

Primer Certamen

Tiempo 120 min.

1.- Responda en forma breve y clara

- a) Mencione un ejemplo en donde la conmutación de circuitos sea mejor que la conmutación de paquetes. ¿Por qué?

Resulta mejor cuando deseamos garantía en la tasa de bits obtenida y un retardo estable: por ejemplo en una conversación telefónica.

- b) Mencione dos aplicaciones que usen TCP y dos que usen UDP como protocolo de transporte.

TCP: La WEB, el correo electrónico, FTP, telnet.

UDP: DNS, Telefonía en Internet

Nota: SMTP es un protocolo, no una aplicación. Ítem http, La aplicación es la WEB o WWW, pero no páginas web.

- c) Mencione tres aplicaciones que usen el modelo o arquitectura Cliente-Servidor

FTP, la WEB, el correo electrónico, ssh, telnet

- d) ¿Por qué un cliente puede tener dirección IP dinámica pero un servidor no?

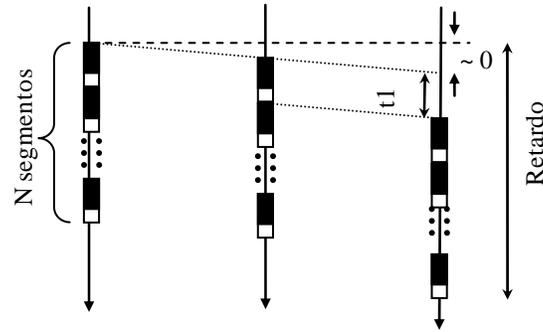
Porque el cliente es quien inicia la llamada o envía el primer requerimiento y en éste va la información que el servidor requiere para enviar la respuesta. Si el servidor obtuviera su dirección en forma dinámica, el cliente no sabría a quién hacer llegar el llamado.

Nota: Algunos hicieron notar correctamente que si se usara dirección IP dinámica habría que cambiar dinámicamente el DNS y usar valores pequeños para el tiempo de expiración de las cache.

- e) Cuando la información llega a la capa transporte de una máquina, ¿qué parte del paquete permite saber a qué aplicación o proceso debe hacerlo llegar?

El puerto.

2.- Considere el envío de un archivo grande de F bits desde Host A a Host B. Hay dos enlaces (y un switch) entre A y B, y los enlaces están descongestionados (esto es, el retardo de las colas es despreciable). El Host A segmenta el archivo en segmentos de S bits cada uno y agrega 40 bits de encabezado a cada segmento, formando un paquete de $L=40+S$ bits. Cada enlace tiene una tasa de transmisión de R bps. Encuentre el valor de S que minimiza el retardo de mover el archivo desde el Host A al Host B. Desprecie el retardo de propagación.



$$N = \lceil F / S \rceil$$

$$t_1 = (S + 40) / R$$

$$t_{\text{retardo}} = (N + 1) * t_1$$

$$t_{\text{retardo}} = (F / S + 1) * (S + 40) / R$$

$$\frac{dt_{\text{retardo}}}{dS} = (-40F / S^2 + 1) / R$$

$$\frac{dt_{\text{retardo}}}{dS} = 0 \Rightarrow S_{\text{Para_retardo_mínimo}} = \sqrt{40F} = 2\sqrt{10F}$$

Nota: En esta pregunta muchos fallaron. A mi juicio el problema fue no poder modelar la situación adecuadamente. Con un dibujo adecuado se facilita la solución. Se puede verificar que la segunda derivada es mayor que cero en ese punto, por lo cual se trata de un mínimo.

3.- Suponga que deseamos visitar una página WEB que contiene cuatro imágenes además de su texto. Todas las imágenes se encuentran en el mismo servidor. Si despreciamos los tiempos de transmisión, y sea T_0 el RTT entre navegador y servidor, cuánto tiempo demora el navegador en bajar la página:

- Se usa http no persistente y sin conexiones paralelas.
- Se usa http persistente con pipeline.
- Se usa http persistente sin pipeline.

a) T_0 para establecer la conexión + T_0 para traer página de texto que contiene las referencias a los objetos + $4 * 2T_0$ para hacer las cuatro conexiones y traer los objetos \Rightarrow Total = $10T_0$

b) T_0 para establecer la conexión + T_0 para traer el texto + T_0 para traer los cuatro objetos \Rightarrow Total = $3T_0$

c) T_0 para establecer la conexión + T_0 para traer el texto + $4T_0$ para traer los cuatro objetos \Rightarrow Total = $6T_0$

Nota: Es fundamental que pongamos atención en entender cómo funcionan las redes. Todo tiene una razón de ser y si no la saben pregúntenme. En este caso para bajar una página es preciso bajar primero el texto de la página html. En éste están las referencias a los otros objetos referenciados en ella. Por esto no podemos pedir ningún objeto sin antes recibir el texto (un RTT no importando el mecanismo de transferencia.)

Algunos destacaron correctamente que se asume que las direcciones IP del servidor es conocida (está en cache), es decir, no era necesario hacer la consulta DNS antes de traer la página.

4.- Luego de ejecutar ping un par de veces se obtiene:

```
agustin@agustin2005 ~$ ping www.google.com
PING www.l.google.com (66.249.87.99) 56(84) bytes of data.
64 bytes from 66.249.87.99: icmp_seq=1 ttl=245 time=228 ms

--- www.l.google.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 228.669/228.669/228.669/0.000 ms
agustin@agustin2005 ~$ ping www.google.com
PING www.l.google.com (66.249.87.104) 56(84) bytes of data.
64 bytes from 66.249.87.104: icmp_seq=1 ttl=245 time=206 ms

--- www.l.google.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 206.704/206.704/206.704/0.000 ms
agustin@agustin2005 ~$
```

- ¿Qué direcciones IP posee el nombre www.google.com?
- ¿Qué justifica el mapeo de un nombre de máquina a más de una dirección IP?
- ¿Qué servidor específico hace este mapeo?

- Posee al menos dos direcciones IP, la 66.249.87.99 y la 66.249.87.104.
- Lo justifica la posibilidad de tener varios servidores atendiendo las peticiones de un mismo servicio, así es posible distribuir la carga de requerimientos entre varias máquinas.
- El servidor autoritario para www.google.com es el encargado de asignar un mapeo distinto en la medida que el servicio ofrecido por www.google.com es requerido. Cabe destacar que el servidor DNS no distingue si la resolución de nombre es requerida por un navegador o por otra aplicación (ping en este caso).

5.- En una red de paquetes conmutados el retardo asociado a un nodo y su enlace de salida está dado por:

$$d_{\text{nodo}} = d_{\text{proc}} + d_{\text{cola}} + d_{\text{trans}} + d_{\text{prop}}$$

¿Cuál sería la expresión equivalente para una red de circuitos conmutados?

La expresión equivalente es: $d_{\text{nodo}} = d_{\text{proc}} + d_{\text{trans}} + d_{\text{prop}}$

Esto debido a que en conmutación de circuitos no hay encolamiento en los nodos, el canal de salida tiene capacidad asignada para cada circuito por lo tanto la tasa de llegada de bits corresponde a la tasa de salida para cada circuito establecido. Se requiere procesamiento para mover los datos del canal entrante al que le corresponda en la salida y los tiempos de transmisión y propagación son inevitables.

6.- Se tiene una red donde los usuarios generan tráfico entrante tipo web con tamaño promedio de objetos de 150000 bits, la tasa promedio de requerimientos del navegador (browser) de cada usuario es de 2 [objetos/s].

Antes de poner el cache institucional:

- ¿Cuál es la utilización de la red LAN con 4 usuarios?
- ¿Cuál es la utilización del enlace de acceso?

Demanda de tráfico = $4 * 2 [\text{obj/s}] * 150000 [\text{bits/obj}] = 1.2 \text{ Mbps}$, luego se tiene una utilización del 12% de la red LAN. (a)
se tiene una utilización del 80% del enlace de acceso. (b)



Luego se instala un cache institucional el cual tiene una tasa de éxito (acierto) del 75%.

- ¿Cuántos usuarios pueden ser atendidos sin copar la utilización de alguno de los enlaces? ¿qué enlace alcanza primero su máxima utilización?

La utilización de la red LAN es máxima cuando se usan 10 Mbps en las solicitudes de contenido, luego: $n * 2 * 150000 \text{ bps} = 10 \text{ Mbps} \Rightarrow n = 33$ usuarios

Por su parte, la utilización del enlace de acceso es máxima para cuando el 25% de los requerimientos completan su capacidad, luego:

$n * 2 * 150000 * 0.25 \text{ bps} = 1.5 \text{ Mbps} \Rightarrow n = 20$ usuarios.

Por lo tanto el número máximo de usuarios a atender es de 20. El enlace de acceso trabaja así con capacidad máxima.

- ¿Cuál es la utilización del otro enlace cuando se atiende el mayor número de usuarios?

En estas condiciones la utilización del otro enlace es de:
 $20 \times 2 \times 150000 \text{bps} / 10 \text{ Mbps} = 0.6 = 60\%$

Nota: Nuevamente es preciso entender bien qué ocurre en este escenario. Si el cache atiende el 75% de las solicitudes, significa que éste responde a esos contenidos y por lo tanto se evita ese tráfico por el enlace. Sin embargo, toda respuesta usa la red LAN y en todos los casos se hace desde el cache. Cuando el cache tiene el contenido, "no hay" tráfico por el enlace de acceso; cuando no lo tiene, igual pasa por el servidor Proxy y luego es enviado al navegador del usuario.