

## Soluciones - Segundo Parcial – 19 de noviembre de 2022

(ref: sol\_prc20221119.odt)

### Instrucciones

- Indique su nombre completo y número de cédula en cada hoja.
- Numere todas las hojas e indique en la primera la cantidad total de hojas que entrega.
- Escriba las hojas de un solo lado y utilice una caligrafía claramente legible.
- Comience cada pregunta en una hoja nueva.
- Sólo se responderán dudas de letra. No se responderán dudas de ningún tipo durante los últimos 30 minutos de la prueba.
- La prueba es individual y sin material. Apague su teléfono celular mientras esté en el salón de la prueba.
- Duración: 2 horas. Culminadas las 2 horas, el alumno no podrá modificar de ninguna forma las hojas.
- Justifique todas sus respuestas.

### Pregunta 1 (10 puntos)

Considere el protocolo TCP:

- a) Describa, en orden, los paquetes intercambiados para el inicio y el cierre de una conexión.
- b) Explique el mecanismo utilizado por TCP para detectar y recuperarse de la pérdida de segmentos.
- c) Explique la relación del mecanismo anterior con el *RTT* (*Round Trip Time*) y describa como TCP estima el *RTT*.

### Solución Pregunta 1

a) Inicio:

Paso 1: el host cliente envía un segmento con la bandera SYN encendida al servidor. Especifica el nro. de secuencia inicial. No contiene datos

Paso 2: el host servidor recibe el segmento con la flag SYN, responde con un segmento con las flags SYN y ACK encendidas y también sin datos. El servidor reserva buffers y especifica el nro. de secuencia inicial del servidor.

Paso 3: el cliente recibe el segmento anterior y responde con un segmento con la bandera ACK encendida. Este segmento puede contener datos.

Cierre:

Paso 1: el cliente envía un segmento con la bandera FIN encendida al servidor

Paso 2: el servidor recibe el segmento y responde con un segmento con la bandera ACK. Cierra la conexión y envía un segmento con la bandera FIN.

Paso 3: el cliente recibe el segmento y responde con un segmento con la bandera ACK.

Paso 4: el servidor recibe el segmento y cierra la conexión.

b) TCP crea un servicio RDT sobre el servicio no confiable de IP. El mecanismo utilizado para recuperarse de la pérdida de segmentos son los timers y retransmisiones. TCP utiliza un único timer de retransmisión. Las retransmisiones son disparadas por: eventos de timeout y ACKs duplicados

c) El mecanismo de utilizar timers tiene el desafío de establecer cual es el valor adecuado para dicho conteo a cero. Si el timer es menor al RTT produciría retransmisiones

innecesarias, si muy mayor tendría una lenta reacción a la pérdida de un segmento.

Es por esto que TCP estima el RTT (RTTEstimado) en base a un promedio de varias medidas recientes de RTT y no solamente el valor actual de “RTTMuestra”. En detalle:

el RTT de muestra (RTTMuestra), para un segmento es la cantidad de tiempo que transcurre desde que se envía el segmento (es decir, se pasa a IP) hasta que se recibe el correspondiente paquete de reconocimiento del segmento. En cualquier instante, RTTMuestra se estima a partir de uno solo de los segmentos transmitidos pero todavía no reconocidos, lo que nos proporciona un nuevo valor de RTTMuestra aproximadamente cada RTT segundos. Además, TCP nunca calcula RTTMuestra para un segmento que haya sido retransmitido.

TCP mantiene un valor promedio, denominado RTTEstimado, de los valores RTTMuestra. Para obtener un nuevo valor de RTTMuestra, TCP actualiza RTTEstimado según la fórmula siguiente:

$$\text{RTTEstimado} = (1 - \alpha) \times \text{RTTEstimado} + \alpha \times \text{RTTMuestra}$$

### Pregunta 2 (10 puntos)

Considere una red de dispositivos donde los protocolos utilizados en la capa de red y en la capa de enlace de datos son IP y Ethernet, respectivamente. En ese contexto general:

- a) Mencione y describa muy brevemente las 2 funciones principales de la capa de red y vincule cada una de ellas a los conceptos de Plano de Control y Plano de Datos.
- b) Considere 2 segmentos LAN (LAN1 y LAN2) interconectados mediante 1 router R1. Suponga que se debe enviar un datagrama IP desde la PC1 en LAN 1 hacia la PC2 en LAN 2. Para el caso más general, indique las direcciones MAC de origen y destino de todos los mensajes ARP necesarios para lograrlo. Asigne nombres genéricos (por ejemplo, MACPC1 para PC1) a las direcciones a utilizar en cada caso.
- c) ¿Con cuál de las funciones mencionadas en la parte a) se vincula directamente el protocolo ARP? Explique porqué.

### Solución Pregunta 2

a) Las dos funciones son reenvío (forwarding) y enrutamiento (routing).

El reenvío (asociado al plano de datos) hace referencia a la acción local que realiza un router al transferir un paquete desde una interfaz de un enlace de entrada a una interfaz del enlace de salida apropiado.

El enrutamiento (asociado al plano de control) hace referencia al proceso que realiza la red en conjunto para determinar las rutas extremo a extremo que los paquetes siguen desde el origen al destino.

b) Se asignan las siguientes direcciones MAC a las siguientes interfaces de red:

eth de PC1: MACPC1

eth de PC2: MACPC2

eth de R1 en LAN1: MACR1LAN1

eth de R1 en LAN2: MACR1LAN2

ARP request desde PC1 para conocer MAC de interfaz eth de R1 en LAN1

MAC origen: MACPC1, MAC destino: FF:FF:FF:FF:FF:FF

ARP reply desde R1

MAC origen: MACR1LAN1, MAC destino: MACPC1

-----se envía el paquete IP en LAN 1 -----

ARP request desde R1 para conocer MAC de interfaz eth de PC2

MAC origen: MACR1LAN2, MAC destino: FF:FF:FF:FF:FF:FF

ARP reply desde PC2

MAC origen: MACPC2, MAC destino: MACR1LAN2

-----se envía el paquete IP en LAN 2 -----

c) Se vincula directamente con la función de reenvío, pues a partir de la información disponible en la tabla de forwarding se determina por cuál de las interfaces del dispositivo se debe reenviar cada paquete; posteriormente se puede requerir el uso de ARP a los efectos de determinar la dirección destino de la Capa de Enlace de Datos (en general, dirección MAC "Ethernet"). De esa forma se podrá construir la trama que contendrá cada paquete, a ser reenviado en la interfaz de red correspondiente.

### Pregunta 3 (10 puntos)

- a) Suponga una red compuesta por tres *hosts* conectados al mismo *switch* sin VLANs configuradas y cada uno en un puerto distinto del mismo. Responda de forma justificada a las siguientes preguntas:
  - i. ¿Cuántos dominios de colisión hay en dicha red?
  - ii. ¿Cuántos dominios de broadcast hay en dicha red?
  - iii. ¿Cambiarían sus respuestas si ahora se reemplaza el *switch* por un *hub*?
- b) Describa el algoritmo CSMA/CD de acceso al medio utilizado en *Ethernet*.
- c) Describa el mecanismo de *backoff* exponencial binario ejecutado luego de una colisión.

### Solución Pregunta 3

a-i) 3 dominios de colisión. La propiedad de *store and forward* del switch permite que cada puerto sea un dominio de colisión independiente.

a-ii) Solo uno, pues estamos hablando de la misma LAN. El switch es un dispositivo de capa 2 que no separa subredes.

a-iii) Para la pregunta i ahora tendría un solo dominio de colisión dado que el HUB simplemente reenvía los bits que recibe por un puerto por todos los demás.

b)

1. El adaptador de red obtiene un datagrama de la capa de red, prepara una trama de capa de enlace y la coloca en el buffer del adaptador.
2. Si el adaptador detecta que el canal está inactivo (es decir, no hay señal de energía entrando al adaptador desde el canal), comienza a transmitir la trama. Si, por otro lado, el adaptador detecta que el canal está ocupado, espera hasta que detecta que no hay señal,

y luego comienza a transmitir la trama.

3. Mientras transmite, el adaptador monitoriza la presencia de energía de señales provenientes de otros adaptadores que usan el canal de transmisión.
4. Si el adaptador transmite la trama completa sin detectar señales de otros adaptadores, la transmisión de la trama se da por finalizada. Si, por el contrario, el adaptador detecta señales de otros adaptadores mientras transmite, aborta la transmisión (es decir, deja de transmitir su trama) y manda una señal de "jam" para que todas las estaciones se enteren que hubo colisión.
5. Despues de parar la transmisión, el adaptador espera una cantidad de tiempo aleatoria y luego regresa al paso 2.

c)

La espera aleatoria se implementa mediante el algoritmo de backoff exponencial binario. Cuando al transmitir una trama se experimentan  $n$  colisiones, el nodo elige un valor de  $K$  al azar entre  $\{0, 1, 2, \dots, 2^{n-1}\}$ . De este modo, cuantas más colisiones experimente una trama, mayor será el intervalo desde el cual  $K$  es elegido. Para Ethernet, la cantidad real de tiempo que espera un nodo es  $K$  veces la cantidad de tiempo necesario para enviar 512 bits a Ethernet (y  $n$  está topeado en 10).

#### Pregunta 4 (10 puntos)

- a) Describa el problema de conteo infinito de los algoritmos de vector-distancia y explique porque se produce. Muestre el problema utilizando un ejemplo sencillo.
- b) ¿Por qué el problema no se produce en los algoritmos de estado de enlace?
- c) Describa al menos un mecanismo que se puede usar para mitigar este problema.

#### Solución Pregunta 4

a)

Los algoritmos del tipo Bellman–Ford presentan problemas del tipo conteo a infinito (count-to-infinity) debido a que cada nodo solo recibe información parcial de sus vecinos. De cada vecino solo recibe un vector con los destinos a los que ese vecino puede llegar y el costo a ese destino. En particular, un nodo no conoce el camino de un vecino a un cierto destino. El conteo a infinito se da cuando un nodo informa a un vecino de que puede llegar a cierto destino pero el camino por el que llega pasa por ese nodo. En estos casos, cuando hay un cambio en la red, este nodo con esa información puede difundirla a sus vecinos pero la misma puede estar desactualizada.

Es decir, la causa central del problema parte de que A le dice a B que hay algún camino a cierto destino, no hay forma que B sepa que B es parte de dicho camino.

Se puede exemplificar con una red conectada como: A–B–C–D....., y la métrica a utilizar sea el número de hops. Supongamos que se desconecta A. B no recibe actualización desde A, pero si recibe una actualización desde C, quién no sabe que A se bajó. C piensa que está a 2 saltos de distancia desde C ( $C \rightarrow B \rightarrow A$ ), lo que es falso. Como B no sabe que es camino es a través de si mismo, actualiza su tabla con el nuevo valor: " $B \rightarrow A = 2 + 1$ ". Luego, B actualiza su información a C, y dado que C es alcanzable a través de B (desde el punto de vista de C), C decide actualizar su tabla a: " $C \rightarrow A = 3 + 1$ ". Esto se propaga lentamente a través de la red hasta llegar a infinito.

b)

El problema no se produce en los algoritmos de estado de enlace porque en este caso, cada nodo cuenta con una visión completa de la topología de la red y sobre esta información ejecuta de forma local un algoritmo para hallar los caminos de costo mínimo. Es decir, cada nodo conoce todo el camino de costo mínimo a un destino.

c) Se puede utilizar reversa envenenada. Este mecanismo consiste en que un nodo A cuando tiene un camino a un nodo D que pasa por un vecino B, a B le informa que su costo a D es infinito.

### Pregunta 5 (10 puntos)

- Cuando un paquete IP arriba a un *router* que no es el destino, este debe ser procesado para luego dejar el *router*, ¿cuál o cuales son los encabezados del paquete IP que se deben procesar para ello?
- ¿Es posible que al procesar un paquete IP exista más de una opción de siguiente salto en la tabla? Describa el método utilizado para seleccionar la mejor opción de reenvío.
- Suponga que un paquete con IP destino 172.16.1.64 ingresa a un *router*, cuya tabla contiene los siguientes prefijos: 172.16.0.0/23, 172.16.1.0/26 y 172.16.1.0/24. Muestre de forma precisa como hace el método en b) para este escenario y diga cuál es el seleccionado.

### Solución Pregunta 5

- Como mínimo tenemos 2 campos que son revisados para la tarea de reenvío: Destination IP Address y el campo TTL. Lo primero es que el TTL se decrementa, para saber si el mismo debe o no ser enrutado. Si el campo TTL llega al valor cero, entonces el paquete no será reenviado, y en su lugar el router responderá un ICMP Time exceeded al origen, es decir a la IP que está presente en el encabezado Source IP Address. Caso contrario la IP de destino es necesaria, porque el paquete será reenviado, y para saber hacia dónde es que se utiliza el destino.
- Cuando existen varias coincidencias, el router aplica la regla de coincidencia con el prefijo más largo; es decir, busca la entrada más larga de la tabla con la que exista una coincidencia y reenvía el paquete a la interfaz de enlace asociada con el prefijo más largo. Este mecanismo es conocido con el nombre de Longest Prefix Match (LPM por su sigla en inglés). Entonces la recorrida de la tabla debe ser sobre todos los prefijos y se elige el prefijo mas largo.

c)

#### Prefijo

10101100	00010000	00000000	--> X
10101100	00010000	00000001	--> Y
10101100	00010000	00000001	--> Z

Destino del paquete a reenviar

## Redes de Computadoras

10101100 00010000 00000001.01000000

En este caso el destino hace match con el primer y tercer prefijo, el segundo no coincide porque el último bit es diferente. De los 2 prefijos con coincidencia se decide por el tercero, ya que es el prefijo mas largo, por ende se reenvía el paquete por la interfaz Z.