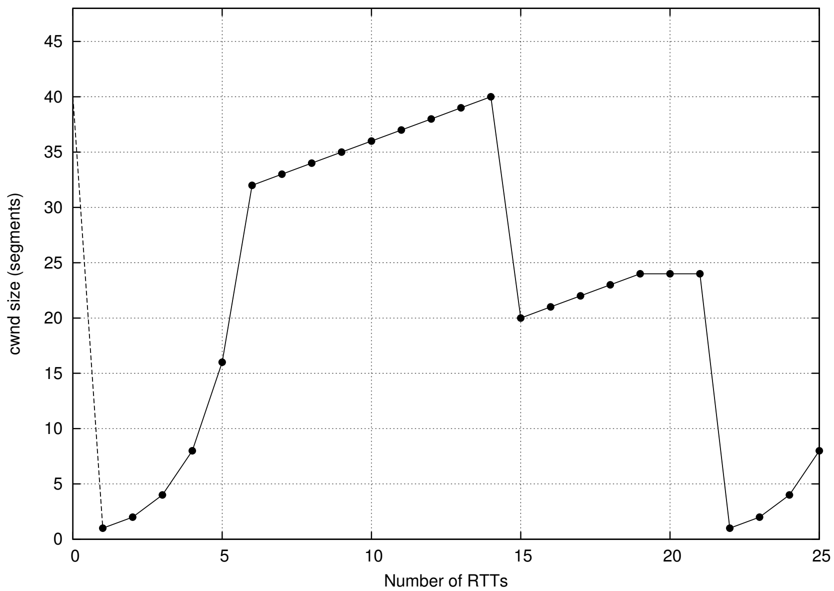
The *AdvertisedWindow* is smaller than *Delay* Bandwidth*; thus, it is limiting the effective bandwidth. Only an *AdvertisedWindow* worth of data can be transferred during a single RTT. The effective bandwidth is

$$\begin{aligned}
EffectiveBandwidth &= AdvertisedWindow/RTT \\
&= 2^{16} \text{ bytes}/384 \text{ ms} \\
&= 166.67 \text{ KB/sec} \\
&= 1.30207825 \text{ Mbits/sec}
\end{aligned}$$

c)

With the AIMD mechanism, the TCP congestion window increases by 1 for each RTT. A flow with a longer RTT needs more time to increase its congestion window size, compared to a flow with a short RTT. The flow with the longer RTT always has a smaller congestion window size and receives less bandwidth.

3 TCP/Congestion Control



Assume TCP Reno (meaning Fast Retransmit and Fast Recovery) is the protocol experiencing the behavior shown above. Assume that the TCP flow has been operating for some time, meaning that the number of RTTs shown are with respect to when you started observing the flow's behavior.

a) Identify the time periods when TCP slow start is operating.
b) Identify the time periods when TCP congestion avoidance is operating (AIMD).
c) After the 14th RTT, is the segment loss detected by a triple duplicate ACK or by a timeout?
d) What is the initial value of ssthreshold, before the first congestion avoidance interval?
e) What is the value of ssthreshold at the 19th RTT?
f) What is the value of ssthreshold at the 24th RTT?
g) Assuming a packet loss is detected after the 25th RTT by the receipt of a triple duplicate ACK, what will be the values of the congesion-window size and of ssthreshold?

Answer

a) 1-6, 22-25
b) 6-22 (or 19, if you consider 20-22 a"timeout" phase)
c) Duplicate Acknowledgment. If a timeout happened, cwnd would drop to 1
d) 32. At 32, the flow exits slow start and proceeds to AIMD.

e) 20. At a cwnd size of 40, the cwnd is halved to 20 and the ssthreshold is set to 20.
f) 12, At a cwnd size of 24, a timeout occurs. ssthreshold is set to half of 24 (12), and the cwnd size drops to 1.
g) 4,4. The triple dup ack causes the ssthreshold to be set to half the value of cwnd, which was previously 8 at the conclusion of the 25th RTT. Due to fast recovery, the cwnd is set to 4, or 7 if you considered inflation.

Notes: For part a), I wasn't too strict about the inclusive vs exclusive ranges. For part b), I also accepted 19 as an upper range. Most people who lost points on this problem missed part g). It looked like people assumed the cwnd kept growing. However, part g) states that a triple duplicate ack is detected after the 25th RTT, meaning that some packet was lost when the cwnd was 8. I accepted either 4 or 7 (inflation) as the answer for the cwnd value.

4 TCP/Congestion Control

Host A sends a file consisting of 9 MSS-sized segments to a host B using TCP. Assume that the 4th segment in the transmission is lost. Assume the retransmission timeout is T, the one-way latency is d, and that T > 4*d. Ignore the transmission time of the segments and of the acknowledgements. Also, assume the TCP three-way handshake has completed, but no data has been transmitted.

a) Assume no fast retransmission or fast recovery. Draw the time diagram showing each segment and acknowledgement until the entire file is transferred. Indicate on the diagram all changes in the cwnd and ssthresh. How long does it take to transfer the file?

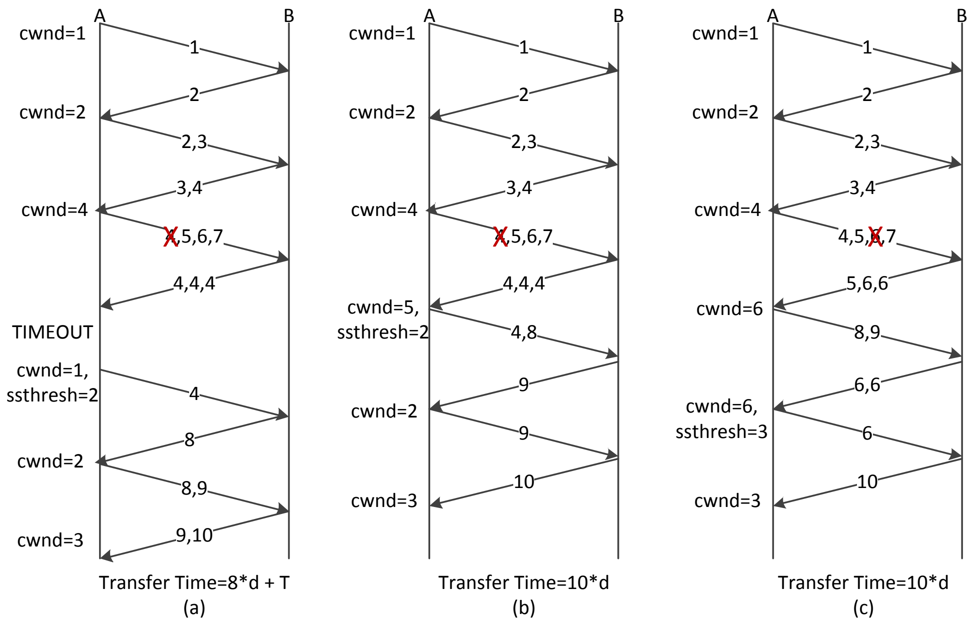
b) Answer part (a) assuming TCP Reno, i.e., the TCP version that implements both fast retransmission and fast recovery.

c) Answer part (b) assuming that only the 6th segment is dropped.

Notes

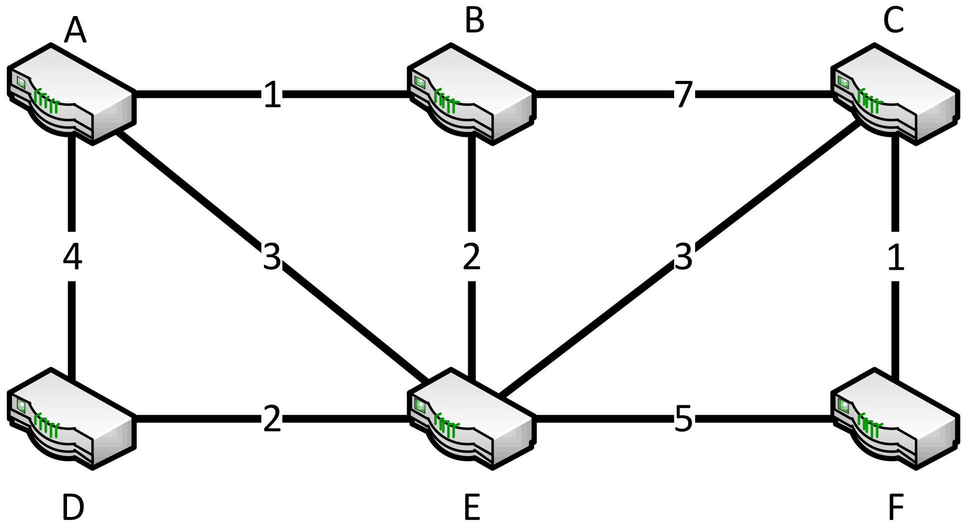
- For Fast Recovery, assume that each duplicate acknowledgment increases cwnd by 1.
- For Fast Recovery, assume that, upon receiving a non-duplicate acknowledgment, cwnd drops back to ssthresh.
- If the value of cwnd is fractional, you should round it to the closest larger integer.
- The transfer time is the time interval measure at source A from the time the first segment is sent until the acknowledgement of the last segment is received

Answer



Notes: People lost the most number of points on this problem. People indicated on their timing diagrams that segments 5,6 and 7 are resent; I automatically deducted 2.5 pts for the first subproblem, and 2 pts for the second subproblem if you did this. You should know by now (after doing the project and seeing this material in lecture/discussion) that the receiver has a buffer to place out of order segments. Also, people forgot to read the lecture notes about the ACKing conventions; I explicitly posted on moodle that you should review those conventions before doing this problem. Also, note that when you inflate your window in b), the sender has the ability to send out one additional segment, namely, 8. I included an additional subproblem c) if you would like more practice on this type of problem.

5 Link State



Given the following network topology, fill in the table showing the step- by-step operation of the link-state algorithm for node B.

Step	Confirmed	Tentative

Answer

Step	Confirmed	Tentative
1	(B,0,-)	
2	(B,0,-)	(A,1,A), (C,7,C), (E,2,E)
3	(B,0,-), (A,1,A)	(C,7,C), (E,2,E)
4	(B,0,-), (A,1,A)	(C,7,C), (E,2,E), (D,5,A)
5	(B,0,-), (A,1,A), (E,2,E)	(C,7,C), (D,5,C)
6	(B,0,-), (A,1,A), (E,2,E)	(C,5,E), (D,4,E), (F,7,E)
7	(B,0,-), (A,1,A), (E,2,E), (D,4,E)	(C,5,E), (F,7,E)
8	(B,0,-), (A,1,A), (E,2,E), (D,4,E), (C,5,E)	(F,6,E)
9	(B,0,-), (A,1,A), (E,2,E), (D,4,E), (C,5,E) (F,6,E)	

Notes: People lost minor points on this problem. People did not notice that to reach F, you go through E, then C, then F; this gives you a cumulative cost of 6. I only looked at the last entry in your table.