Workshop – 1 Submission

Answer – 1:

(Part - I)

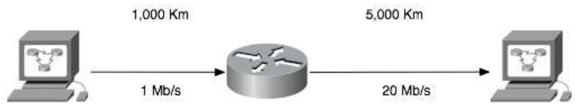

Figure 1. The Simplified network (Source ---- Router ---- Destination)

For given problem the following details are given:

- Number of Hops: 2
- Packet Data Size (L): 1000 bytes
- Propagation Speed ($S_{propogation}$): 2.5 x 10^8 m/s
- Distance Source_Router (D_{Source_Router}) = 1000 km
- Distance Router_Destination ($D_{Router\_Destination}$) = 5000 km
- Transmission Rate Source_Router (R_{Source_Router}) = 1 Mb/s
- Transmission Rate Router_Destination ($R_{Router\_Destination}$) = 20 Mb/s

Now we have to find total end-to-end delay assuming that no other traffic is presented. For this case the total end-to-end delay would be as follows:

$$t_{delay} = t_{delay(Source\_Router)} + t_{delay(Router\_Destination)}$$

First of all we'll find $t_{delay(Source\_Router)}$.

$$t_{delay(Source\_Router)} = (D_{transmission} + D_{propogation})$$

$$[(L/R_{Source\_Router}) + (D_{Source\_Router}/ S_{propogation})$$

After calculating, $\mathbf{t_{delay(Source\_Router)}}$ **= 0.001004 s.**

Then we'll find $t_{delay(Router\_Destination)}$.

$$t_{delay(Router\_Destination)} = (D_{transmission} + D_{propogation})$$

$$[(L/R_{Router\_Destination}) + (D_{Router\_Destination} / S_{propogation})$$

After calculating, $\mathbf{t_{delay(Router\_Destination)}}$ **= 0.002005 s.**

This would result in total end-to-end delay for a packet will be **0.003s**.

---

*x denotes multiplication throughout this file.

(Part - II)

Traffic intensity will play a crucial role in estimating the queueing delay, because if the traffic is heavy and other packets are also queuing for transmission, then the queuing delay will be long (Kuros and Ross, pp. 37) [1].

Traffic Intensity = (L x a) / R
Where L = packet size; a = average packet arrival rate; R = output transmission rate

$$\text{If Traffic Intensity } = \begin{cases} \dfrac{(\text{L x a})}{R} \approx 0, & \text{Queueing delay small.} \\ \dfrac{(\text{L x a})}{R} \to 1, & \text{Queueing delay becomes large.} \\ \dfrac{(\text{L x a})}{R} > 1, & \text{Average rate at which bits arrive exceeds} \\ & \text{the rate at which bits can be transmitted.} \end{cases}$$

*x denotes multiplication throughout this file.

Answer – 2:

If the network resources send data at a constant bit rate, for the case packet-switching and circuit-switching will work as both switching will use the network resources equally. According to (Kurose and Ross, pp. 22) [1] to send a message from a source-end-system to a destination-end-system, the source breaks long messages into small chunks of data known as packets. Between the source and destination, each packet travels through communication links and packet switches. Packets are transmitted over each communication link at a rate equal to the full transmission rate of the link. So, if a source-end-system or a packet switch is sending a packet of L bits over a link with transmission rate R bits/sec, then the time to transmit the packet is L/R seconds. While according to (Kuros and Ross, pp. 27) circuit-switching networks, the resources needed along a path to provide for communication between the end systems are reserved for the duration of the communication session between the end systems. Thus, it will use the available resources as the traffic rate is constant.

Answer – 3:

(Part - I)

For given problem the following details are given:

- File size (F): 10 Gbits = 10 x 10^9 bits
- Upload rate (U_s) = 20 Mbps = 20 x 10^6 s
- Download rate (D_{min}) = 1 Mbps = 1 x 10^6 s
- Number of links (N) = 10, 100, 1000
- Upload rate (U) = 200 kbps, 600 kbps, 1 Mbps

Now for calculating the minimum distribution time for client-server distribution the formula is as follows:

$$D_{Client\_Server} = \max\{(N \times F) / U_s , F / D_{min}\}$$

So for considering the case of Number of links = 10 and Upload rate of 300 kbps the distribution time will be 10,000 sec, and same calculating for the other links and upload rates the chart for client-server would be as follows.

Table 1. Distribution time for client-server distribution chart.

		Number of links (N)		
		10	100	1000
U	200	10000 sec	50000 sec	500000 sec
	600	10000 sec	50000 sec	500000 sec
	1	10000 sec	50000 sec	500000 sec

Now for calculating the minimum distribution time for P2P distribution the formula is as follows:

$$D_{P2P} = \max\{F / U_s , F / D_{min} , (N \times F) / (U_s + \sum_{i=1}^{N} Ui)\}$$

So for considering the case of Number of links = 10 and Upload rate of 200 kbps the distribution time will be 10,000 sec, and same calculating for the other links and upload rates the chart for client-server would be as follows.

Table 2. Distribution time for P2P distribution chart.

		Number of links (N)		
		10	100	1000
U	200	10000 sec	25000 sec	45454 sec
	600	10000 sec	12500 sec	16129 sec
	1	10000 sec	10000 sec	10000 sec

(Part - II)

According to (Kurose and Ross, pp. 149) [1] BitTorrent is a popular P2P protocol for file distribution. According to calculations and analysis, it seems that BitTorrent are suitable for larger files. The reason is that BitTorrent uses multiple computers and receive larger files in packets. Then the torrent breaks these packets into small pieces, thus letting you download that valuable file from many so that for every packet you download, it gets uploaded to another person who wants it. It doesn't just get it from one host but shares the loads from many hosts or different people at the same time. This can make it a much faster system for transferring large files than a P2P network.

*x denotes multiplication throughout this file.

Answer – 4:

(Part - I)

According to (Kurose and Ross, pp. 104, 105) [1] The HTTP header has four types of fields as follows:

- General header
- Request header
- Response header
- Entity header

Thus the header field has a blank between the header and the entity body for the response as well as request line as shown in figure 2.

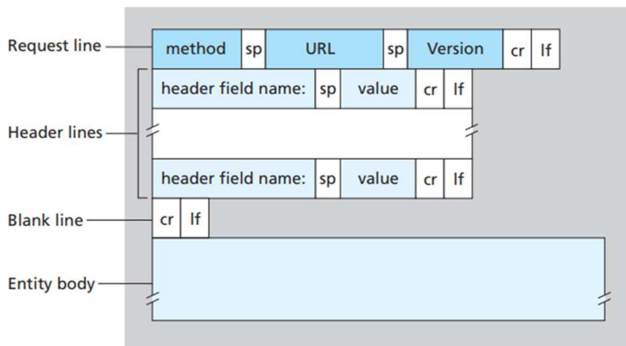

Figure 2. General format of an HTTP request message (Kurose and Ross, pp. 105) [1] .

The blank between this header needed such that framing mechanism which has been designed for requiring alterations. This eventually enhances the backward compatibility.

(Part - II)

The content length header is needed for HTTP as it helps in separating the needed content for references. The header provides the length in bytes of the content in the message. Finally, after a carriage return and line feed, the message contains the content. In this case, the content provides information about IP address and how to receive information/content (Kurose and Ross, pp. 630) [1] .

*x denotes multiplication throughout this file.

Answer – 5:

(Part - I)

According to (Kurose and Ross, pp. 139) [1] because of caching a local DNS server can also cache the IP addresses of TLD servers, thereby allowing the local DNS server to bypass the root DNS servers in a query chain. The reason for such event occurred is that the DNS is not cached in the process as the delay time would be only for DNS lookup. Also the webpages and images are cached there won't be delay. So the end-user will be able to view the webpage and/or images when the DNS will be resolved.

(Part - II)

According to (Kurose and Ross, pp. 139) [1], If a hostname/IP address pair is cached in a DNS server and another query arrives to the DNS server for the same hostname, the DNS server can provide the desired IP address, even if it is not authoritative for the hostname. Because hosts and mappings between hostnames and IP addresses are by no means permanent, DNS servers discard cached information after a period of time (often set to two days). In the event of this as the domain is cached in the local DNS, while the web-page is not cached, the time delay would due to the load time of the original web-page because the content needs to be fetched from the main server located at a greater distance.

(Part - III)

In such case as the domain and the web-page is not cached, the delay time will be higher as the domain and web-page's content are being transferred from the remote server.

For the problem statement we're assuming that the information/content transferring at the speed of light (3×10^8 m/s) and the Distance to access the website is 1500 Km.

For calculating the total time to transfer the data, the formula would be as follows:

$$\text{Time} = \text{Distance} / \text{Speed}$$

Which is 1500 / (3×10^5) sec. Now, if we consider that the data is in the image format and assumes the file size to be 1 Mb = 1024 bits, then the total time will be as follows:

$$\text{Total time} = (\text{File size} \times \text{Distance}) / \text{Speed}$$

Which is **5.12 sec** for transmission of 1 Mb image size.

*x denotes multiplication throughout this file.

Answer – 6:

The following topics, I think are most valuable and really helpful if we could review it in the workshop session.

- Packet Switching and Circuit Switching
- End-to-end delay for P2P distribution time as the calculation is bit tricky as well as the concept too.
- The concept of DNS caching
- Socket Programming (Even though it is self-learning module, a brief intro will help to understand it faster.)
- A quick intro to Wireshark (As there is many quizzes, we have to give each week)

References:

[1] Kurose, James F., and Keith W. Ross. *Computer networking: a top-down approach*. Addison Wesley.

*x denotes multiplication throughout this file.