Remember & Understand

1. Mention the encapsulation process in sending a web page request to retrieve home page in web server aie2022network.edu with IP address 18.108.27.9. Mention your answers for b and c in table format specified. (10 marks total)

a) You can mention the major roles and responsibilities of each layer. (2 marks Remember)

b) Specify the benefits of using layered model on the internet. (2 marks Understand)

c) Mention the name of the message and protocol used for each of the layers. (2 marks remember)

d) Apply the major encapsulation information for the given scenario. You can assume the addresses in client side. (4 marks, apply)

Layer	Protocol	Message name	Encapsulated Major Information

Ans:

b) benefits of layered approach (any of these) – 2 marks

- **Assist in protocol design** because protocols that operate at a specific layer have defined information that they act upon and a defined interface to the layers above and below.
- **Foster competition** because products from different vendors can work together.
- Prevent technology or capability changes in one layer from affecting other layers above and below. Provide a common language to describe networking functions and capabilities.

Layer	Protocol	Message name	Encapsulated Major Information
Application	HTTP	Data	http GET request. host: www.aie2022network.edu
Transport	TCP	Segments	Source Port: 1400 Destination Port: 80
Internet	IP	Packet	Source IP: 192.168.1.5 Destination IP: 18.108.27.9
Data Link	Ethernet	Frame	Source MAC: Requesting client's MAC (00-ac-bd-ef-12-23) Destination MAC: MAC of the default gateway of requesting client

2. List the different application-level protocols and transport-level protocols involved in each of these activities, with proper justification for your answer. Ensure that you expand your abbreviation answers. (4 marks, Apply)

M1: Sending an email from a mail client to the mail server.
M2: Downloading an email from the mailbox server to a mail client.
M3: Opening and viewing emails using a web browser.
M4: Finding the IP address of the given URL or domain name.

Hint: M1 – SMTP, TCP M2 – POP3, TLS or SSL over TCP, M3 – HTTP over TCP, M4 – DNS, UDP

M1: Sending an email from a mail client to the mail server.

Application-level protocol: SMTP (Simple Mail Transfer Protocol)

SMTP is responsible for sending emails. When you compose and send an email from your mail client, it uses SMTP to communicate with the mail server.

Transport-level protocol: TCP (Transmission Control Protocol)

SMTP relies on a reliable transport protocol for the transmission of email messages. TCP ensures that the data is delivered accurately and in the correct order.

M2: Downloading an email from the mailbox server to a mail client.

Application-level protocol: POP3 (Post Office Protocol version 3) or IMAP (Internet Message Access Protocol)

POP3 and IMAP are used for retrieving emails from a mail server to a mail client. POP3 is more traditional and downloads emails to the client, whereas IMAP allows users to view and manipulate emails on the server.

Transport-level protocol: TCP (Transmission Control Protocol)

Similar to sending emails, the download process requires a reliable transport protocol to ensure the accurate and ordered delivery of email data.

M3: Opening and viewing emails using a web browser.

Application-level protocol: HTTP (Hypertext Transfer Protocol) or HTTPS (HTTP Secure)

When you open and view emails through a web browser, you're likely using a webmail service. Webmail interfaces typically use HTTP or HTTPS for communication between the web browser and the mail server.

Transport-level protocol: TCP (Transmission Control Protocol)

As with most web-related activities, TCP provides a reliable connection for the transfer of data between the web browser and the server.

M4: Finding the IP address of the given URL or domain name.

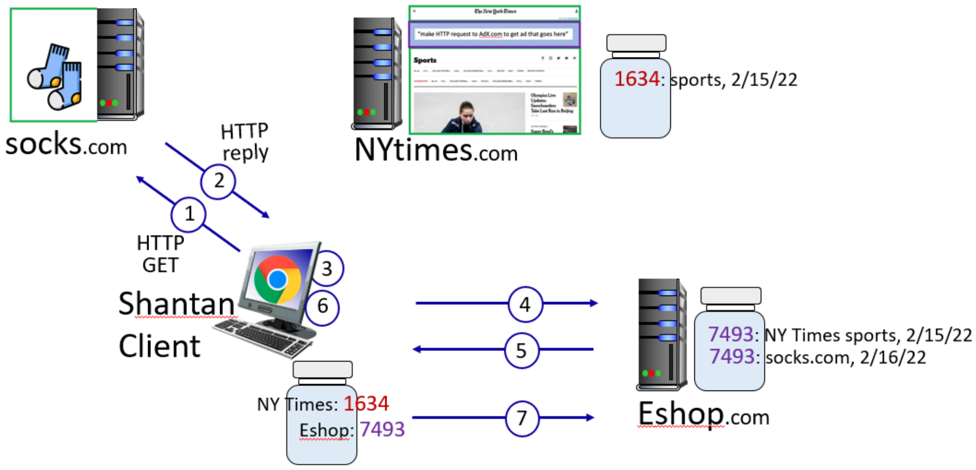
Application-level protocol: DNS (Domain Name System)

DNS resolves human-readable domain names to IP addresses. When you input a URL into a web browser, the browser uses DNS to find the corresponding IP address.

Transport-level protocol: UDP (User Datagram Protocol)

DNS typically uses UDP for its communication due to its lightweight and connectionless nature. Speed is crucial for DNS resolution, and UDP is well-suited for this purpose.

3. Shantan always prefers doing online shopping, as he can add items to her shopping cart and do the purchase later as per his convenience. He is accessing Eshop.com in his browser for the first time is specified in activity 3 and revisiting the same website later is mentioned in activity 6.
   a) Give the HTTP message types, and key parameters used in figure with labels 4, 5 and 7 to accomplish this. (3 marks, Analyze)
   b) Evaluate how is it possible for the Website **Eshop**.com which is an HTTP-based stateless server to be capable of maintaining his shopping history and his sports interest? (3 marks Evaluate)



Ans:

a) 4 = usual HTTP GET request to Eshop.com, 5 = usual HTTP response to Shantan client browser with parameter set-cookie: 7493, 7=usual HTTP Get request to Eshop.com with cookie: 7493

b) Cookies in the client side keep track of the cookies for different website based on external allowance of usage of cookies by the user. Cookies database is maintained in webserver which will mention the cookie: 7493 to keep track of the user and its website usage behavior like sports website is saved. Through this the stateless web server is able to retrieve the usage interest and website usage behavior.

4. Consider a packet of length *L* which begins at end system A and travels over three links to a destination end system. These three links are connected by two routers. Let *di, si,* and *Ri* denote the length, propagation speed, and the transmission rate of link *i,* for *i* = 1, 2, 3. The router delays each packet by *dproc.*
   a) Create a network topology with nodes and links as notified in the scenario. (2 marks, create)
   b) Assuming no queuing delays, in terms of *di, si, Ri,* (*i* = 1,2,3), and *L,* what is the total end-to-end delay for the packet? (3 marks, evaluate)
   c) Suppose now the packet is 1,500 bytes, the propagation speed on all three links is 2.5 * 10^8m/s, the transmission rates of all three links are 2 Mbps, the router processing delay is 3 msec, the length of the first link is 5,000 km, the length of the second link is 4,000 km, and the length of the last link is 1,000 km. For these values, what is the end-to-end delay for 2 packets with same length L that are

transmitted sequentially? You can draw flowgraph for easy calculation. (5 marks – evaluate)

Hint:
The first end system requires L/R_1 to transmit the packet onto the first link; the packet propagates over the first link in d_1/s_1; the packet switch adds a processing delay of d_{proc}; after receiving the entire packet, the packet switch connecting the first and the second link requires L/R_2 to transmit the packet onto the second link; the packet propagates over the second link in d_2/s_2. Similarly, we can find the delay caused by the second switch and the third link: L/R_3, d_{proc}, and d_3/s_3.

Adding these five delays gives

$$d_{end\text{-}end} = L/R_1 + L/R_2 + L/R_3 + d_1/s_1 + d_2/s_2 + d_3/s_3 + d_{proc} + d_{proc}$$

To answer the second question, we simply plug the values into the equation to get 6 + 6 + 6 + 20+16 + 4 + 3 + 3 = 64 msec.


```
 A     R1     R2     B

---   ----   ----   ---

 |      |      |      |

 |--------|--------|--------|
```

In this topology, "A" and "B" are end systems, "R1" and "R2" are routers, and the lines represent links connecting them.

b) The total end-to-end delay for a packet with no queuing delays can be calculated using the following formula:

=transmission+propagation+processing*dtotal*=*d*transmission+*d*propagation+*d*processing

Where:

- �transmission*d*transmission is the transmission delay,
- �propagation*d*propagation is the propagation delay, and
- �processing*d*processing is the processing delay.

The transmission delay (transmission*d*transmission) is given by *L*/*R*, where *L* is the length of the packet and *R* is the transmission rate.

The propagation delay (propagation*d*propagation) is given by propagation=length/*d*propagation =*d*length/*s*, where length*d*length is the length of the link and *s* is the propagation speed.

The processing delay (processing*d*processing) is the delay introduced by the routers and is constant (processing*d*processing).

processing$dtotal$=L/R+$d$length/s+$d$processing, where length$d$length, s, and R are specific to each link.

c) **End-to-End Delay for 2 Packets Transmitted Sequentially:**

The total end-to-end delay for two packets transmitted sequentially is the sum of the delays for each packet. Since they are transmitted one after the other, the second packet's transmission can start only after the first packet has been completely transmitted.

total=2×(L/R+$d$length/s+$d$processing)

Flowgraph:

```
 A     R1      R2     B
---    ----    ----   ---
 |      |       |      |
 | Pkt1  |      |      |
 |--------|     |      |
 |       | Pkt2  |     |
 |--------|--------|   |
```

This flowgraph represents the sequential transmission of two packets from A to B through routers R1 and R2. Pkt1 represents the transmission of the first packet, and Pkt2 represents the transmission of the second packet. The arrows indicate the direction of packet flow.

5. Consider an application that transmits data at a steady rate (for example, the sender generates an N-bit unit of data every k time units, where k is small and fixed). Also, when such an application starts, it will continue running for a relatively long period of time. (Total – 10 marks) Answer the following questions, briefly justifying your answer:

a. Would a packet-switched network or a circuit-switched network be more appropriate for this application? Why? (2 marks analyze)

b. Assume that there are multiple applications like that mentioned and requires reliability and elastic throughput. These applications are transmitting data simultaneously in the same link at the steady rate. Mention how multiple applications can share the same link. (2 marks apply)

c. Mention the transport layer service and protocol suited for these applications. (2 marks apply)

d. Analyse the performance of these multiple applications in given scenarios by filling the below table. (4 marks analyze)

	Circuit Switching	Packet Switching
When transmitted sum of application data rates less than the capacity of the link.		
When transmitted sum of application data rates greater than the capacity of the link.		

Ans:

a) A circuit-switched network would be well suited to the application because the application involves long sessions with predictable smooth bandwidth requirements. Since the transmission rate is known and not bursty, bandwidth can be reserved for each application session without significant waste. In addition, the overhead costs of setting up and tearing down connections are amortized over the lengthy duration of a typical application session.

b) Multiplexing is required if we wanted to share a link for sending data from multiple applications. FDM for circuit switching and Statistical multiplexing for packet switching.

c) TCP as the application requires reliability and elastic throughput is enough.

d) Filling the table.

	Circuit Switching	Packet Switching
When transmitted sum of application data rates less than the capacity of the link.	All applications can have performance and bandwidth will be equally shared. Frequency Division Multiplexing (FDM) can be used.	All applications can have good performance without packet loss or delay. Statistical multiplexing can be used.
When transmitted sum of application data rates greater than the capacity of the link.	Some applications cannot be allocated with link bandwidth resource and need to wait till the other applications allotted the bandwidth transmit.	There can be potential packet loss or delay and all applications can send data in the congested link.

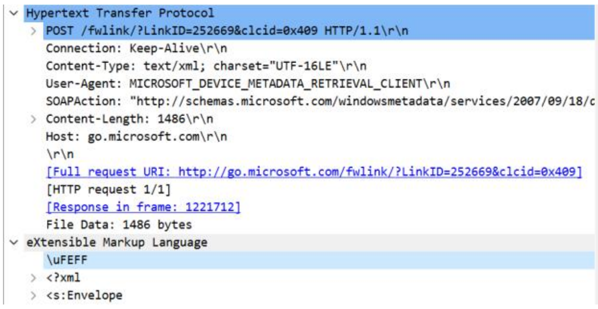
6. Consider the following scenario, where a web client requests a webserver to retrieve a webpage, which consists of 3 different jpeg images and one gif image in it. The size of the base HTML file is 128Kb, each jpeg image is of size 100Kb and the gif image is 50 Kb. [Assume that they are connected over a single transmission link of bandwidth 2 Mbps and the one Round Trip Time to be 4 ms]

a) Find the total time taken to download the webpage in the following cases. Illustrate your answer by creating the flowgraph for each of the scenarios.

   a. HTTP with a persistent connection and no parallel TCP connections (4 marks)

   a. HTTP with non-persistent connection and no parallel TCP connections (4 marks)

b) Identify the http message type and persistent or non-persistent connection based. Also justify your answer by mentioning the relevant fields from the screenshot. (2 marks)



```
Hypertext Transfer Protocol
  POST /fwlink/?LinkID=252669&clcid=0x409 HTTP/1.1\r\n
  Connection: Keep-Alive\r\n
  Content-Type: text/xml; charset="UTF-16LE"\r\n
  User-Agent: MICROSOFT_DEVICE_METADATA_RETRIEVAL_CLIENT\r\n
  SOAPAction: "http://schemas.microsoft.com/windowsmetadata/services/2007/09/18/c
  Content-Length: 1486\r\n
  Host: go.microsoft.com\r\n
  \r\n
  [Full request URI: http://go.microsoft.com/fwlink/?LinkID=252669&clcid=0x409]
  [HTTP request 1/1]
  [Response in frame: 1221712]
  File Data: 1486 bytes
eXtensible Markup Language
  \uFEFF
  <?xml
  <s:Envelope
```

Hint: b) POST upload message and persistent connection – since connection field mentions Keep-Alive

6) a) i) Time=RTT+Transmission time+Propagation time+Processing time+Queuing time

Given:

- RTT (Round Trip Time) = 4 ms
- Bandwidth (*B*) = 2 Mbps = 2×1062×106 bps
- Base HTML file size = 128 Kbits = 128,000 bits
- JPEG image size = 100 Kbits = 100,000 bits (each)
- GIF image size = 50 Kbytes = 50,000 bytes = 400,000 bits (1 byte = 8 bits)

i. HTTP with a persistent connection and no parallel TCP connections:

1. Transmission time (per resource) = SizeBandwidth BandwidthSize
2. Propagation time = 2 * RTT (for request and response)
3. Processing time and Queuing time are negligible in this case since we assume an ideal scenario with no processing or queuing delays.

Total time=RTT+3×Transmission time+Propagation timeTotal time=RTT+3×Transmission time+Propagation time

=4 ms+3×(128 Kbits2 Mbps)+2×4 ms=4ms+3×(2Mbps128Kbits)+2×4ms

ii. HTTP with non-persistent connection and no parallel TCP connections:

1. Transmission time (per resource) = SizeBandwidthBandwidthSize
2. Propagation time = 2 × RTT (for request and response)
3. Processing time and Queuing time are negligible in this case since we assume an ideal scenario with no processing or queuing delays.

Total time=RTT+4×Transmission time+Propagation timeTotal time=RTT+4×Transmission time+Propagation time

=4 ms+4×(128 Kbits2 Mbps)+2×4 ms=4ms+4×(2Mbps128Kbits)+2×4ms