Teoría de las Comunicaciones
19 de Junio de 2024
2^{do} **Parcial**

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Apellido: [redacted]	LU: [redacted]	Hojas ->	Ej.1	Ej.2	Ej.3	Ej.4	
			1	1	1	1	
Nombres: [redacted]		Calif. ->	R	B	B-	B-	**Final:** A

Todas las respuestas se consideran válidas **solo** si están debidamente justificadas.

Ejercicio 1

El protocolo TCP permite 3 maneras distintas de cerrar la conexión. Describir 2 de los cierres explicitando los flags de los segmentos. *No hace falta aclarar los números de secuencia.*

Ejercicio 2

El control de congestión de TCP usa retroalimentaciones para cambiar el valor de la ventana de congestión (CWND). Explicar alguna de las retroalimentaciones aclarando cómo se actualiza CWND.

Ejercicio 3

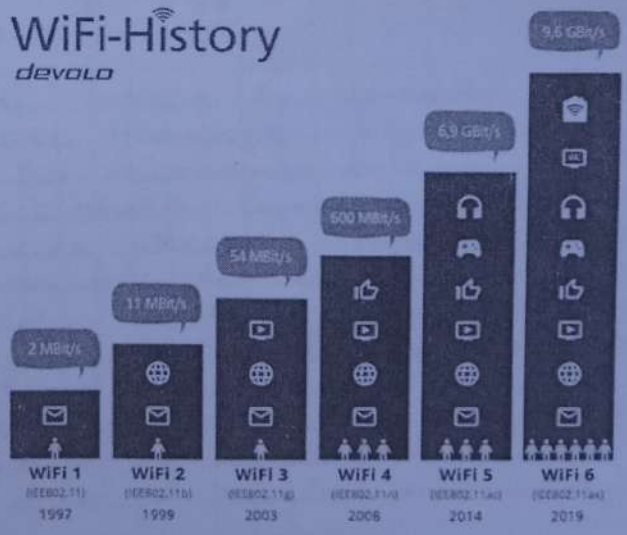
Suponiendo que un registro A asignado al nombre de dominio dc.uba.ar tiene un valor de TTL de 3 días, y cambia la dirección IP de 157.92.27.128 a 157.92.27.200. ¿Cuál será la respuesta de los *Resolvers* que tengan el registro en cache durante los 2 días posteriores al cambio?

Ejercicio 4

Explicar las diferencias entre los mensajes GET y POST de HTTP. Mostrar un ejemplo de cada tipo de mensaje.

Ejercicio 5 (OPCIONAL)

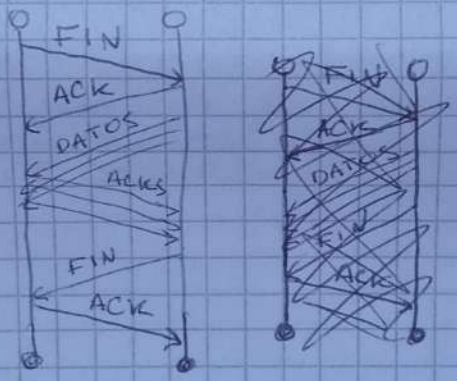
Asumí que tu hogar que es una casa con varios dormitorios, tiene un servicio de Internet por fibra óptica de 100 Megas de bajada y 10 Megas de subida. En la manzana todas las casas vecinas tienen contratado el mismo servicio que cuenta con un Wi-Fi 4 (ver Figura a la derecha) y Ethernet. Dos personas acceden vía Wi-Fi desde sus compus en cada dormitorio respectivo, vos desde tu compu en un escritorio y dos personas con el celu vía Wi-Fi están el Jardín. Estas bajando un archivo de 800 GB desde un servidor en USA, y te parece "lento" para los 100 Mbps tienes contratados. ¿Qué escenarios pueden darse por lo cual la performance sea baja?



(B)

Ejercicio 1

Una de las formas de cerrar una conexión TCP consta de cuatro mensajes. Cuando alguna de las dos partes de la conexión desea finalizarla envía un mensaje con el flag FIN activado (y posiblemente ~~un ACK~~ el flag ACK reconociendo bytes enviados por la otra parte anteriormente). Cuando la otra parte recibe el FIN, envía un segmento con el flag ACK encendido reconociendo el nro de secuencia del mensaje FIN que recibió (lo indica en un campo del header). De esta forma, el emisor del FIN cierra su extremo de la conexión, informando, con el FIN, que no enviará más datos. El extremo que no envió el FIN puede seguir enviando datos y estos deberían ser reconocidos (ACKed) por el otro. En algún momento este extremo enviará un segmento con el flag FIN activado indicando que no desea enviar más datos, la contraparte al recibir esto enviará un ACK reconociendo este FIN. y cerrará la conexión. Al recibir este último ACK, el emisor del FIN cerrará, también, la conexión.



Otra forma distinta de cerrar la conexión es mediante el intercambio de tres mensajes. Igual que en el caso anterior, uno de los extremos envía un segmento con el flag FIN activado. Cuando el otro extremo recibe esto puede decidir también cerrar su extremo de la conexión por lo que el segmento que envía no contiene solo el flag ACK activado, si no también el flag FIN. ~~El enviar esto, cada extremo~~ Cuando el que inició el cierre recibe esto, envía un segmento con el flag ACK activado. Como ambos extremos de la conexión fueron cerrados, cierra la conexión. El otro, al recibir este ACK, hará lo mismo.

Ejercicio 2. (B)

Una de las retroalimentaciones que utiliza el control de congestión es la llegada de ACKs(1) Se los considera una retroalimentación positiva ya que es un indicativo de que los paquetes están llegando a destino, es correcto asumir que la red no tiene congestión. Por esto, no solo se envía un nuevo segmento ~~en el utilizando el espacio que dejó~~ "ocupando" el lugar que liberó el que acaba de ser ACKed, si no que también se incrementa la CWND, esperando que la red sea capaz de soportar todavía más paquetes.

Cómo se actualiza la CWND depende del algoritmo que está ejecutando en ese momento el emisor. Si se encuentra en la fase de slow start, se incrementa el CWND en * MSS, esto consigue que en un RTT se duplique el tamaño de CWND, consiguiendo un crecimiento exponencial en el tiempo. (siempre que se mantenga en slow start)
Si en cambio, se encuentra en la fase de congestion avoidance, el incremento no es tan grande, se le aumenta una porción del SMSS. (Otra opción es no tomar los ACKs como realimentación y simplemente aumentando 1SMSS cada RTT).
Si se está en la fase de fast recovery la llegada de un ACK que reconoce nuevos datos hace que se establezca la CWND igual al SSTHRESH.

(1) De ACKs que reconocen nuevos datos en particular, no los duplicados.

* min (N, SMSS) con N la cantidad de nuevos datos reconocidos por el ACK.

B

Ejercicio 3

Si los resolvers almacenaron el registro en cuestión en su cache hace menos de 3 días entonces responderán con el registro que tienen almacenado*. Si, en cambio, lo almacenaron hace más de tres días, el registro estará vencido por lo que el resolver iniciará una búsqueda ~~recursiva~~ DNS iterativa ~~posiblemente con un root server mediante el que obtendrá el~~ ~~name server de la zona ar, con este obtendrá el de la zona uba.ar, con uba.ar obtendrá el name server de la zona dc.uba.ar. Este último name server que obtendrá (la IP del NS), será uno autoritativo por lo que deberá tener el correcto de IP reflejado, retornando esta nueva IP~~.

posiblemente con un root server, a menos que tenga algún otro match con el nombre dc.uba.ar. El root server no tendrá el registro A de dc.uba.ar por lo que retornará el NS de ar (y posiblemente su IP con un glue record). ~~El resolver le~~

~~consultará la IP del name server de ar por dc.uba.ar (el registro A), y análogamente a lo que sucedió antes, obtendrá la IP del NS de uba.ar, lo mismo ocurrirá al consultarle a esta última. Finalmente se le consultará al name server de dc.uba.ar. Como es un servidor autoritativo de su zona el registro A dc.uba.ar que contenga debe estar actualizado. De esta forma, se obtiene la IP actualizada~~

consultará la IP del name server de ar por dc.uba.ar (el registro A), y análogamente a lo que sucedió antes, obtendrá la IP del NS de uba.ar, lo mismo ocurrirá al consultarle a esta última. Finalmente se le consultará al name server de dc.uba.ar. Como es un servidor autoritativo de su zona el registro A dc.uba.ar que contenga debe estar actualizado. De esta forma, se obtiene la IP actualizada

* Es decir, el que tiene la IP sin actualizar.

B-

Ejercicio 4

El mensaje GET de HTTP indica que se está queriendo acceder a un recurso que ofrece determinado servidor en un determinado path.

El mensaje POST indica que se quiere enviar datos a un servidor a un determinado path.

Un posible mensaje GET puede ser

GET /archivos/redes.txt Host: dc.uba.ar.

que indica que el cliente quiere recibir el recurso archivos/redes.txt del servidor con nombre dc.uba.ar.

Un posible mensaje POST puede ser

POST ~~/archivos~~/nuevoArchivo Host: dc.uba.ar
→ y los parametros?

que indica que el cliente quiere enviar posiblemente, un archivo, al servidor con nombre dc.uba.ar.

esto no es tipo de HTTP

[Los contenidos del archivo estarán encriptados de alguna forma el el cuerpo del mensaje HTTP.]

Entre los headers del mensaje HTTP se encuentra el content-type. En el caso del GET el servidor indicará allí el tipo de recurso que está enviando. En el caso del POST, ~~entonces~~ el cliente indicará allí el tipo de datos que está enviando.