Name or UFID: ________________________________

CNT5106C Computer Networks

Prof. Ahmed Helmy

Midterm Exam

Instructions:

1. Write your name and gator UFID on this page. Also write your name (or last 4 digits of your gator UFID) at the top of every page.
2. Please read the instructions for each question.
3. The exam is Closed Book and Closed Notes. The formulas you may need are at the end of this exam paper.
4. Check that your exam is complete with 10 pages (including 1 page formulas at the end).
5. Please note the number of points assigned to each question and budget your time accordingly.
6. The time of the exam is 1 hour and 30 minutes.

Name:____

UFID:____

	Maximum	Score
Page 2	6	
Page 3	6	
Page 4	12	
Page 5	9	
Page 6	14	
Page 7	12	
Page 8	8	
Page 9	11	
Total	**78 (70 + 8 extra)**	

For the multiple choice questions please write down the letter corresponding to the best answer in the space provided (on the dotted line). You must give exactly 'one' answer for each question. In general you are not required to write down the steps of the solution unless you are explicitly asked to 'Clarify'.

1. Rate based vs. TCP congestion control

(3 points) A new congestion control mechanism for the Internet was designed. The new mechanism follows the same principles of rate adaptation used in ATM ABR with a rate increase factor (RIF) of 1/16 and a rate decrease factor (RDF) of 1/16. If long flows using this mechanism share a bottleneck (i.e., congested) link with long TCP flows. What can be said about the fairness of the bandwidth distribution over long periods: [assume no explicit rate (ER) notification is used]

A- TCP is AIMD while the rate adaptation is not, and so TCP will have more throughput

B- TCP will use slow start and capture more bandwidth while the rate adaptation will be slow to increase its bandwidth, and so overall TCP will win

C- TCP will cut down its window size to 1 with every timeout, while rate adaptation will be slow to react to congestion and so rate adaptation will have more throughput

D- When bandwidth is available TCP will open up its window size quickly and rate adaptation will increase slowly, so TCP will have more throughput on average

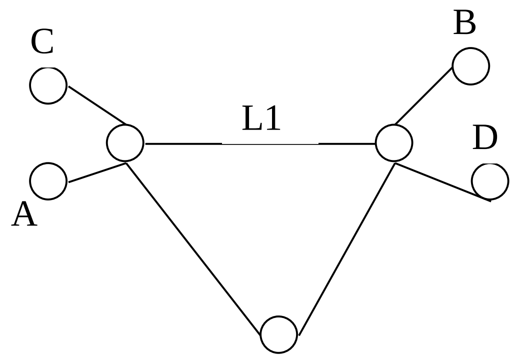
Answer: …………C………..

2. (3 points) In a rate-based congestion control mechanism, the rate increase is done according to the relation rate = rate + MaxRate x Inc, and the rate decrease is done according to the relation rate = rate x Dec, where Inc is the increase factor and Dec is the decrease factor. If x1>x2, both x1 and x2 < 1, which of the following rate factors would result in the *slowest* reaction to network dynamics:

A- Inc=x1, Dec=x1
B- Inc=x1, Dec=x2
C- Inc=x2, Dec=x1
D- Inc=x2, Dec=x2

Answer: ……………C……………………

Name or UFID: ________________________________



3. (3 points) Consider two TCP connections (as shown in the figure), connection 1 between nodes A&B starts at time t1 and connection 2 between C&B starts at time t2. Consider failure of link L1 at time t3, where t1<t3<t2, and then L1 heals at time t4>t2. The two connections end at time t5 >> t4. For connection 1, do you expect that 'cwnd' will be decreased:

A- at t2 only
B- at t3 only
C- at t4 only
D- at t3 and t4

Answer: ………D……………..

Clarify (mention why): …t3, link failure , packet loss, decrease cwnd, t4,link-healing, packet loss due to low bandwidth-delay-product, decrease cwnd

4. (3 points) (See previous question) At t4, do you expect:
A- Connection 1 to suffer more loss
B- Connection 2 to suffer more loss
C- Both connections to suffer the same, since TCP will cut cwnd to 1
D- Neither will suffer loss when the link heals at t4

Answer: ………B…………………………….

Clarify (mention why): …connection1 will be in cong. Avoidance (less aggressive) , while connection 2 will be in slow start (i.e., more aggressive) when the link heals ………………………………………………………………
………………………………………………………………………………………….

5. (5 total points) DHTs:

I. (2.5 points) What is a DHT and what is its function in peer-to-peer networks?

II. (2.5 points) Explain with the aid of a drawing how a circular DHT accommodates nodes joining and leaving. [Hint: briefly explain successor/predecessor settings]

Answer:

a. DHT stands for distributed hash table and is a structured (algorithmic) way of accessing the data by consistently storing and retrieving it from different nodes using the same process/algorithm using hash functions or tables. It is utilized in p2p networks to add structure to the overlay network, which aims to simplify and add efficiency to the process of searching for content and files in the p2p network.

b. [similar to Q19, 20 in hwk 1]

6. DNS *(7 points)*

a. (4.5 points) Give three reasons (arguments) against having one centralized DNS server.

Answer:

With one DNS server we have the following drawbacks: (3 are enough)

1. Single point of failure: if the DNS server crashes, so does the entire Internet
2. Traffic concentration: the single server would have to handle all DNS queries for all HTTP requests and email messages for hundreds of millions of hosts.
3. Delayed responses: since the single server can only be close to a very few hosts, most of the hosts will have to travel large distances (and experience propagation delay), and traverse many links (some of which maybe congested) to reach the server.
4. Book-keeping and updates (maintenance): the DNS server would have to keep track of every new host or every removed host in the Internet. This doesn't scale.

b. (2.5 points) Discuss a mechanism to improve DNS performance and explain how the performance can improve.

Answer:

Using DNS caching is one way to improve DNS performance, first by reducing the delay required to get the address resolution (since the cache servers are now closer to the requesting hosts), and by reducing the overall load of DNS going to the higher level DNS servers.

7. ***Peer-to-peer Networks Architecture***
   (6 points) Mention three different architectures for peer-to-peer (p2p) networks (or overlays) in the Internet. Give example of a real p2p network for each architecture. Briefly describe how the queries propagate in each of the architectures.

Answer:

1. Centralized directory of resources/files, as in Napster. The query is sent to the server that maintains all information and hence the query is answered through local lookups in the server's database, with minimum overhead on the network.

2. Fully distributed, non-centralized architecture, as in Gnutella, where all peers and edges form a 'flat' overlay (without hierarchy). Search and query is more involved and incurs high overhead with flooding (or broadcast).

3. Hierarchical overlay, with some nodes acting as super nodes (or cluster heads), or nodes forming loose neighborhoods (sometimes referred to as loose hierarchy, as in BitTorrent or Skype).
   Avoids flooding to search for resources during queries. The query is send from the requesting peer to its cluster head, then from the cluster head to the other cluster heads. Each cluster head keeps track of the resources in its cluster.

8. (3 points) In the ATM ABR rate control, if the explicit feedback indicates congestion, then the equation for reduction is invoked cause rate = rate – rate * RDF. Someone suggested that this may not be aggressive enough during severe congestion and that 'multiplicative' decrease should be used by having rate = rate * RDF. What do you think of this argument? Do you agree, or disagree, and why?

Answer:
rate = rate – rate * RDF = rate * (1 – RDF) = rate * alpha

hence, the 2 equations give the same effect and the difference is in the way the parameter RDF is set… hence, I disagree

9. (10 points) Stop-and-wait protocol is used for flow control over a link that is 2km long, with 10Mbps capacity, and 10% packet error rate. Packet length is 1000bits.

i. (2 points) What is the utilization of this protocol?

For stop-and-wait: $u = \dfrac{1-p}{(1+2a)}$, where p=0.1 is the frame error probability

transmission delay=1000/10^7=100μsec, propagation delay=2.10^3/2.10^8=10μsec

$a = \dfrac{Tprop}{Trans} = \dfrac{propagationDelay}{transmissionDelay}$ =10/100=0.1

Utilization u=0.9/(1+0.2)=0.75=75%

ii. (2 points) Assume the capacity is increased from 10Mbps to 100Mbps (with everything else being the same), what is the new utilization?

Transmission delay=1000/10^8=10, and a=1
Utilization u=0.9/(1+2)=0.3=30%

iii. (6 points) Suggest 3 ways to bring the utilization close to what it was with the 10Mbps (you cannot suggest reducing capacity to 10Mbps!).

1- Increase the frame/packet length by 10 fold to 10,000 bits.
2- Reduce the length of the link by 10 fold (from 2km to 200m)
3- Use a sliding-window mechanism Go back N or selective reject. For example, if we use selective repeat with window *w*=3, we get u=3*0.3=0.9=90%.

10. (4 points) In ATM ABR rate control, there are several forms of the explicit congestion feedback including **1.** Binary notification using the congestion indication (CI) and no-increase bits (NI) and **2.** Explicit rate (ER). Which form of notification you think should be done by the network, and do you think the network or the end points should calculate the ER or the end points? Clarify your arguments.

Answer:
- on one hand, calculating ER in the network does not take into account the application specific requirements [the end points know more about the requirements and may be in a better position to calculate ER that would be better from an application stand point]
- on the other hand, the end-points are not aware of the detailed network parameters (where is congestion, how much capacity can the congested switch support, how many other flows pass through the congested switch, what'' the fair share of each of those flows... etc.). The network (i.e., the congested switch in particular) has all these parameters and can have a better estimation of the ER from a network point of view
- One compromise that can be done is to have the network send binary (CI) notification first, to give the end-points a chance to respond and cut down their rate. If late on, the end-points' response was not sufficient to alleviate the congestion in the network, then the network can calculate the ER that would relieve congestion and sends the ER notification to the sources accordingly.

11. (9 points: 3+3+3) Imagine that an explicit congestion notification (ECN) bit was added to the IP packet header, and *all* routers in the network support its use (i.e., when a router senses congestion it sets the ECN bit in the packets).

(a) Suggest modifications to TCP such that it takes advantage of this explicit congestion notification.

(b) Mention two scenarios in which this may improve TCP's performance.

(c) If not all routers implement ECN support, mention any drawbacks (if any) of your suggested modifications in (a) or (b) above. [If you think there are none, then mention so explicitly and justify it]

Answer:

(a) i. at the receiver side the receiver needs to understand the ECN bit and include it in the Ack,
ii. at the sender side the sender needs to understand the ECN bit and cut its window (perhaps using fast re-tx fast recovery when the ECN bit is set, or cwnd=1 with 'n' ECN bits are set consecutively).
iii. the sender can now invoke the window reduction mechanism only when the ECN bit is set (because now TCP can distinguish between congestion-related loss and other (e.g., BER/error-related) loss

(b) i. Over lossy links (e.g., over wireless links where BER may be high and the errors are not due to congestion (performance of TCP would improve if it does not cut its window due to BER-related losses)
ii. instead of 'dropping' packets (that TCP would have to re-transmit) routers just set the ECN bit (and that can be done before at the onset of congestion, thus avoiding severe congestion)

(c) if the congested router does not implement ECN then the TCP (if implements the mechanism in (b) above) would not backoff thinking that it's not congestion related loss.

12. (3 points) In fast retransmit and recovery we increase the window by 1 with every Ack, isn't this similar to slow start? Would it lead to exponential opening of the window? [You can describe using a simple window drawing if you want.]

Answer:

NO. in slow start , when the ack is received two actions are taken in the window adjustment algorithm:

1- the start of the window is advanced (to the last ack'ed segment/byte), and 2- the cwnd is increased by 1

in fast retransmit fast recover the acks that are received are those for the last in-order segment/byte received by the receiver, and hence the start of the window is 'frozen' at that value,… so the only action taken is the increase of cwnd, which does not lead to exponential opening of the window as in slow start.

13. Proxy Caching:
(4 points) Discuss three advantages of having proxy caches.

Answer:

1. Enhances (and reduces or improves) the response time for the cached objects since it will be fetched from a location much closer than the original server.

2. Reduces the congestion on the access network and links (leading to the Internet cloud), where normally queue length and delays are high. This also leads to improving performance for other (non-cached) objects.

3. Hides the original querier (since the proxy is not sending the query on the original requester's behalf). This may be required in some applications for privacy or anonymity.

14. (4 points) You need to design a system that supports 100 users using packet switching (statistical multiplexing), and there are two options you need to consider. The first, is a system that consists of one link with capacity 5Gbps. The second, is a system with 10 links in parallel, each with capacity of 500Mbps. Which system would you choose and why? Assume that each source is active 40% of the time and that the rate of a source when active is 100Mbps. You may make reasonable assumptions that you see necessary.
Answer:
Input load (utilization) $\rho = \boldsymbol{\alpha}.\ N.\ R\ .\ Ts = 0.4 * N\ R\ /\ M$, where α is the active fraction of time on average, N is the number of sources, R is the rate when a source is active and M is the capacity of the link (Ts is the service time)
For both systems the utilization ρ is the same: $\rho = 0.4 * 10 * 100M/500M = 0.8=80\%$

Note that, based on the queuing theory equations (for M/M/1 and M/D/1) the average queue occupancy (or queue length) is a function of ρ only, hence the memory required for the 5Gbps link is equal to only one of the 500Mbps link. Hence the second system (with 10 * 500Mbps links) needs 10 times the buffer. This is due to the statistical multiplexing and sharing among the 100 flows in the case of the 5Gbps link.

Also, the delay incurred on each link (or in each queue) is proportional to Ts (the service time) or inversely proportional to M (the link capacity). Hence the first system (with 5Gbps link) will incur less queuing delay than each of the 500Mbps links.
So in both memory requirement and delay performance the first system is better.

15. **TCP and fairness** (11 points)
I- (3 points) What would happen if UDP flows share the same bottleneck with TCP flows? Would the capacity be split fairly between the flows? Why?

Answer:
UDP is a non-responsive flow that does not react (by dropping its rate or reducing its window) to congestion or loss. TCP reduces its window with every packet loss. So the result is that TCP would back-off and UDP would not, which results in unfairly high throughput for UDP at the expense of very small share of bandwidth for TCP flows.

II- (3.5 points) If an attacker using UDP traffic is attacking TCP flows, what do you think the best strategy would be to achieve maximum reduction of TCP throughput with the minimum number of UDP packets? (i.e., which link(s) should be the target?)

Answer:
The UDP traffic should target 'bottleneck' links (those with the min remainder of capacity) along the path(s) followed by TCP flows. As soon as the UDP traffic pushes the bottleneck link beyond its capacity (which would require the min UDP packets, as compared to other, non-bottleneck links), the TCP flows will backoff and the attack will be effective.

III- (4.5 points) Do you think the same strategy would be as effective if the attacker was using short TCP flows instead of UDP? Why? Clarify your answer.

Answer:
No, the short TCP flows are still responsive, unlike UDP, so attacking the bottleneck link will cause the attacking flows to lose momentum (by backing off) and the attack is not likely to be as effective as in the UDP case.
[extra: Short flows need some space/capacity to open up their window size, gain momentum, and then they can have an effectively bad effect on other long lived TCP flows, so attacking a link with reasonable extra capacity (but not the bottleneck link) would be a better strategy.]

Name or UFID: ______________________________

Useful formulae:

Utilization:

- $a = \frac{Tprop}{Trans} = \frac{propagationDelay}{transmissionDelay}$
- $propagationDelay = \frac{Distance}{S}$, $S=2x10^8 m/s$
- For stop-and-wait: $u = \frac{1-p}{(1+2a)}$, where p is the probability that a frame is in error.

Utilization for sliding-window mechanisms with window of w:

- Go back N: $u = \frac{1-p}{1+2ap}$, if w fills the pipe, or $u = \frac{w(1-p)}{(1+2a)(1-p+wp)}$ otherwise
- Selective repeat: $u = (1-p)$, if w fills the pipe, or $u = \frac{w(1-p)}{(1+2a)}$ otherwise
- M/D/1: queuing delay $Tq = \frac{Ts(2-\rho)}{2.(1-\rho)}$; Ts is service time & ρ is link utilization
- M/D/1: average queue length or buffer occupancy $q = \lambda.Tq = \rho + \frac{\rho^2}{2.(1-\rho)}$
- M/M/1: queuing delay $Tq = \frac{Ts}{(1-\rho)}$, buffer occupancy: $q = \frac{\rho}{(1-\rho)}$
- TCP:
  - slow start CongWin+=1 per ACK,
  - congestion avoidance CongWin+=1 per RTT,
  - EstimatedRTT(k)= (1-α)*EstimatedRTT(k-1) + α*SampleRTT(k), $0<\alpha<1$
  - DevRTT= (1-β)*DevRTT+ β*|SampleRTT – EstimatedRTT|, $0< \beta <1$
  - TimeoutInterval = EstimatedRTT + 4*DevRTT

ATM ABR rate-based congestion control:

- Increase: Rate = min(PCR, Rate + PCR x RIF)
- Decrease: Rate = max(MCR, min[ER, Rate – Rate x RDF])

Probability distributions and stochastic processes:

- Geometric distribution: ***x*** is the number of Bernoulli experiments until success, $Pr[X=k]=q^{k-1}p$, $E(X)=1/p$
- Binomial distribution: ***x*** is the number of successes in ***n*** Bernoulli experiments/trials

  $P(X=k) = \binom{n}{k} q^{n-k} p^k, \binom{n}{k} = \frac{n!}{(n-k)!k!}$, E[X]=np
- Poisson Distribution: $Pr[X=k]= (\lambda^k/k!)\, e^{-\lambda}$, $E[X]=Var[X]= \lambda$
- Exponential distribution: $f(x)=\lambda e^{-\lambda x}$, $F[x]=1-e^{-\lambda x}$, $Pr[X>x]=1-F[x]=e^{-\lambda x}$, $E[X]=1/\lambda$