

Solución - Segundo Parcial – 23 de noviembre de 2024 (ref: solucionprc2024123.odt)

Instrucciones

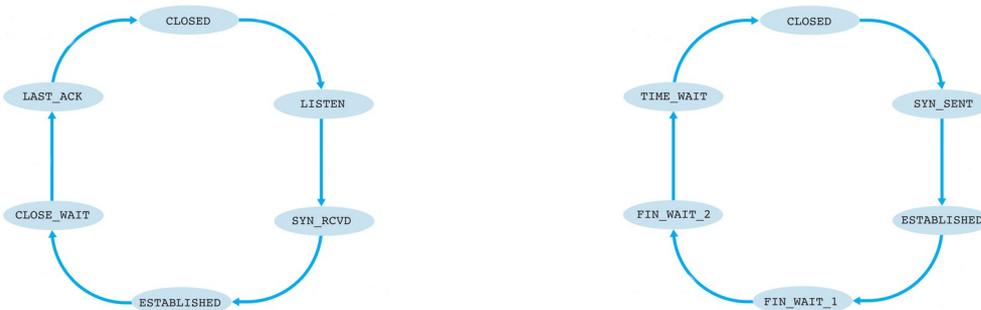
- Indique su nombre completo y número de cédula en cada hoja.
- Numere todas las hojas e indique en la primera la cantidad total de hojas que entrega.
- Escriba las hojas de un solo lado y utilice una caligrafía claramente legible.
- Comience cada pregunta en una hoja nueva.
- Sólo se responderán dudas de letra. No se responderán dudas de ningún tipo durante los últimos 15 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 los siguientes diagramas correspondientes a secuencias de estados seguidas por una entidad TCP.

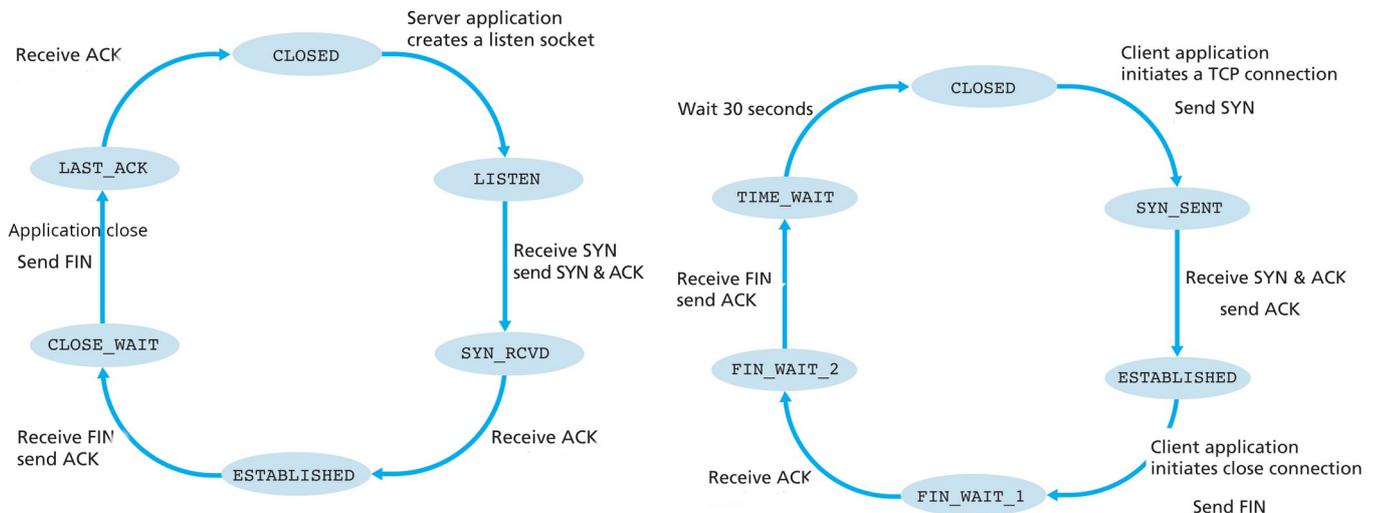
- Indique qué diagrama corresponde al lado cliente y cuál al lado servidor. Justifique.
- Complete los diagramas indicando para cada transición el evento por el que se toma la transición y la acción generada (cuando corresponda).

Solución



a) El de la izquierda es el servidor y el de la derecha el cliente. Se puede ver que el del cliente comienza con un envío de SYN mientras que el servidor comienza en el estado LISTEN, esperando por nuevos clientes.

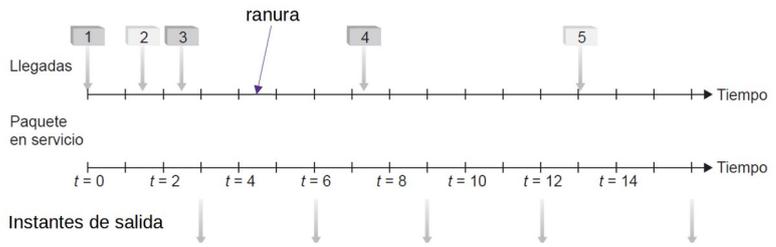
b)



Pregunta 2 (10 puntos)

- Defina el concepto *packet scheduling* ("planificación de paquetes").
- Considere el siguiente ejemplo donde una interfaz utiliza 2 clases de prioridades:

gris oscuro → alta prioridad (1,3,4)
gris claro → baja prioridad (2,5)
 Suponga que todos los paquetes son de igual tamaño ("3 ranuras") y que antes de comenzar a arribar el paquete 1, la interfaz estaba inactiva.



Indique qué paquete es enviado en cada instante de salida, para los *scheduling Priority* y *Round Robin*. Justifique.

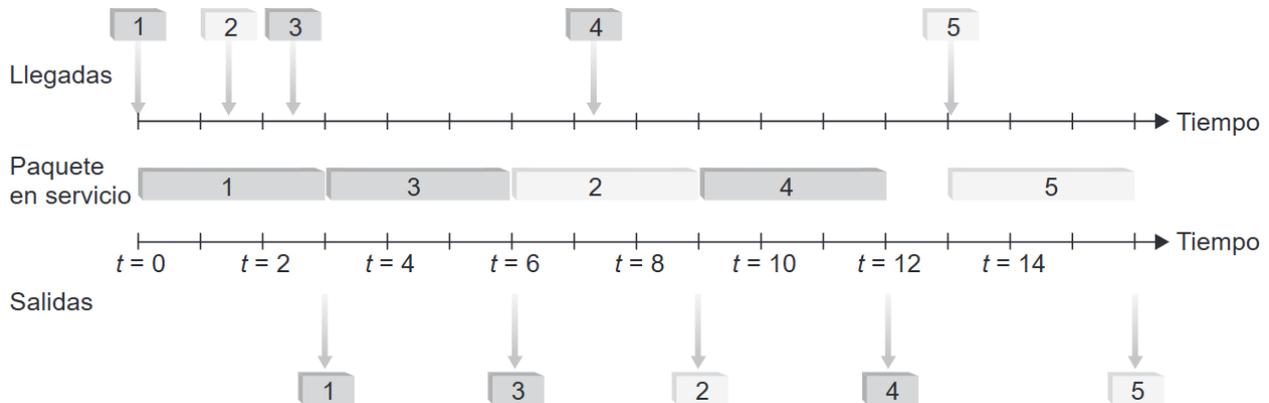
Solución

a)

Consiste en determinar el orden en el que se transmiten a través de un enlace saliente los paquetes existentes en la cola.

b)

Priority – el orden es 1, 3, 2, 4, 5



El paquete 1 llega y, al encontrar el enlace inactivo, comienza a ser transmitido. Durante la transmisión del paquete 1 llegan los paquetes 2 y 3, que se introducen en las colas de baja prioridad y de alta prioridad, respectivamente. Después de la transmisión del paquete 1, se selecciona para la transmisión el paquete 3 (un paquete de alta prioridad) con preferencia al paquete 2 (el cual, aunque llegó antes, es un paquete de baja prioridad). Al finalizar la transmisión del paquete 3, se comienza a transmitir el paquete 2. El paquete 4 (un paquete de alta prioridad) llega durante la transmisión del paquete 2 (un paquete de baja prioridad). Con la disciplina de cola con prioridad sin desalojo, la transmisión de un paquete no se interrumpe una vez que se ha iniciado. En este caso, el paquete 4 se pone en cola de transmisión y comienza a ser transmitido en cuanto se completa la transmisión del paquete 2. Finalmente, desde que termina de transmitir el paquete 4 hasta que puede transmitir el paquete 5, pueden ocurrir 2 cosas: quedarse bloqueado en la clase "de baja prioridad" esperando el próximo paquete (en este caso el 5) y hacer un "ping-pong" entre ambas clases, hasta que vuelve a la clase "de baja prioridad", encuentra el paquete 5, y lo transmite.

Round Robin – el orden es 1, 2, 3, 4, 5

Al terminar de transmitir el paquete 1, en la otra clase tiene disponible un paquete para transmitir, el 2, por lo que lo envía. Luego vuelve a la clase "de alta prioridad", encuentra el paquete 3, y lo transmite. A continuación en la clase "de baja prioridad" no encuentra paquete para transmitir; vuelve a la de alta prioridad, encuentra el paquete 4 y lo transmite. Finalmente, vuelve a la clase de "baja prioridad", encuentra el paquete 5 y lo transmite.

Una variante que se podría presentar (no solicitada en la letra) es que existiera un tiempo en cada clase, de espera a que aparezca un paquete para transmitir ("bloqueo"), antes de ir a la siguiente clase en el proceso de round robin. En ese caso, por ejemplo la transmisión del paquete 4 quedaría bloqueada hasta que apareciera el paquete 5 en su clase correspondiente.

Pregunta 3 (12 puntos)

- Mencione y describa el mecanismo utilizado por OSPF para conocer la topología de la red.
- ¿Por qué es necesario enviar mensajes HELLO con una dirección de destino de multicast conocida?
- ¿Cómo se utiliza la información topológica obtenida para resolver el problema de enrutamiento?

Solución

a)

OSPF utiliza el mecanismo de *flooding* (inundación) controlado para que todos los routers reciban toda la información topológica de toda la red. Cuando un router tiene un nuevo LSU (Link State Update) lo envía a todos sus vecinos, y luego cada router lo reenvía a todos sus vecinos excepto al que se lo envió. Para controlar la inundación se utilizan números de secuencia y TTL.

b)

Dado que en principio no se conocen los vecinos, y siempre pueden aparecer nuevos, los mensajes de descubrimiento HELLO se envían por todas las interfaces de un router con una dirección destino conocida que todos los routers aceptan. De esa forma, al recibir un mensaje HELLO, un router sabe que el origen del mensaje es un vecino, ya que estos mensajes tienen un alcance de un solo salto.

c)

La base de datos topológica sirve de entrada del algoritmo de Dijkstra que se ejecuta en cada router que implementa OSPF (y en general, enrutamiento Link-State), y resuelve el problema de los caminos más cortos desde el nodo origen (el router donde se ejecuta el algoritmo) a todos los demás nodos, construyendo el árbol de cubrimiento de caminos más cortos cuya raíz es el router que lo ejecuta.

Pregunta 4 (10 puntos)

- Describa los dos atributos principales de BGP y cómo se utilizan en el proceso de decisión cuando existe más de una posibilidad para una determinada ruta.
- ¿Por qué es necesario en un Sistema Autónomo interconectado con otros ejecutar algoritmos de enrutamiento interno (tales como OSPF o RIP) y externos (BGP)?
- Explique cómo se elige la interfaz de salida en un enrutador interno para enviar un paquete a un destino fuera del Sistema Autónomo.

Solución

a)

Una ruta en BGP consiste en un prefijo de red y un conjunto de atributos, donde destacan el NEXT-HOP (próximo salto hacia el prefijo destino), y el AS-PATH (lista de números de AS que se deben atravesar para llegar al prefijo destino).

En el proceso de decisión BGP la métrica utilizada es el largo del AS-PATH (por eso se dice que BGP es un algoritmo "path-vector").

Si esta métrica es igual para una determinada ruta, se consideran los NEXT-HOP (gateways), y se adopta la técnica de la "papa caliente", es decir, se utiliza el gateway que tiene una ruta de menor costo de acuerdo al protocolo de enrutamiento interno (por ejemplo OSPF o RIP).

b)

BGP permite determinar el gateway para alcanzar determinado prefijo externo, mientras que el protocolo de enrutamiento interno (IGP) encuentra una ruta para llegar a dicho gateway.

c)

De acuerdo a lo anterior, una vez que BGP determina el gateway ejecutando su proceso de decisión, el IGP encuentra un camino a dicho gateway dentro del AS.

Pregunta 5 (8 puntos)

- Mencione y explique brevemente el mecanismo de control de acceso al medio utilizado en Ethernet.

- b) Describa el mecanismo de *Self Learning* utilizado por los *switches* para armar su tabla de reenvío. Es necesario refrescar las entradas en esta tabla? En caso afirmativo, ¿por qué?

Solución

a)

El mecanismo se denomina Carrier Sense Multiple Access / Collision Detection (CSMA/CD).

Cuando la placa de red tiene una trama para enviar “sensa” o “escucha” el medio, y si no hay otra transmisión, comienza a transmitir. Si se produce una colisión, se corta la transmisión y se envía una señal conocida (“jam”). En cambio si no hay colisión, la transmisión fue exitosa.

En caso de colisión, la placa de red espera un tiempo determinado por un algoritmo de “backoff exponencial”, que consiste en que luego de M colisiones, se sortea un número K entre {0, ..., $2^{\text{exp}(M)} - 1$ } y se espera un tiempo $K * 512 * \text{Tbit}$, donde Tbit es el tiempo de transmisión de un bit.

b)

Un switch con su tabla vacía, cada vez que recibe una trama, asigna la correspondencia {puerto, MAC origen de la trama} y la ingresa en su tabla, con un determinado TTL. Si el destino ya tiene una entrada en la tabla, se envía por el puerto correspondiente; si no, se hace flooding de la trama. Cuando el sistema final responda esa trama a la dirección origen, el switch puede agregar una nueva entrada en su tabla. Como se mencionó, las entradas tienen un TTL, o sea, deben ser refrescadas, dado que se pueden conectar terminales nuevos, y/o mover terminales de un puerto a otro.