

**Primer Certamen (Tiempo: 90 min.)**

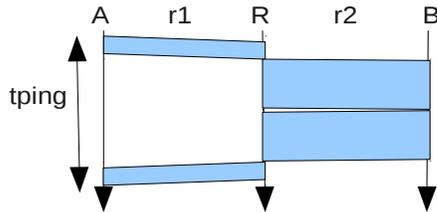
Si algo no está claro, haga un supuesto razonable, anótelos, y responda conforme a ello.

1.- (50 puntos) Responda en forma breve y clara:

- a) En conmutación de paquetes mencione una ventaja de usar paquetes de tamaño grande y una ventaja de usar tamaño de paquete pequeño.  
**Tamaño grande: Como el encabezado es de tamaño fijo, el usar paquetes grandes aumenta la eficiencia de lo enviado: más bytes transportados corresponden a información.**  
**Tamaño pequeño: Se reduce el retardo de las aplicaciones. No se requiere esperar acumular muchos datos para enviarlos.**
- b) ¿Por qué la fibra óptica es más inmune a ataques comparada con un par de cobre trenzado?  
**Porque es más difícil de intervenir para ponerse en el medio sin que el enlace lo note.**
- c) Indique cuál es la principal causa de paquetes perdidos en redes cableadas.  
**El espacio acotado de memoria de los ruteadores, la cual se puede llenar en situaciones de congestión.**
- d) Liste los nombres de las capas de servicio del modelo TCP/IP ubicadas bajo la capa aplicación e indique cuál es la función de cada una de ellas.  
**Transporte: Llevar paquetes desde un proceso en la máquina origen a un proceso en la máquina destino.**  
**Red: Rutear paquetes desde el computador origen al destino.**  
**Enlace: Transferir paquetes desde un nodo a otro adyacente.**  
**Física: Transferir bits a través de un enlace.**
- e) Explique cómo es posible que tres navegadores accedan a [www.google.cl](http://www.google.cl) y terminen bajando información de tres máquinas distintas. ¿Cómo funciona esto?  
**Porque el servidor de [www.google.cl](http://www.google.cl) está replicado en varias máquinas (granja de servidores) para poder atender a más clientes o brindarles un mejor servicio. Funciona a través del DNS autoritario de [www.google.cl](http://www.google.cl), cuando los navegadores desean resolver la IP para este nombre, este DNS responde con Ips diferentes según la carga de cada máquina.**
- f) Nombre tres diferencias entre el cliente y el servidor en aplicaciones de arquitectura cliente-servidor.  
**\* Cliente no requiere estar permanentemente conectado para usar un servicio, el servidor debe estar siempre corriendo.**  
**\* El cliente puede correr desde cualquier IP origen, el servidores debe tener una IP fija.**  
**\* El puerto del servidor debe ser conocido, el puerto del cliente no es relevante.**  
**\* El cliente es quien envía un requerimiento, el servidor es quien responde.**
- g) ¿Qué significa que una aplicación sea escalable?  
**Significa que el servicio que ofrece no se deteriora significativamente en la medida que aumenta el número de clientes.**
- h) Mencione una aplicación que tenga una arquitectura híbrida entre cliente-servidor y peer-to-peer y explique por qué.  
**Skype o Messenger. En ambos casos es de arquitectura híbrida porque el perfil de cada usuario está almacenado centralmente (es la parte cliente-servidor). Cuando se establecen llamados o chateos, éstos se comunicando a las máquinas de usuarios terminales directamente (es la parte peer-to-peer).**
- i) Una empresa decide actualizar la máquina de corre su servidor web. Explique por qué razón el servidor de la máquina antigua debería ser mantenido por un tiempo luego que el nuevo servidor entra en servicio. ¿De qué depende el tiempo en que ambos servidores están activos?  
**Debe mantenerse activo porque la asociación nombre del servidor – IP puede estar en el cache de varios DNS, los cuales seguirán enviando a la IP antigua a los clientes.**  
**El tiempo depende de cuál es el tiempo de expiración de la información guardada en esos caches. (Nota: este tiempo es el ttl almacenado en el servidor DNS autoritario para el nombre del servidor).**
- j) ¿Cómo es posible que su lector de correo muestre una dirección destinatario distinta de la suya y diferente a cualquier lista de correo?  
**Porque el protocolo SMTP permite que en el campo destinatario de la sección datos pongamos lo que se nos ocurra.**

2.- (25 puntos) Considere la comunicación entre dos computadores A y B conectados a través de un único router R. La tasa de bits del enlace entre A y R es de 100 Mbps. No sabemos la tasa del enlace R-B. Para estimarla, a uno de sus compañeros se le ocurre hacer desde A dos pings hacia B. En el primer ping, además del encabezado estándar, pone 100 bytes de datos y obtiene un tiempo de ida y vuelta total de 0,40 [ms]. En el segundo ping pone 200 bytes de datos y obtiene un tiempo total de ida y vuelta de 0,74 [ms]. Despreciando tiempos de procesamiento, colas y propagación, estime la tasa de bit del enlace R-B y el tamaño del encabezado de los paquetes ping.

La situación se puede modelar como sigue:



**r1: tasa enlace A-R,**

**r2: tasa enlace R-B,**

$$t_{ping1} = 2 \frac{(h+d)}{r_1} + 2 \frac{(h+d)}{r_2} = 2(h+d) \left( \frac{1}{r_1} + \frac{1}{r_2} \right)$$

$$t_{ping2} = 2(h+2d) \left( \frac{1}{r_1} + \frac{1}{r_2} \right) = 2(h+d) \left( \frac{1}{r_1} + \frac{1}{r_2} \right) + 2d * \left( \frac{1}{r_1} + \frac{1}{r_2} \right) = t_{ping1} + 2d \left( \frac{1}{r_1} + \frac{1}{r_2} \right) \text{ luego,}$$

$$\frac{1}{r_2} = \frac{t_{ping2} - t_{ping1}}{2d} - \frac{1}{r_1} \quad r_2 = \left( \frac{t_{ping2} - t_{ping1}}{2d} - \frac{1}{r_1} \right)^{-1} = \left( \frac{0,34 * 10^{-3}}{2 * 8 * 100} - \frac{1}{100 * 10^6} \right)^{-1} = 4,94 [Mbps]$$

$$t_{ping1} = 2(h+800) \left( \frac{1}{100 * 10^6} + \frac{1}{4,94 * 10^6} \right) = 2(h+800) * 0,21 * 10^{-6}$$

$$h = \frac{0,4 * 10^{-3}}{0,42} * 10^6 - 800 = 952 - 800 = 152 \text{ bits} = 19 [byte]$$

Según las aproximaciones que usted haga, puede llegar a valores levemente diferentes.

3.- (25 puntos) En esta pregunta se explora los tiempos de bajada de páginas web usando un proxy-cache y HTTP no persistente.

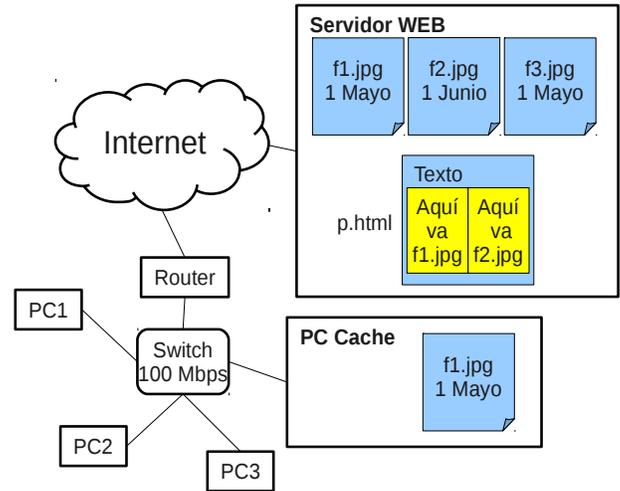
\* Durante la mañana del 1 de junio, el usuario del PC1 baja la página p.html desde el servidor web indicado en la figura. Desde que presiona return para pedir la página hasta que ésta llega al navegador pasan 1,51 [s]. Él reconoce una falta ortográfica en el texto y le avisa al dueño de la página quien hace la corrección del texto.

\* Luego el usuario del PC2 solicita la misma página p.html. Su tiempo de llegada de la página es de 0,52[s].

\* Finalmente el usuario del PC3 solicita sólo la fotografía f2.jpg y mide un tiempo de 10 [ms].

- a) Si luego el usuario del PC1 solicita sólo la fotografía f3.jpg, ¿qué tiempo de llegada de la foto estima usted?
- b) Con estos datos, ¿puede usted estimar el tiempo de llegada de la página p.html para el usuario del PC3? Desarrolle o explique.
- c) Si luego el usuario PC3 solicita sólo la foto f3.jpg ¿puede usted estimar el tiempo de su llegada? Desarrolle o explique.

Nota: En la figura se indica la situación inicial. Suponga que las condiciones de la red (congestión y otros) se mantienen durante las acciones pedidas. Suponga que inicialmente del cache sólo tiene el archivo indicado. En servidor se muestran los archivos y las fechas de su última actualización. La fecha/hora de p.html no se muestra pues cambia durante el escenario mostrado. Suponga vacía el caché local de cada navegador y que la IP del servidor es conocida por los PC de la red. Todas las fotos son de igual tamaño.



Cuando PC1 pide p.html, su computador se conecta con el proxy-caché, éste no tiene la página, luego éste la solicita al servidor. Cuando el texto de la página llega, éste la guarda en caché y la envía a PC1 y cierra la conexión. Sea todo este tiempo  $t_{servidor}$ .

El navegador de PC1 se da cuenta que debe pedir las fotos f1.jpg y f2.jpg. Pide en secuencia (HTTP no persistente) ambas fotos al proxy-caché. La vigencia de f1.jpg es verificada con el servidor y enviada a PC1. Sea  $f_{cache}$  el tiempo desde que se pide hasta que llega la foto al navegador. Al pedir f2.jpg, ésta no está en caché, es pedida al servidor, enviada a PC1 y guardada en caché. Sea  $f_{servidor}$  el tiempo para traer fotos desde el servidor.

Así para el tiempo hasta la llegada de la página al navegador tenemos:

$$t_{servidor} + f_{cache} + f_{servidor} = 1,51 [s]$$

Al término de la solicitud de p.html, el caché tendrá el texto y las dos fotografías, pero el primero inmediatamente queda obsoleto respecto a la versión en el servidor.

Cuando PC2 solicita la misma página, el tiempo para traer el texto será igual al caso anterior, como también el tiempo para traer ambas fotos desde el caché. Tendremos:

$$t_{servidor} + 2f_{cache} = 0,52 [s]$$

Cuando PC3 pide sólo la f2.jpg, el caché hará una verificación con el servidor, pero no será necesario bajarla, será el mismo tiempo de bajar las fotos del caché en caso anterior. Luego el tiempo hasta que llega al navegador es:

$$f_{cache} = 0,01 [s]$$

Así tenemos:  $f_{cache} = 0,01 [s]$        $t_{servidor} = 0,5 [s]$        $f_{servidor} = 1 [s]$

- a) La fotografía f3.jpg no está en caché su tiempo de llegada será 1 [s].
- b) El texto será retornado desde el caché, no sabemos cuánto retardo toma esto, pues no sabemos su tamaño ni RTT en red local. Sólo podemos decir que será igual a “el tiempo en traer el texto del caché” + 0,02 [s].
- c) El tiempo será de 0,01 [s] pues estará en caché por haber sido bajada desde PC1.