

Tercer Certamen

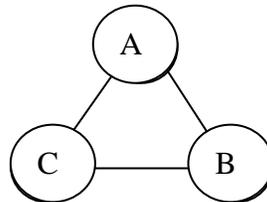
Tiempo 90 min.

1.- (20 pts.)

a) Muestre un ejemplo donde la estrategia de difusión "Inundación no controlada" genere una tormenta de difusión. (En Inundación no controlada cuando un nodo recibe un paquete lo envía a todos sus vecinos, menos el vecino desde donde llegó).

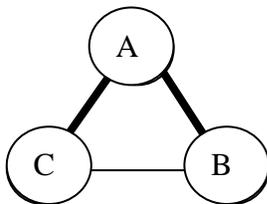
b) En su ejemplo muestre cómo Reverse path forwarding (reenvío por camino inverso) resuelve este problema.

a) Basta con tener un ciclo en la red.



Si enviamos una difusión desde A, saldrá hacia C y B, luego desde C sale a B y desde B a C. El paquete desde B a C luego es enviado hacia A, y el C a B es enviado hacia A. Como resultado tendremos dos paquetes circulando en ambos sentidos en la red.

b) En esta situación, supongamos que las líneas gruesas representan el camino a tomar cuando se envían paquetes de C a A y de B a A. En este caso la difusión es enviada desde A a C y B. Al llegar a B y C ésta es re-enviada a C y B. Cuando estas últimas llegan, no nuevos re-envíos son hechos por cuanto el camino de llegada no corresponde al camino inverso hacia la fuente (A). De este modo se evita la inundación no controlada.



2.- (12 pts.) A nivel enlace de datos, cuando se envían paquetes unicast en una red Ethernet éstos son dirigidos a la dirección MAC de la interfaz correspondiente. ¿Qué dirección MAC adoptan los mensajes multicast que van dirigidos a varias interfaces dentro de una subred?

La dirección MAC de los mensajes multicast no se resuelve vía ARP (Address Resolution Protocol), sino son mapeados directamente a una dirección MAC multicasts definida en Ethernet. La dirección MAC toma los 25 bits más significativos según el estándar Ethernet (o 802.3) destinados a direcciones multicast en IP y los 23 bits menos significativos de la dirección multicast IP para formar la dirección MAC.

3.- (24 pts.) Las aplicaciones multimedia actuales como telefonía IP usan UDP como protocolo de transporte porque las restricciones de tiempo real hacen inconveniente la retransmisión de paquetes usada en TCP.

a) ¿Qué ocurre con la tasa de envío del transmisor y pérdidas de paquete en TCP versus lo que ocurre en UDP cuando hay congestión en la red?

b) ¿Qué sugeriría usted en aplicaciones tipo telefonía IP para enfrentar situaciones de congestión?

a) Cuando hay congestión en algún router su buffer se tiende a llenar con lo cual se producen pérdidas de paquetes. La congestión afecta de distinta manera a ambos protocolos. Mientras UDP no tiene control de congestión, la tasa de envío depende únicamente de la aplicación sobre UDP, el cual no reacciona ante congestión; por otro lado TCP reduce la tasa de transmisión al detectar pérdidas, con lo cual la tasa de la aplicación también se reduce. UDP experimenta pérdidas de paquetes las cuales afectan a la aplicación y TCP detecta las pérdidas de paquetes, baja su tasa y se sobrepone a las pérdidas, el servicio que ofrece es siempre sin pérdidas.

b) Dado que telefonía IP usa UDP, sugeriría incluir a nivel de aplicación algún tipo de realimentación desde los receptores como lo sugiere RTCP en donde se informa la tasa de paquetes perdidos observada por receptores. Así el transmisor puede reaccionar bajando la tasa usando códigos de mayor compresión (bajando la calidad). Además de lo anterior sugeriría, incluir algún mecanismos de Forward Error Correction para sobreponerse a las pérdidas generadas por RED (Random Early Detection).

4.- (24 pts.)

a) ¿Cómo determina un transmisor TCP el tamaño de la ventana de recepción (receiver window)?

b) ¿Cómo TCP determina el MSS (Maximum Segment Size)?

c) TCP asume pérdida de paquete cuando se reciben tres acuses de recibo duplicados y cuando hay timeout. Indique qué ocurre con la ventana de congestión en cada caso y por qué se hacen cosas distintas en cada caso.

a) Lo termina en base a los acuses de recibo y el tamaño de ventana informada por el receptor en los mismos paquetes. Con esta información el transmisor sabe la capacidad de buffer del receptor cuando éste recibió el paquete correspondiente al acuse de recibo. A esta capacidad el transmisor descuenta el tráfico ya enviado y en viaje desde el envío del paquete cuyo acuse de recibo se informa. Así haciendo la resta de ventana informada por receptor menos cantidad de bytes en viaje, el transmisor determina el tamaño efectivo de la ventana de recepción.

b) (esta pregunta fue hecha en clases y se dio como estudio personal) TCP lo hace enviando paquetes de tamaños variados usando los MTU (Maximum Transmission Unit) de las tecnologías de enlace de datos conocidas. Adicionalmente fija un bit en los paquetes IP indicando a los router el No fragmentar el paquete. Cuando la fragmentación es obligada, el router envía un

mensaje a la fuente indicando que el mensaje debe ser fragmentado. Así el transmisor TCP probando con distintas MTUs determina cual es el tamaño mayor que no genera fragmentación y pasa a ser el MSS.

c) Cuando se reciben tres acuses de recibo duplicados se reduce la ventana de congestión a la mitad y se inicia un incremento lineal. Cuando hay timeout la ventana de congestión se reduce al tamaño de un segmento y se inicia la partida lenta hasta alcanzar la mitad de la ventana cuando se produjo el timeout. En ambos casos se reacciona distinto porque si bien en los dos casos hay pérdidas de paquetes (indicios de congestión), el recibo de acuses de recibo duplicados en el primer caso es una indicación que la congestión no es severa y una reducción de ventana podría bastar. En el caso de timeout no hay indicios de tráfico recibido por lo cual la congestión se asume severa y se reinicia la ventana desde su menor tamaño.

5.- (20 pts.) ¿Qué problema resuelve cada uno de los siguientes mecanismos:

- a) Algoritmo de Nagle,
- b) Random Early Detection (RED) ?

a) Nagle: procura reducir el overhead de los paquetes cuando la tasa de bytes de la fuente es baja respecto de la capacidad de su enlace. Nagle propone esperar por la espera de los acuses de recibo antes de enviar algún paquete que no ha completado el MSS. Si el paquete completa el MSS es enviado inmediatamente.

b) Random Early Detection resuelve el problema generado por la estrategia de descarte de paquetes cuando el buffer de los router se llena. En este caso todos los paquetes eran eliminados y consecuentemente todas las conexiones TCP reducían sus tasas liberando la condición de congestión pero en forma sincrónica reiniciando un nuevo periodo de congestión. El efecto neto de esta estrategia era el paso cíclico entre periodos de congestión y de bajo uso de los enlaces. RED propuso entonces iniciar el descarte de paquetes en forma aleatoria y por adelantado a una situación de llenado del buffer. Así diferentes conexiones reducen su tasa y la aleatoriedad usada evita sincronía de las distintas conexiones TCP logrando mejor throughput agregado en los routers.