CNA 2015 Exam can be found at:
https://docs.google.com/document/d/1xf5QyModQj53FavI0z1qI7zB8tvbDrkuNSCEGGE-xl0/edit?usp=sharing

BLACK: EXAM QUESTIONS
BLUE: CLINT'S CURRENT ANSWERS
RED: FEEDBACK/ANSWERS FROM ANYONE ELSE FOR CLINT TO MODERATE/MODIFY
MAGENTA: REQUESTS FOR CLARIFICATION

Application Layer

Question 1 [Total for Question 1: 29 marks]

(a) What information is used by a process running on one host to identify a process running on another host? [2 marks]
IP address and Port Number

(b) Why does HTTP run on top of TCP rather than on UDP? [2 marks]
Because TCP is more reliable than UDP because TCP is connection oriented where there is a guarantee of the transmitted packet in reaching the destination, and HTTP requires all data to be received without gaps in the correct order

(c) Consider an HTTP client that wants to retrieve a Web document at a given URL. The IP address of the HTTP server is initially unknown. What transport and application-layer protocols besides HTTP are needed in this scenario (list all that will be used)? [3 marks]
DNS request is sent out over UDP to obtain the IP address mapped to the hostname
HTTP client then establishes a TCP connection with the server

Application layer protocols: DNS and HTTP
Transport layer protocols: UDP for DNS; TCP for HTTP

(d) Consider a short, 10-meter link, over which a sender can transmit at a rate of 150 bits/sec in both directions. Suppose that packets containing data are 100,000 bits long, and packets containing only control (e.g., ACK or handshaking) are 200 bits long. Assume that N parallel connections each get 1/N of the link bandwidth. Now consider the HTTP protocol, and suppose that each downloaded object is 100 Kbits long, and that the initial downloaded object contains 10 referenced objects from the same sender. Firstly, consider parallel downloads via parallel instances of non-persistent HTTP. Secondly, consider persistent HTTP. Do you expect significant performance gains over the non-persistent case? Justify and explain your answer (please clearly show your reasoning and summary calculations). [12 marks]

Note that each downloaded object can be completely put into one data packet. Let Tp denote the one-way propagation delay between the client and the server. First consider parallel downloads via non-persistent connections. Parallel download would allow 10 connections share the 150 bits/sec bandwidth, thus each gets just 15 bits/sec. Thus, the total time needed to receive all objects is given by:

--Why 11 (connections + sends)? (Calc below)--

(200/150+Tp + 200/150 +Tp + 200/150+Tp + 100,000/150+ Tp) + (200/(150/10)+Tp + 200/(150/10) +Tp + 200/(150/10)+Tp + 100,000/(150/10)+ Tp) = 7377 + 8*Tp (seconds)

Then consider persistent HTTP connection. The total time needed is give by:

(200/150+Tp + 200/150 +Tp + 200/150+Tp + 100,000/150+ Tp) + 10*(200/150+Tp + 100,000/150+ Tp) =7351 + 24*Tp (seconds)

Assume the speed of light is 300*10^6 m/sec, then Tp=10/(300*10^6)=0.03 microsec. Tp is negligible compared with transmission delay

Thus, we see that the persistent HTTP does not have significant gain (less than 1 percent) over the nonpersistent case with parallel download.

(e)
i. The equation for determining the download time for a peer-to-peer file sharing is given below.
D_p2p>max(F/U_s,F/d_min,NF/(u_s+SUM(u_i)))
Explain what each of the following terms represents (for example:"The time for one peer to download a single copy of the file") [3 marks]

1. F/u_ s = The time needed for a server to upload one copy of the file
2. F/d_min = The time needed for a slowest client to download one copy of the file
3. u_s+SUM(u_i) = The total upload capacity of the system as a whole

= bandwidth for all peer and server in P2P mode

u_s=server upload rate
d_min=download rate (slowest)
F=file size

ii. Using the same variables (F, u, d) give an equation for the minimum download time for a client/server file download. [3 marks]
= max(F/d, F.u)
= F/min(u,d)
Both good answers!

(f) Consider the network shown in figure 1. The hosts in the left network are requesting resources using HTTP from the origin servers on the Internet. 30% of web requests result in a cache hit (the request has been previously downloaded and can be served from the cache).

Each host (not including the cache) generates a request every 20 seconds and each request requires the download of 1Mbit. What download bandwidth is needed on the access link, L1, in order for this HTTP download traffic to utilise a no more than 80% of the link download Capacity? [4 marks]

0.175 Mbits/s

(4*1*0.7)/20 = 0.14/0.8 = 0.175 Mb/s

You should take clients only not cache and gateway

Shit i see it. I would miss where it shows the Cache in the exam lol.

Transport Layer

Question 2

(a) Answer True (T) or False (F). Please note that each wrong answer will incur a penalty of 1 mark so guess at your own risk. Please only write the number of the statement and T or F in your answer books. [12 marks]

1. Each TCP socket is identified with a 2-tuple: source port number;and destination port number.

F - is 4-tuple (source/destination) * (IP/port)

2. Host A is sending Host B a large file over a TCP connection. Assume Host B has no data to send Host A. Host B will not send acknowledgements to Host A because Host B cannot piggyback the acknowledgements on data.

F - All data sent in TCP gets ACKed

3. Suppose Host A is sending Host B a large file over a TCP connection. The number of unacknowledged bytes that A sends cannot exceed the size of the receive buffer.

T

F. Even ignoring that TCP will send bytes periodically when the receive buffer is 0 there are other examples where this is possible.

4. Suppose Host A sends one segment with sequence number 38 and 4 bytes of data over a TCP connection to Host B. In this same segment the acknowledgement number must be 42.

False. The acknowledgement number has nothing to do with the sequence number. The ack. number indicates the next sequence number A is expecting from B.

Page 5 of Transport Slides: ACK = SEQ + number of bytes

Figure 3.37 of textbook: Receiving SEQ=92, 8 bytes of data sends ACK=100

Both seem to indicate True!?!?!

http://web.eecs.utk.edu/~qi/teaching/ece453f06/hw/hw7_sol.htm <- answers Nice!

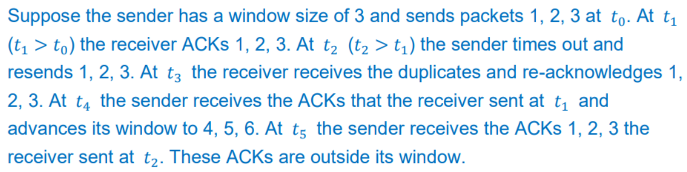
5. With the SR protocol, the receiver will only acknowledge packets within its current window.

http://www.cmlab.csie.ntu.edu.tw/~rod245 has similar (Q3a)

I'm thinking False (after reinterpreting the question)

How can a receiver ever ACK outside of its window?

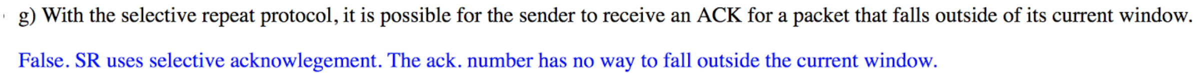
Just because SR can buffer doesn't change this.



Suppose the sender has a window size of 3 and sends packets 1, 2, 3 at t_0. At t_1 ($t_1 > t_0$) the receiver ACKs 1, 2, 3. At t_2 ($t_2 > t_1$) the sender times out and resends 1, 2, 3. At t_3 the receiver receives the duplicates and re-acknowledges 1, 2, 3. At t_4 the sender receives the ACKs that the receiver sent at t_1 and advances its window to 4, 5, 6. At t_5 the sender receives the ACKs 1, 2, 3 the receiver sent at t_2. These ACKs are outside its window.

What's that got to do with the receivers window?

Yea maybe confused between the two



g) With the selective repeat protocol, it is possible for the sender to receive an ACK for a packet that falls outside of its current window.

False. SR uses selective acknowlegement. The ack. number has no way to fall outside the current window.

Good point lol. So the previous block of text was for GBN?

I'm not sure, other sources seem to be saying that SR senders can receive ACKS outside their window.. Will have to confirm but I'm fairly certain RECEIVERS can only ACK packets in their window

Yes for sure! The sender side is just if you can receive an ACK, move the window because of it, then a duplicate ACK comes afterwards that u just discard. So that happens in GBN. But I guess SR wouldn't send a duplicate ACK so this couldn't happen...

6. With GBN, it is possible for the sender to receive an ACK for a packet that falls outside of its current window.

True

(b) Why does TCP include a deviation in its timeout calculation? Why doesn't it just use the Estimated RTT? [4 marks]

If the chosen RTT is too small (ie: we are not accounting for deviation) then there is going to be a lot of unnecessary retransmission. Conversely if the chosen RTT is too large, any required data retransmission will be unnecessarily delayed. Estimated RTT is only a guess at the current average RTT, which means that half the RTTs will be more than this average. By adding 4 times the deviation to the estimated RTT, we allow for RTTs that are above average, but not "too far" above average.

(c) Consider a reliable data transfer protocol that uses only negative acknowledgements (NAK).

i. Suppose the sender sends data only infrequently. Would a NAK-only protocol be preferable to a ACK-only protocol? Why (think about the receiver's view and recovery from a failed transmission)? [3 marks]

ACK-only would be preferable - with a NAK only protocol, the receiver only realises there is an error when it receives the (x+1)th packet i.e. if the receiver gets packet x-1, then misses packet x, it will only know that when it receives x+1. Therefore, if data is sent infrequently, the (x+1)th packet may not come through for a long time, and the receiver won't know it has missed a packet. (A much better/clearer way of saying what I was thinking)

ii. Now suppose the sender has a lot of data to send and the end-to-end connection experiences few losses. In this second case, would a NAK-only protocol be preferable to a protocol that uses ACKs? Why ? [2 marks]

NAK-only would be preferable - as discussed above, a NAK-only protocol receiver will know packet x is lost when packet x+1 is received. If the data rate is higher, and there are less losses, the errors are less likely to occur and are more likely to be corrected quickly. Also, there is much less acknowledgement overhead (as we are not ACKing everything we receive) with a NAK-only protocol if the BER is low.

iii. Given a choice of the ACK-only scheme in TCP and the NAK-only scheme just described which scheme will you use if you protocol needed to work under high data rate and low BER conditions? [2 marks]

Choose the NAK-only protocol - less acknowledgement overhead (since NAKs are only sent out when packets are lost and there is a low BER) and the high data rate allows the lost packets to be recognised and retransmitted earlier.
Note: BER is Bit Error Rate

(d) Slow-start and Congestion avoidance are two phases of TCP's congestion avoidance mechanism.
i. At what point does TCP change from slow-start to congestion avoidance and how is this value determined? [3 marks]
Move from slow-start to congestion avoidance once cwnd (congestion window variable) gets above ssthresh (slow-start threshold). Ssthresh is set at half the cwnd when a time-out occurs.

ii. Why should three duplicate ACKs trigger fast retransmit/fast recovery instead of dropping back to slow start? [2 marks]
3 dup ACKs means information is not being lost to congestion, but is still coming through. Not congestion, so don't need to drop back to lowest rate (start of slow start) to relieve congestion, but should still slow down

Routing and Internet Protocol
Question 3 [Total for Question 2: 28 marks]

(a) You are asked to assign IP addresses to all of the interfaces found in the network identified by A, B, C, D, E, F shown in Figure 2. In your assignment you must make sure that you do not waste any IP addresses given that you only have five hosts and a router. Please clearly show your work to justify the IP address assignment and write down three IP addresses for A, B, and C. [4 marks]
5 hosts + router = 6 IP addresses, and the first and last addresses in a network are the network address and the broadcast address, so we need 8 addresses = 3 bits (2^3=8_
IP is 32 bits long. 3 bits are going to be for host, and 29 for network. So first 3 bytes will all be the same:
X.Y.Z.(host)
First 5 bits of (host) will all be the same (on the same network), and the lower 3 will differ, so IP's will be (assuming those first 5 bits are zero) X.Y.Z.1, X.Y.Z.1, X.Y.Z.2, … X.Y.Z.6

IP X.Y.Z.0 is the network address (host with all bits set to 0)
IP X.Y.Z.7 is the broadcast address (host with all bits set to 1)
All are on network X.Y.Z.0/29

(b) Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching (like the example in lecture where the organisation changed providers) and has the following forwarding table:
Prefix Match Interface

Prefix Match	Interface
1	0
10	1
111	2
otherwise	3

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range. [6 marks]

0) 1xxxxxxx : 11000000 to 11011111 (192 to 223 = 32 addresses)
1) 10xxxxxx : 10000000 to 10111111 (128 to 191 = 64 addresses)
2) 111xxxxx : 11100000 to 11111111 (224 to 255 = 32 addresses)
3) Others : 00000000 to 01111111 (0 to 127 = 128 addresses)

Sanity check: 32+64+32+128=256 addresses, so whole address range has been covered

Use process of elimination, starting with the match giving most info:
111 has to be 11100000 to 11111111
10 has to be 10000000 to 10111111
1 cannot start with 10, so must start with 11, and cannot start with 111, so must end with 110xxxxx
So 1 is 11000000 to 11011111
Otherwise is what is left: everything starting with 0, so 00000000 to 01111111

(c) The Figure 3 shows a three-node network that uses a distance-vector protocol for routing.

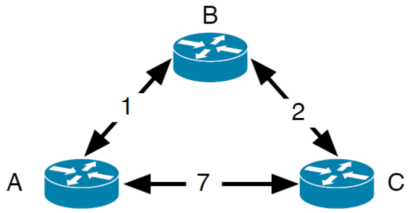


Figure 3: A three-node network showing bi-directional link costs

i. Assuming that 999 represents the value 'infinity', what are the initial routing tables for each node? [3 marks]

Note it's likely this network has poisoned reverse enabled - if A routes to C via B, A will tell B its distance to C is infinity so that B will not route to C via A

Node A	Via B	Via C
To B	1	9
To C	3	7

Node B	Via A	Via C
To A	1	9
To C	999	2

Node C	Via A	Via B
To A	7	3
To B	8	2

ii. Using the distance-vector protocol, calculate and show the final, converged, routing tables for all nodes. You should clearly show all working – marks will be deducted if you fail to show your Working. [12 marks]

Costs of 1 and 2 are not going to improve. Isn't it just going to converge to the best possible distances? Not sure how to show working here - so will leave it for someone else, or get back to it later.

Node A	Via B	Via C
To B	1	?
To C	?	?

Node B	Via A	Via C
To A	1	?
To C	?	2

Node C	Via A	Via B
To A	?	?
To B	?	2

(d) What network characteristic(s) would make distance vector a better choice than link state route determination? [2 marks]

(e) When IPv4 fragments a packet, how does it know which fragments belong to the original packet when it has to reassemble them? [2 marks]
IPv4 datagram has a fragmentation flag, 16-bit identifier, and 13-bit fragmentation offset. The flag shows that the data is fragmented, the identifier gives which original packet the fragment belongs to, and the fragmentation offset shows how each fragment fits into the original packet.

TL;DR: it knows which fragments belong to the original packet by the identifier in the header

Switching and Link Layer Protocols
Question 4 [Total for Question 4: 23 marks]

(a) Consider the LAN in Figure 2
i. How many subnets can you find? Write down the labels (letters) of the hosts in each subnet. [2 marks]
Routers indicate the edge of a subnet. As there is only 1 router, and 2 groups of hosts on each side of the router, there are 2 subnets:
1) Hosts A B C D E F
2) Hosts C G H I J
The router C belongs to both subnets

ii. What is the physical topology of the subnet(s)? [1 mark]
Both subnets have star topology
Was there somewhere different physical topologies were addressed in the course?
Bus, tree, mesh

iii. How does the switch know A is reachable via interface 1, B reachable via interface 2 and so on (Use a simple example to briefly explain your answer)? [3 marks]
Via broadcasting. (Detailed example later - unless someone else wants to do it)

(b) Explain how a packet enters a Multi-Protocol Label Switching (MPLS) network, how it is routed through the network and how it is ultimately delivered. Consider using a simple diagram to help you explain your answer. [6 marks]

It's complicated! Yes, I know how it works - I'm just going to put it in the back of my mind for a while and figure out how to explain it well

(c) Suppose two nodes start to transmit at the same time a packet of length L over a broadcast channel of rate R. Now, denote the propagation delay between the two node as d prop . Will there be a collision if d prop < L/R? Explain why or why not. [2 marks]

L/R=transmission time

d prop < L/R means propagation delay is less than transmission time

Will there be a collision though? I'm not confident to answer this one

(d) Suppose the information portion of a packet contains 10 bytes consisting of the integers 1 through to 10 where each integer is represented using a single 8-bit unsigned binary number representation. Compute the Internet checksum for this data. [4 marks]

```
/* Calculate checksum as per RFC-1071, which RFC-768 defines as:
  "Checksum is the 16-bit one's complement of the one's complement sum ..."
  where "one's complement sum" means that all overflows from 16th bit onwards
  are added back to the checksum
*/
int16_t ComputeChecksum(struct pkt packet) {
   int i;
   int64_t checksum=packet.seqnum+packet.acknum;
   for (i=0; i<20; i+=2)
          checksum+=(packet.payload[i]<<8)+packet.payload[i+1];
   while (checksum>>16)
          checksum=(checksum & 0xFFFF)+(checksum>>16);
   return ~checksum;
}
```

What this is doing is grouping every pair of bytes into 16 bits, then adding that 16 bit number to the running checksum. The overflow of the checksum is then wrapped around and re-added back to the checksum before all bits are inverted, and a 16 bit checksum is given back

Our data is (thank you Mr Python script!)

['0b00000001', '0b00000010', '0b00000011', '0b00000100', '0b00000101', '0b00000110', '0b00000111', '0b00001000', '0b00001001', '0b00001010']

Now, to pair up and add (odd entries to the left, even entries to the right)

0b00000010,00000001 = 513

0b00000100,00000011 = 1027

0b00000110,00000101 = 1541

0b00001000,00000111 = 2055
0b00001010,00001001 = 2569
Sum = 5136
2^16 = 65536. We are less than this, so we don't need to worry about wrap around addition
5163 = 0b0001010000101011
Quick way to invert bits: 65535-5163 = 60372 = 0b1110101111010100

(e) You have received the following data and crc code: 100110001. The generator being used is: 1101 Is there an error in this data or has it been received correctly? Show your work. [5 marks]
0b100110001 = 305
0b1101 = 13
305%13=6. Not 0, so there is an error in the data
Equivalently, 13 does not divide 305 evenly, therefore error
Are we able to do it like this instead of the polynomial division? This would be much faster.

ICMP and Security
Question 5

(a) Explain how traceroute uses ICMP in order to trace the routers along the path to a destination [5 marks]

(b) What are the differences between message confidentiality and message integrity? [2 marks]
Confidentiality: only sender and intended receiver should be able to understand the contents of the transmitted message

Integrity: ensure that the content of communication is not altered by accident

(c) Suppose N people want to communicate with each of N -1 other people using symmetric key encryption. All communication between any two people, i and j, is visible to all other people in this group of N , and no other person in this group should be able to decode their communication. How many keys are required in the system as a whole? Now suppose that public key encryption is used. How many keys are required in this case? [4 marks]

A single and common key is used to encrypt and decrypt the message in symmetric key encryption - 1 key is needed

In public key encryption, one public key is shared in the communication and used to encrypt the message (public key) and one private key is used for decrypting the message.
N group have the communication, so N public key and N private key

That's better :0

N(N-1)/2 for symmetric (every person needs a key for every other person)

For public each person has a public and a private key therefore 2N

[Total for Question 5: 11 marks]

End of exam