COMS3200 Computer Networks I

Introduction

Can you explain the **TCP/IP stack**?

TCP – Transmission Control Protocol (Reliable)

The TCP/IP Stack, or the internet protocol suite, is a set of communication protocols used by the Internet or similar networks.

TCP/IP Stack

- Application (e.g. FTP, SMTP, HTTP)
- Transport (TCP, UDP)
- Network (IP, routing protocols)
- Link (e.g. Ethernet, 802.111 (wifi), PPP)
- Physical (bits 'on the wire')
- Alternate: OSI model adds Presentation and Session layers between Application and Transport
- Data is encapsulated (message, segment, datagram, frame)

Can you compare the **packet switching** and **circuit switching** networks with a number of users?

packet-switching: hosts break application-layer messages into packets.

- forward packets from one router to the next, across links on path from source to destination
- each packet transmitted at full link capacity
- packet switching allows more users to use network!

circuit switching: end-end resources allocated to, reserved for "call" between source & destination.

- dedicated resources: no sharing
- circuit segment idle if not used by call (no sharing)

Circuit switching has fixed number of potential users, packet switching can have more users so long as they aren't all communicating at once.

Packet switching:

- Better for 'bursty data'
- Can share resources
- Simpler (no call setup)
- Congestion is a problem (packet delay and loss)
- Protocols needed for reliable data transfer and congestion control

Can you calculate **various delays**?

Application layer

Can you find and explain detailed information on a given **application message** (e.g., HTTP request/response, DNS request/response, DHCP)?

message formats:

- headers: fields giving info about data
- data: info(payload) being communicated

DHCP: Dynamic Host Configuration Protocol overview:

- host broadcasts "DHCP discover" msg [optional]
- DHCP server responds with "DHCP offer" msg [optional]
- host requests IP address: "DHCP request" msg
- DHCP server sends address: "DHCP ack" msg

Can you explain the request and response of **application messages**?

typical request/reply message exchange:

- client requests info or service
- server responds with data, status code

Transport layer

The **TCP sliding windows** are byte oriented. What does this mean?

It means that the sequence and acknowledgement numbers refer to bytes instead of segments. For example, the value of the ack-field in a segment defines the number of the next byte a party expects to receive.

Can you explain the differences between **UDP and TCP**?

TCP is reliable between sending and receiving process, has flow control (won't overwhelm receiver), congestion control (throttle sender when network overloaded)

- Does not provide latency, minimum throughput guarantees or security
- 'Connection-oriented' – requires setup between client and server processes

UDP is unreliable between sending and receiving process

- Does not provide reliability, flow control, congestion control, throughput guarantee, security, connection setup

Security can be done at app layer using SSL/TLS

Give examples of applications where it is good to use **UDP and TCP** respectively.

UDP is a very simple protocol with minimal overhead. If a process wants to send a small message and does not care much about reliability, it can use UDP. Sending a small message using UDP takes much less interaction between sender and receiver than using TCP. UDP is used in multimedia and multicast applications, such as multiplayer games. If reliability is wanted on the other hand, TCP should be chosen. FTP and Telnet use TCP as transport layer protocol.

How does **TCP support reliable delivery of packets**? Briefly explain three mechanisms.

- SEQ & ACK
  - Messages can be confirmed as arrived safely
  - Resend if not!
- Checksum
  - Check that bits haven't been flipped
- Handshake
  - Check that server/host can facilitate comm.
- Window size
  - Advertise acceptable receive length! Don't overflow!

Can you explain how to guarantee **reliable delivery of application messages**?

- Deals with bit errors using checksums, and ACKs and possibly NAKs to ask for retransmission of erroneous packets
- Adds sequence number to deal with duplicates
- Deals with loss by waiting for a 'reasonable' time for the ACK, retransmitting if no ACK

Can you find information (port number, direction, application protocol etc) given **transport layer protocol header** (TCP or UDP) dump?

TCP Segment structure

- 16-bit source port #
- 16-bit dest port #
- 32-bit sequence #
- 32-bit ack #
- 16-bit flagset: header length, unused padding, URG, ACK valid, PSH, RST, SYN, FIN
- 16-bit receive window
- 16-bit internet checksum
- 16-bit URG data pointer
- Variable length options
- Variable length data

Can you calculate **checksum** and verify it?

UDP Checksum

- Goal: detect 'errors' (e.g., flipped bits) in transmitted segment
- Sender treats segment contents (inc. header fields) as sequence of 16-bit numbers
- Sender **calculates checksum by addition (one's complement sum) of segment contents**
- Sender puts the value into the checksum field
- Receiver computes checksum of segment
- Checks if computed checksum equals the checksum field value
- If not, errors detected
- If yes, no error detected (though errors could still have occurred)

Can you explain the differences between **flow control and congestion control**?

Flow control makes sure that the receive buffers aren't getting full, Congestion Control makes sure that the connection isn't getting full.

Flow control implements throttling the speed to prevent application buffer overflow

Congestion Control throttles speed to prevent the loss of packets in a congested network

What is the main **mechanism** used to implement TCP **flow control**?

Flow control mechanism is the receive window that the sender maintains for the receiver.

Describe one of the **mechanisms** that is used to implement TCP **congestion control**.

Congestion control mechanism is the congested packet telling not to send any more.

What is the difference between **connectionless** and **connection-oriented** transport layer protocols?

Connectionless: UDP: just send packets without init a connection, which is fast

Connection-oriented: TCP: handshake, agree on terms, reliable, slow

Mechanism or Protocol for **connectionless** transport?

UDP

Mechanism or Protocol for **connection-oriented** transport?

TCP

What is the difference between **confidentially** and **authentication** in secure data transport?

Confidentially: cannot determine who sent / who will recv / what data details.

Authentication: producing a certificate to allow for confidential transport -> is the person who they say they are.

Mechanism for **Confidentially**?

TLS/SSL

Mechanism for **Authentication**?

TLS/SSL (RSA checks)

Network layer

Can you explain the **packet scheduling** in a router?

Can you explain the **packet fragmentation** and how it works?

Can you explain the differences between the **operation of distance vector and link-state routing algorithms**?

Link state routing protocols	Distance vector routing protocols
All routers have complete topology, link cost information	Routers know only physically connected neighbours and the link cost to them
The link costs are broadcast to all the routers in the network from a single controller	Requires an iterative process of computation, exchange of information with neighbours

In Distance-Vector, each node tells its neighbours everything it knows about the network. In Link-State, each router broadcasts the state of its own links to the entire network. Leaving each router to build up a graph of the network.

Describe one advantage of **link-state routing**.

LS has the advantage that it updates router's graphs quicker across the network because it broadcasts changes.

Describe one advantage of **distance-vector routing**.

DV advantage is that it uses less memory and CPU by not having to build up graphs, only populate a routing table

Given a network graph, can you make a table that contain the **minimum-cost routes** from a source node to all other nodes using **Dijkstra's algorithm and Distance Vector algorithm (Bellman-Ford algorithm)**, respectively?

Can you explain **NAT**?

NAT: network address translation

motivation: local network uses just one IP address as far as outside world is concerned:

- range of addresses not needed from ISP: just one IP address for all devices
- can change addresses of devices in local network without notifying outside world
- can change ISP without changing addresses of devices in local network
- devices inside local net not explicitly addressable, visible by outside world (a security plus)

Can you explain the difference between **IPv4 and IPv6**?

no fragmentation/reassembly allowed at intermediate routers:

- These operations can be performed only the source and destination.

ICMPv6:

- new version of ICMP
- additional message types, e.g. "Packet Too Big"
- multicast group management functions

Header checksum:

- removed entirely to reduce processing time at each hop
    - Since the IPv4 header contains a TTL field, the IPv4 header checksum needed to be recomputed at every router, a costly operation

options:

- No longer a part of the standard IP header.
- cf. IPv4 option: used for network testing, debugging, security, and more. This field is usually empty.

Link layer

Can you explain how **odd/even parity bit** works? Can you find parity bit given binary digits? Can you explain the **2-D parity and its limitation**?

A parity bit is a check bit, which is added to a block of data for error detection purposes. It is used to validate the integrity of the data. The value of the parity bit is assigned either 0 or 1 that makes the number of 1s in the message block either even or odd depending upon the type of parity. Parity check is suitable for single bit error detection only.

The two types of parity checking are

Even Parity – Here the total number of bits in the message is made even.

Odd Parity – Here the total number of bits in the message is made odd.

Two-Dimensional Parity can detect as well as correct one or more-bit errors. If a one or more-bit error takes place then the receiver will receive the message with the changed parity bit. It indicates that some error has taken place which means the error is detected.

DRAWBACKS

In some cases, an only odd number of bit errors can be detected and corrected but even number of errors can only be detected but not corrected.

In some cases, this method is not able to detect even no bit error.

Can you calculate/show how **CRC is used to detect error(s)**?

In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.

At the destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and is therefore accepted.

A remainder indicates that the data unit has been damaged in transit and therefore must be rejected.

Can you explain why **forward error correction (FEC)** is used? Can you show how **FEC is used to detect and/or correct error(s)**?

Forward Error Correction (FEC) is a technique used to minimize errors in data transmission over communication channels. In real-time multimedia transmission, re-transmission of corrupted and lost packets is not useful because it creates an unacceptable delay in reproducing: one needs to wait until the lost or corrupted packet is resent. Thus, there must be some technique which could correct the error or reproduce the packet immediately and give the receiver the ability to correct errors without needing a reverse channel to request re-transmission of data.

- **Reduce retransmission on error**
- **Increase reliability of wireless systems**

Using Hamming Distance:

For error correction, the minimum hamming distance required to correct t errors is:

$dmin = 2t + 1$

For example, if 20 errors are to be corrected then the minimum hamming distance must be 2*20+1= 41 bits. This means, lots of redundant bits need to be sent with the data. This technique is very rarely used as we have large amount of data to be sent over the networks, and such a high redundancy cannot be afforded most of the time.

Can you explain **MAC address**? Can you find relevant information given MAC address?

MAC Addresses are unique 48-bits hardware number of a computer, which is embedded into network card (known as Network Interface Card) during the time of manufacturing. MAC Address is also known as Physical Address of a network device. In IEEE 802 standard, Data Link Layer is divided into two sublayers

- Logical Link Control (LLC) Sublayer
- Media Access Control (MAC) Sublayer

MAC address is used by Media Access Control (MAC) sublayer of Data-Link Layer. MAC Address is worldwide unique, since millions of network devices exists, and we need to uniquely identify each. Function: used 'locally" to get frame from one interface to another physically connected interface (same network, in IP-addressing sense)
The first six digits (called the "prefix") represent the adapter's manufacturer, while the last six digits represent the unique identification number for that specific adapter. The MAC address contains no information about which network a device is connected to.

Can you explain how **MAC addresses/IP addresses** are used in a **LAN** and between different LANs?

MAC addresses are the low-level basics that make your local ethernet based network work. Local means that the network devices are either directly connected through a cable or by WiFi or over a network hub or network switch.

Network cards each have a unique MAC address. Packets that are sent on the ethernet are always coming from a MAC address and sent to a MAC address. If a network adapter is receiving a packet, it is comparing the packet's destination MAC address to the adapter's own MAC address. If the addresses match, the packet is processed, otherwise it is discarded.

How do IP addresses and MAC addresses work together?

IP is a protocol that is used on a layer above ethernet. Another protocol for example would be IPX. IP allows connecting of different local networks and thus forming a corporate network or the global internet.

When your computer wants to send a packet to some IP address x.x.x.x, then the first check is if the destination address is in the same IP network as the computer itself. If x.x.x.x is in the same network, then the destination IP can be reached directly, otherwise the packet needs to be sent to the configured router.

Up to now things seem to have gotten worse, because now we have two IP addresses: one is the original IP packet's target address, the other is the IP of the device to which we should send the packet (the next hop, either the final destination or the router).

Since ethernet uses MAC addresses, the sender needs to get the MAC address of the next hop. There is a special protocol ARP (address resolution protocol) that is used for that. Once the sender has retrieved the MAC address of the next hop, he writes that target MAC address into the packet and sends the packet.

Can you explain the **MAC protocols** and their differences (e.g., CSMA)?

channel partitioning, by time, frequency, or code

- Time Division, Frequency Division

random access (dynamic),

- ALOHA, S-ALOHA, CSMA, CSMA/CD
- carrier sensing: easy in some technologies (wire), hard in others (wireless)
- CSMA/CD used in Ethernet
- CSMA/CA used in 802.11

taking turns

- polling from central site, token passing
- Bluetooth, FDDI, token ring

Describe the key difference between **CSMA/CD and CSMA/CA media access protocols**.

Key	CSMA/CA	CSMA/CD
Effectiveness	CSMA/CA is effective before a collision.	CSMA/CD is effective after a collision.
Network Type	CSMA/CA is generally used in wireless networks.	CSMA/CD is generally used in wired networks.
Recovery Time	CSMA/CA minimizes the risk of collision.	CSMA/CD reduces recovery time.
Conflict Management	CSMA/CA initially transmits the intent to send the data, once an acknowledgment is received, the sender sends the data.	CSMA/CD resends the data frame in case a conflict occurs during transmission.
IEEE Standards	CSMA/CA is part of the IEEE 802.11 standard.	CSMA/CD is part of the IEEE 802.3 standard.
Efficiency	CSMA/CA is similar in efficiency as CSMA.	CSMA/CD is more efficient than CSMA.

Give an example of a link layer protocol that uses **CSMA/CD (carrier sense multiple access with collision detection)**.

802.3 (Ethernet)

Give an example of a link layer protocol that uses **CSMA/CA (carrier sense multiple access with collision avoidance)**.

802.11 (wireless) since we cannot be fully sure of collisions

Can you explain how **VLAN** is working?

switch(es) supporting VLAN capabilities can be configured to define multiple virtual LANS over single physical LAN infrastructure.

Can you explain how **MPLS** is working?

is a routing technique in telecommunications networks that directs data from one node to the next based on short path labels rather than long network addresses, thus avoiding complex lookups in a routing table and speeding traffic flows.

What is difference between **collision detection and collision avoidance** in link layer protocols?

CD: detect collision and stop transmission algorithmically (CSMA/CD)

CA: check channel clear before transmission

Mechanism for **Collision Detection**?

CSMA/CD

Mechanism for **Collision Avoidance**?

CSMA/CA

Relevant questions in the past final exam

- 2020 final exam
  - Q4, Q5, Q12,
- 2019 final exam
  - Q4, Q6, Q7, Q9
- The past exams are available at:
  - 2018 final exam: Q2(a), Q3, Q5
  - 2017 final exam: Q1(a), Q2, Q4(e)&(f)

Minus: 15 - (40 - (59/100 * 20 + 80.7647/100 * 20)) = 2.95294

Minus In final: (15 - (40 - (59/100 * 20 + 80.7647/100 * 20))) / 60 * 100 = 4.92156666667

Mock - COMS3200 Semester One Final Examination 2021

QUESTION 1 {Transport layer} The following is a dump (contents) of a **UDP header** in hexadecimal format.

E555 *0015* 0040 *3A6B*

(a) What is the **source port** number in decimal form? Show your working in the working sheet. $E555_{16}$ = 58709

(b) What is the **destination port** number? Show your working in the working sheet. 0015_{16} = 21

(c) What the **total length of the user datagram** in bits? Show your working in the working sheet.

0040_{16} = 64 bytes = 64*8 = 512 bits

(d) What is the **length of the data** in bytes? Show your working in the working sheet.

Since the header is 8 bytes the data length is 64 − 8 = 56 bytes.

(e) What is the **checksum value** in hexadecimal form? 3A6B

(f) Is this packet directed from a client to a server or vice versa? A client to a server

(g) What is the **application-layer protocol**? DNS

The following is a dump (contents) of a **TCP header** in hexadecimal format.

a b c d e f g

E293 0017 00000001 00000000 5 002 07FF …

(a) What is the **source port** number? $(E293)_{16}$ = 58,003

(b) What is the **destination port** number? $(0017)_{16}$ = 23

(c) What is the **sequence number**? $(00000001)_{16}$ = 1

(d) What is the **acknowledgment number**? $(00000000)_{16}$ = 0

(e) What is the **length** of the header? The HLEN = 5. The header is 5 x 4 (scaling factor) = 20 bytes long

(f) What is the **type of the segment**?

$(002)_{16}$ = $(000000000010)_2$ the right most 6 bits are 000010, which means only the SYN bit is set. This is the SYN segment used for connection establishment.

g. What is the **window size**? $(07FF)_{16}$ or 2047 in decimal. The window size is 2047 bytes.

QUESTION 3 {Quantitative Comparison of Packet Switching and Circuit Switching}

Consider the two scenarios below: a circuit-switching scenario in which **Ncs** users, each requiring a bandwidth of **10 Mbps**, must share a link of capacity **50 Mbps**. A packet-switching scenario with **Nps** users sharing a **50 Mbps** link, where each user again requires **10 Mbps** when transmitting, but only needs to transmit 30 percent of the time.

(a) When **circuit switching** is used, what is the maximum number of circuit-switched users that can be supported?

(Circuit users can't share bandwidth)

For each of the user is allocated 10mbps bandwidth and given link capacity is 50mbps. 50/10 = 5

For the remainder of this problem, suppose packet switching is used.

(b) Suppose there are 10 packet-switching users (i.e., Nps = 10). What is the probability that a given (specific) user is transmitting, and the remaining users are not transmitting? **(How much bandwidth does each user need? Is this less than the total bandwidth?)**

The probability that a specific user is transmitting, ρ, is the percent of the time it is transmitting, i.e. 0.3.

The probability that a specific user is not busy is (1 - ρ).

The probability that the Nps - 1 users are not transmitting is $(1-\rho)^{Nps-1}$.

Thus, the probability that one user is transmitting, and the other users are not transmitting is, $\rho^1(1-\rho)^{10-1}$=0.3x0.7^9= 0.0121060821

(c) What is the probability that one user (any one among the 10 users) is transmitting, and the remaining users are not transmitting?

The probability that exactly one (any one) of the Nps users is busy is Nps times the probability that a given specific user is transmitting and the remaining users are not transmitting (our answer to (c) above), since the one transmitting user can be any one of the Nps users.

Therefore, the fraction of the link used when a user is using the link (and the remaining users aren't transmitting) is, $Nps * \rho^1 * (1-\rho)^{Nps-1}$=10x0.3^1x0.7^9= 0.121060821

(d) What is the probability that any 6 users (of the total 10 users) are transmitting and the remaining users are not transmitting? (C (n, k) = n! / [(n-k)! k!])

Using the formula $\rho^n(1-\rho)^{Nps-n}$ we can find the probability that n specific users are transmitting and Nps - n users are not.

To find the probability that any n (6 in this case) out of the 10 possible users, are transmitting is choose (10, n) $*$ $\rho^n(1-\rho)^{Nps-n}$

c (10,6) $*$ $\rho^6(1-\rho)^{10-6}$= 10! / ((10-6)! x 6!) x 0.3^6x0.7^4= 0.036756909

(e) What is the probability that more than 4 users are transmitting?

The probability that more than 4 users are transmitting is

$$\sum_{n=5}^{10}\binom{10}{n}\rho^n\rho^{Nps-n} = \sum_{n=5}^{10} 10!/\left((10-n)! * n!\right) * 0.3^n * 0.7^{10-n} = 0.15026833$$

QUESTION 4 {HTTP GET}

Suppose that a server receives the following **HTTP GET** message from a client browser:

GET /main/test1.html HTTP/1.1 \r\n
Host: www.uq2.edu.au \r\n
User-agent: Firefox/3.6.12 \r\n
Accept: text/html, application/xhtml+xml \r\n
Accept-language: en, fr; q = 0.8, en-nz; q =0.5 \r\n
Accept-Encoding: gzip, deflat \r\n
Connection: close \r\n
\r\n

(a) What is the **name of the file** that is being retrieved in this GET message? Please use file name only.

test1.html (location: /main/)

(b) What **version of HTTP protocol** does the browser use? Please use only number.

1.1

(c) What is the **language preference** of the browser user mostly preferring to use?

English

(d) Does the browser want to have **persistent connections**? Answer Yes, No or Undecidable.

No

(e) Assume that the browser has received "internal server error" from the web server. What is response code for it? Please provide the numerical value.

500 (https://dynomapper.com/blog/254-the-6-types-of-http-status-codes-explained)

QUESTION 5 Consider a scenario that TCP a sender and receiver communicate over a connection in which the sender-to-receiver segments may be lost in Figure 1. The TCP sender sends initial window of four segments at t=1,2,3,4, respectively. Suppose the initial value of the sender-to-receiver sequence number is 116 and the **first four segments each contain 502 bytes**. The **delay between the sender and the receiver is 7-time units**, and so the first segment arrives at the receiver at t=8. As shown in the figure, two of the four segment(s) are lost between the sender and the receiver.

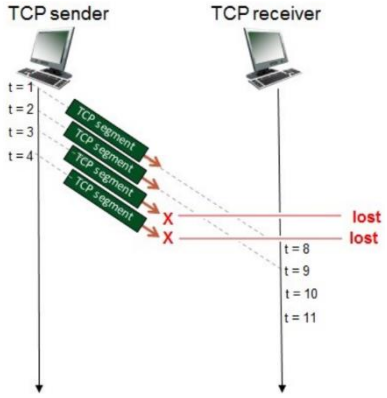


Figure 1. TCP sequence and ACK numbers with segment loss

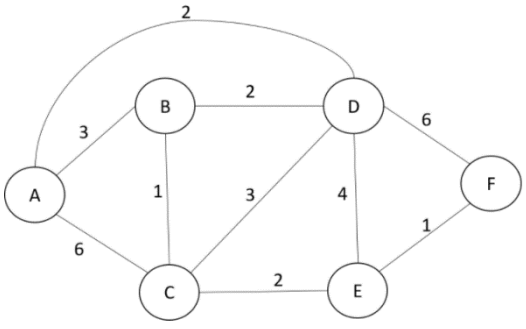
Answer the following questions (2 marks each) in the table below:

- Fill in the sequence numbers associated with the segments sent by the sender.
- Fill in the time the segments were received.
- Fill in the acknowledgment field of each receiver-to-sender acknowledgment and give a brief explanation as to why that particular acknowledgment number value is being used.

Sender-to-Receiver	Time segment sent	Sender-to-receiver segment sequence number field value	Time segment received, and ACK segment sent	Receiver-to-sender ACK field value
Segment 1	1	116	8	618
Segment 2	2	618	9	1120
Segment 3	3	1120	-	No ACK is sent, since this segment was lost
Segment 4	4	1622	-	No ACK is sent, since this segment was lost

QUESTION 6

Figure 2. An example network 1



(a) (10 marks total) Apply the **Dijkstra algorithm** on the example network 1 in Figure 2 to find the minimum-cost routes from **station A** to all other stations. Please make a table for the final value. S is the set of stations whose least-cost path is known; D(v) is the current cost of path from source (i.e., station 1) to station v; p(v) is the predecessor station along path from source to v, that is next to v.

Please use "inf" to specify an infinite cost and "-" to specify no predecessor respectively.

The following table has not been completed filled on purpose. 'X' is used to indicate that that cell will be filled with information.

Step	S	D(B), p(B)	D(C), p(C)	D(D), p(D)	D(E), p(E)	D(F), p(F)
0	A	3, A	6, A	2, A	inf,-	inf,-
1	X	3, A	5, D	X	X	X
2	X	X	4, B	X	6, D	X
3	X	X	X	X	X	X
4	X	X	X	X	X	7, E
5	ABCDEF	X	X	X	X	X

(b) Apply the **Bellman-Ford algorithm** on the example network 1 given in Figure 2 to find the minimum-cost routes from **station B** to all other stations.

Please use "inf" to specify an infinite cost and "-" to specify no next hop respectively.

The following table has not been completed filled on purpose. 'X' is used to indicate that that cell will be filled with information.

Dest.	Hop 1		Hop 2		Hop 3		Hop 4		Hop 5	
	cost	hop	cost	hop	cost	hop	cost	hop	cost	hop
A	3	A	X	X	X	X	X	X	X	X
C	1	C	X	X	X	X	X	X	X	X
D	2	D	X	X	X	X	X	X	X	X
E	inf	-	3	C	X	X	X	X	X	X
F	inf	-	8	X	4	X	X	X	X	X

QUESTION 789 [IP/subnet] Please choose which **class** the following IPv4 Address belongs to.

192.168.1.2 - Class C

2.2.2.2 - Class A

223.265.200.1 - invalid IP address

QUESTION 10 Suppose an ISP owns the block of addresses of the form for IP address 224.1.1.1/24. Suppose it wants to create four subnets from this block, with each block having the same number of IP addresses. What is the **total number of usable hosts**? Show your works on the working sheet.

224.1.1.1: *11100000.00000001.00000001*.00000001

The first 24 bits are fixed, and last bit is also fixed as 1, so only 2^7 total number of available hosts and 2 is 2 reserved addresses. the **total number of usable hosts is therefore 2^7(128) – 2 = 126**

QUESTION 11 Your company wants to utilize the private **class C** IP Address of 192.164.1.0. You are tasked with subnetting the address to get the most networks with at least 30 hosts per subnet. How many networks will be created after you complete subnetting?

https://www.calculator.net/ip-subnet-calculator.html?cclass=c&csubnet=27&cip=192.164.1.0&ctype=ipv4&printit=0&x=62&y=20

192.164.1.0 = 11*000000.10100100.00000001*.00000000

Bits needs for 30 hosts = 5 = 2^5 = 32-2=30 possible hosts.

Bits left for subnets = 3 = 2^3 = 8 possible subnets.

QUESTION 12 Your company wants to utilize the private **class C** IP Address of 192.164.1.0. You are tasked with subnetting the address to get the most networks with at least 30 hosts per subnet.

What is the first usable IP Address in the 1st Network range?

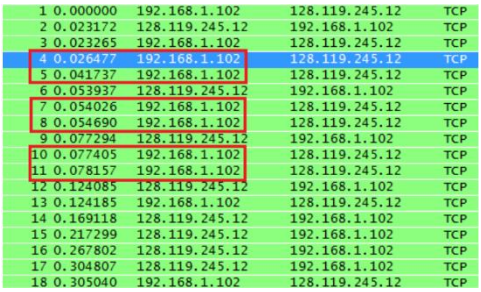
Our second step will be to calculate the new subnet mask, our previous subnet mask was 255.255.255.0 or 11111111.11111111.11111111.00000000 in binary. Since we have borrowed 3 bits from the host portion our new subnet mask will be 11111111.11111111.11111111.11100000 which is 255.255.255.224 when converted to decimal notation.

First network .0 to .31 first useable .1

Second network .32 to .63 first useable .33

192.164.1.1

QUESTION 13 Figure 3 shows a Wireshark screen shot that analyses the trace of a TCP segment sent and received directly by uploading a 150KB text file from a computer to a remote server.
The six segments sent by the client (192.168.1.102) to the server (128.119.245.12) are No. 4, 5, 7, 8, 10, and 11 (these are marked in a red highlighted box).



1	0.000000	192.168.1.102	128.119.245.12	TCP
2	0.023172	128.119.245.12	192.168.1.102	TCP
3	0.023265	192.168.1.102	128.119.245.12	TCP
4	0.026477	192.168.1.102	128.119.245.12	TCP
5	0.041737	192.168.1.102	128.119.245.12	TCP
6	0.053937	128.119.245.12	192.168.1.102	TCP
7	0.054026	192.168.1.102	128.119.245.12	TCP
8	0.054690	192.168.1.102	128.119.245.12	TCP
9	0.077294	128.119.245.12	192.168.1.102	TCP
10	0.077405	192.168.1.102	128.119.245.12	TCP
11	0.078157	192.168.1.102	128.119.245.12	TCP
12	0.124085	128.119.245.12	192.168.1.102	TCP
13	0.124185	192.168.1.102	128.119.245.12	TCP
14	0.169118	128.119.245.12	192.168.1.102	TCP
15	0.217299	128.119.245.12	192.168.1.102	TCP
16	0.267802	128.119.245.12	192.168.1.102	TCP
17	0.304807	128.119.245.12	192.168.1.102	TCP
18	0.305040	192.168.1.102	128.119.245.12	TCP

Figure 3. a Wireshark screentshot

(a) Considering the difference between when each TCP segment was transmitted and when its acknowledgement was received, what is the **Round-Trip Time (RTT)** value of the second of the six segments?

0.077294(no.9) – 0.041737(no.5) = 0.035557

(b) What is the EstimatedRTT of the second segment after receiving the ACK? Assume that the EstimatedRTT equal to the measured the **Round-Trip Time (RTT)** for the first segment.

EstimatedRTT = $(1 - \alpha) * \text{EstimatedRTT} + \alpha * \text{SampleRTT}$ with α is 0.125

= 0.875 x (0.053937 – 0.026477) + 0.125 x 0.035557

= 0.028472125

QUESTION 14 {Error detection and correction}

(a) What is the Internet checksum value for these two 16-bit words (use one's compliment addition). Answer it without space between binary digits.

1000 0110 0101 1110
1010 1100 0110 1000

Step 1. Add the two numbers

= (+1) 0011 0010 1100 0110

Step 2. Carry over the overflow

= 0011 0010 1100 0111

Step 3. Compute one's complement

= 1100 1101 0011 1000

(b) What is the parity bit for 0100111 when the **odd** one-dimensional parity scheme?

1

(c) What is the parity bit of 1100110 when the **even** one-dimensional parity scheme?

0

(d) The two-dimensional (2D) **even** parity scheme is used for the following data:

01110 01010 01001 11001

Suppose that 5 bits are used in one row for the 2D parity. What are the first four parity bits in the column only?

011101

010100

010010

110011

101000

For *k*=2 and *n*=4, we can make the following assignment.

No	Data Block	Codeword
1	00	0001
2	01	0011
3	10	1000
4	11	1110

(e) What is the **minimum Hamming distance** when a codeword block is received with the bit pattern 1001?

1

(f) Can the error be detected (Yes or no) when the received codeword is 1101? Choose one: Yes, No or Undecidable.

Yes, the error can be detected since 1101 is not a valid codeword.

To guarantee the detection of up to s errors in all cases, the minimum Hamming distance in a block code must be dmin = s + 1.

(g) Can the error be corrected when the received codeword is 1001? Choose one: Yes, No or Undecidable.

No. If flip one bit, it could be case 1 or 3.

To guarantee correction of up to t errors in all cases, the minimum Hamming distance in a block code must be dmin = 2t + 1.

2019 Q1 [HTTP GET]

GET /kurose_ross/interactive/quotation4.htm HTTP/1.1

Host: www.univ1.edu.au

Accept: text/plain, text/html, image/jpeg, image/gif, audio/mp4, audio/vnf.wave, video/mp4, video/mpeg, application/, */**

Accept-Language: en-us, en-gb;q=0.5, en;q=0.4, fr, fr-ch, zh, fi

If-Modified-Since: Thu, 25 April 2019 15:20:19 -0700

User Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_3) AppleWebKit/534.53.11 (KHTML, like Gecko) Version/5.1.3 Safari/534.53.10

Q1-1. What is the name of the file that is being retrieved in this GET message? quotation4.htm

Q1-2. What formats of text, images, audio, and video does the client browser prefer to receive?

Plaintext, HTML text, jpeg, gif, mp4 of video and audio, vnf.wave, mpeg, any application, anything of any type

Q1-3. Does the browser sending the HTTP message prefer Swiss French over traditional French? Explain

No. both have equal q values (none supplied) default =1. So, both are accepted

Q1-4. Does the client already have a (possibly out-of-date) copy of the requested file? Explain. If so, approximately how long ago did the client receive the file, assuming the GET request has just been issued?

Yes, presence of if-modified-since indicates that file already present. So, received on 25th April 2019.

Question 2. [HTTP Response]

HTTP/1.1 404 Not Found

Date: Mon, 24 Sep 2018 22:23:34 +0000

Server: Apache/2.2.3 (CentOS)

Content-Length: 74396

Keep-Alive: timeout=39, max=82

Connection: Keep-alive

Content-type: image/html

Q2-1. Was the server able to send the document successfully? Explain.

No. 404 error indicates of resource requested not existing without the server.

Q2-2. When was the file last modified on the server? Never, it doesn't exist / has no memory.

Q2-3. What is the type of file being sent by the server in response? Image/html

Q2-4. What is the default mode of connection for HTTP protocol? Is the connection in the reply persistent or non-persistent? Explain. 1.1/persistence is default. Connection is persistent we have keep-alive message, which means multiple objects sent over 1 TCP connections/handshake.

Question 5. [IP/subnet] Suppose an ISP (internet service provider) owns the block of addresses of the form 101.101.128/17. Suppose it wants to create **four subnets** from this block, with each block having the same number of IP addresses.

Q5-1. What is the **maximum number of hosts** can be connected to each subnet? Show your works.

101. 101. 128/17

*01100001.01100101.1*0000000.00000000

So, 15 bits for hosts for 4 groups equally sized

2^15 IP = 32768 machines

/4 = 8192 machines per group – 2 (broadcast, gateway), so, 8190 hosts on each subnet

Q5-2. What are the prefixes (of the form a.b.c.d/x) for the four subnets?

101.101.128.0/19

101.101.160.0/19

101.101.192.0/19

101.101.224.0/19

Question 6. [Checksum]

Q6-1. If the Internet **checksum** method is adopted, what message will be sent if data is 5AD3EE35? If the message received is 59D4 EF35 B6F6, will the message be accepted? (Show your workings.)

Data = 5AD3 EE35 (16 bits 1s comp sum) 00014908 -> 4909 flip: B6F6

Sent: 5AD3 EE35 B6F6

Recv: 59D4 EF35 B6F6

Not same. Not accepted

Question 9. [MAC address] The following is an example MAC address. 00: A0:C9:14:C8:29

Q9-1. Write down the part in hexadecimal indicating the adapter's manufacturer.

00: A0:C9

Q9-2. What protocol is used to find an IP address given a MAC address of a device?

ARP

2017 Q2

The following table describes the **purpose of different networking protocols**. For each of these protocols, give the acronym (abbreviated name) of the relevant protocol, and the relevant layer of the Internet protocol stack. If more than once correct answer is possible, then any correct answer will be accepted.

Purpose of the Protocol	**Name**	**Layer**
To request and receive web pages from a server	HTTP	Application
Convey network management control and information messages	SNMP	Application
Send email messages to mail server	SMTP	Application
Download email messages from a mail server	IMAP/POP3	Application
Used by hosts and routers to communicate network-level information	ICMP	Network
Convert a hostname to an IP address	DNS	Application
get the MAC address corresponding to an ID -> next to forward to!	ARP	Link
7. Sending intra-AS link-state routing messages	OSPF/IS-IS	Link
8. Setting up multimedia data stream connections	TCP	Transport
9. Providing connectivity between hosts and access points in WiFi networks	802.11?	Link/Physical?
An enhanced version of connection-oriented stream transport which adds security	QUIC	Transport
Communication security. Privacy for data.	SSL	Application
Exchange routing info between AS	BGP	Application
Ethernet – physical conn between hubs, switches & routers	IEEE 802.3	Physical