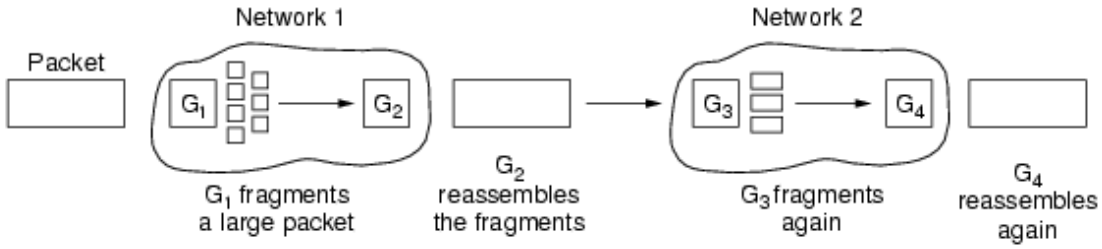
Respuesta Pregunta 1

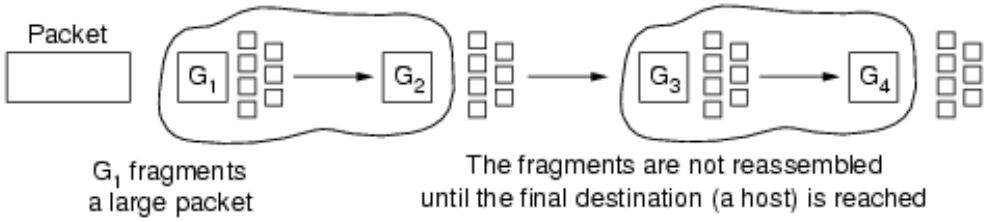
a) El proceso de fragmentación es necesario cuando un paquete grande pretende pasar por una red con un tamaño máximo de paquete demasiado pequeño. Los límites de tamaño máximo de paquete en una red, puede estar dado por diversas razones. Entre ellas se encuentran:

- El hardware (por ejemplo, el tamaño de la trama Ethernet)
- El sistema operativo (por ejemplo, todos los buffers son de 512 bytes)
- Los protocolos (por ejemplo, la cantidad de bits en el campo de longitud de paquete)

b) Fragmentación transparente:



Fragmentación no transparente:



b)

• identificador de paquete

- Todos los paquetes tienen un identificador generado por quien lo origina.
- Cuando se fragmenta un paquete, los fragmentos resultantes comparten el identificador, o sea que todos los fragmentos que pertenecen a un mismo paquete original tienen el mismo identificador. Esto es importante para reensamblar el paquete en destino.

• offset de fragmento

- Es un campo que identifica, para cada fragmento, en qué posición en bytes del paquete original se debe colocar el primer byte del fragmento.

• bandera de MF (More Fragments)

- Se usa para indicar cuál es el último fragmento. Cuando se fragmenta un paquete, todos los fragmentos menos el último tienen la bandera MF en 1. El último la lleva en 0.

• bandera de DF (Don't Fragment)

- Es para indicar que un paquete no debe fragmentarse.
- En caso que un enrutador reciba un paquete con DF=1 y lo deba encaminar por una interfaz que no permite transportarlo sin fragmentación, descartará el paquete y enviará un mensaje ICMP al originador del mensaje, indicándole que lo descartó.

Respuesta Pregunta 2

a) La ventana de recepción en el protocolo TCP se utiliza para el control de flujo, es decir, cumple la función de informar la cantidad de datos (medida en bytes) que un receptor es capaz de recibir. Cuando le arriban datos al receptor, éste los almacena en su buffer, y a medida que los va procesando va vaciando dicho buffer. La ventana de recepción indica el espacio libre en dicho buffer.
La manera que se utiliza en el protocolo TCP, es mediante su información al transmisor. Los receptores informan a los transmisores para no verse saturados. De esta manera, es importante que los transmisores la conozcan cada vez que van a enviar datos, para no sobrecargar los receptores (tener presente que este no es el único elemento que los transmisores tienen en cuenta a la hora de decidir cuantos datos enviar, por lo que en la práctica no siempre se envía todo lo que el receptor puede recibir).
Los receptores informan este valor a los transmisores en un campo del encabezado TCP existente a estos
efectos (WIN).

b) El algoritmo de Nagle intenta evitar en envío de segmentos TCP con muy pocos datos ya que eso hace que haya sobrecarga de encabezados (muchos bytes de encabezados para pocos bytes de datos). Para esto, cada vez que un transmisor tiene datos para enviar, procede de la siguiente manera:

1. En un principio se envía un segmento con la cantidad de datos que disponga el transmisor, y se espera su reconocimiento. Mientras espera el reconocimiento, el transmisor va almacenando los datos que le puedan ir enviando de la capa superior.
2. Una vez que el reconocimiento arriba, se envía un nuevo segmento, conteniendo todos los datos almacenados. Una vez más se espera el reconocimiento del anterior segmento, y se van acumulando los datos que deban ser transmitidos. Si se acumulan suficientes datos para llenar un segmento de tamaño máximo, se envían los datos aunque no haya llegado el reconocimiento del segmento anterior.
3. Se repite el punto 2.
La idea es que no pueda haber en tránsito (sin que el transmisor haya recibido el reconocimento) más de un segmento de tamaño no máximo.

c) El síndrome de la ventana tonta es un inconveniente que puede aparecer debido a los valores de ventana de recepción que los receptores informan. En el caso que un receptor tenga el buffer de recepción lleno, informará ventana cero, pero si un byte es procesado (leido por la capa superior), normalmente se enviaría una actualización del tamaño de la ventana, indicando que puede recibir un byte. Una vez que el transmisor recibe el informe de ventana, enviará un segmento con un byte de datos (el máximo disponible). Este proceso se podría repetir indefinidamente, llevando a una muy mala utilización del enlace, ya que se estarían enviando constantemente segmentos con un solo byte de datos, y por lo menos 40 bytes de encabezado (contando solo IP y TCP). Lo anterior es lo que se conoce como el síndrome de ventana tonta.
La solución que propuso Clark a lo anterior, es que los receptores no envíen las actualizaciones de tamaño de ventana cuando su valor es pequeño. De esta manera se evita la situación detallada anteriormente. Lo que se implementa es que se envía la actualización solamente cuando se tiene libre la mitad del buffer de recepción, o un valor igual al tamaño máximo se segmento previamente acordado, lo que sea menor de éstos valores.

Respuesta Pregunta 3

a. Primero es necesario distinguir dos aspectos de la recursividad. Un servidor puede configurarse para aceptar consultas recursivas o no. Además las consultas pueden formularse

en forma recursiva o no. Si suponemos que las consultas se formulan en forma recursiva, entonces:

Si un servidor de nombres acepta consultas recursivas, cuando se le realice una consulta realizará las búsquedas necesarias para devolver el resultado solicitado. Por ejemplo si

una máquina hace una consulta a un servidor recursivo, obtendrá como respuesta la información solicitada.

Si el servidor no acepta consultas recursivas, cuando se le consulte por un dominio y éste no se encuentre dentro de su zona de autoridad o de su caché, devolverá como respuesta los servidores raíz. De este modo quién realizó la consulta tendrá que comenzar la búsqueda desde la raíz.

b.i En cada uno de los puntos siguientes se enumera una consulta, y su respuesta correspondiente (en caso de ser inmediata). A continuación se enumera el primer grupo de consultas/respuestas, que aparecen para resolver www.fing.edu.uy.

1. En primera instancia la PC consulta a su servidor DNS local, del dominio ucauca.edu.co (llamémosle DNS1) por la dirección de www.fing.edu.uy.

2. DNS1 consulta a un servidor raíz la misma consulta que le realizaron a él. El servidor raíz le devuelve a DNS1 la dirección del servidor que maneja la zona uy.

3. A continuación DNS1 consulta al servidor que maneja la zona uy nuevamente por la misma dirección. Este ultimo, devuelve la dirección de quien maneja la zona edu.uy, llamémosle DnsEduUy.

4. DNS1 repite la consulta anterior, esta vez a DnsEduUy, el que le responde con la dirección el servidor DNS de la zona fing.edu.uy.

5. La ultima consulta de DNS1 es al servidor de fing.edu.uy, nuevamente por la dirección de www.fing.edu.uy. En esta ocasión, la respuesta ya contiene el dato buscado, la dirección IP de www.fing.edu.uy.

6. Por ultimo, si bien no se trata de una consulta DNS, sino que en realidad es la respuesta a la consulta número 1, DNS1 le devuelve a la PC el dato buscado, obtenido en la consulta 5.

Para la consulta por la dirección www.fmed.edu.uy, como no ha expirado el TTL de los datos que el servidor DNS1 obtuvo al resolver www.fing.edu.uy, éste utiliza algunas de las direcciones que tiene almacenadas en cache. En particular, en lugar de comenzar a consultar al servidor raíz lo hace directamente a DnsEduUy. A continuación se detallan las consultas:

7. En primera instancia la PC consulta a DNS1 por la dirección de www.fmed.edu.uy.

8. DNS1 consulta a DnsEduUy por la dirección de www.fmed.edu.uy, quien le responde el servidor de la zona fmed.edu.uy (llam´mosle DnsMed).

9. Luego DNS1 consulta a DnsMed por la dirección buscada, y en la respuesta ya viaja la dirección IP buscada.

10. Por ultimo DNS1 le devuelve la dirección recientemente hallada a la PC.

´

ii En esta parte, las consultas por la dirección www.fing.edu.uy son exactamente las mismas que para la parte i.

A continuación se enumeran las consultas para la dirección www.fmed.edu.uy, que debido a que se ejecutan luego de vencido el tiempo de vida de los datos

anteriormente obtenidos, deben recorrer nuevamente todo el árbol.

´

1. En primera instancia la PC consulta a DNS1 por la dirección de www.fmed.edu.uy.

2. DNS1 consulta a un servidor raíz la misma consulta que le realizaron a él. El servidor raíz le devuelve a DNS1 la dirección del servidor que maneja la zona uy.

3. A continuación DNS1 consulta al servidor que maneja la zona uy nuevamente por la misma dirección. Este ultimo, devuelve la dirección de quien maneja la zona edu.uy, llamémosle DnsEduUy.

4. DNS1 repite la consulta anterior, esta vez a DnsEduUy, el que le responde con

la dirección el servidor DNS de la zona fmed.edu.uy.

5. La ultima consulta de DNS1 es al servidor de fmed.edu.uy, nuevamente por la dirección de www.fmed.edu.uy. En esta ocasión, la respuesta ya contiene el dato buscado, la dirección IP de www.fmed.edu.uy.

6. Por ultimo, si bien no se trata de una consulta DNS, sino que en realidad es la respuesta a la consulta número 1, DNS1 le devuelve a la PC el dato buscado, obtenido en la consulta 5.

Respuesta Pregunta 4

a. En todas las tablas de ruteo siguiente, no aparecen las entradas que hacen referencia a

las redes a que cada uno de los equipos pertenecen (Salvo en el caso del router B, que no necesita entradas adicionales
).

Router A. El router A debe ser capaz de transmitir los paquetes de la máquina 1 hacia internet, que saldrán por el router C. Por lo tanto, la tabla de ruteo será:

Destino	Próximo Salto
0.0.0.0/0 (default)	200.40.30.1 (Router C)

Router B. El router B debe ser capaz de enviarle a la máquina 1 los paquetes que provienen de internet, desde el router C. Estrictamente, como la Máquina 1 pertenece a la misma red que el router B, no es necesario agregar ninguna entrada adicional a la tabla de rutas de dicho router, porque los paquetes que el mismo debe ser capaz de entregar tendrán como dirección destino, 192.168.1.100, a la que el ya “sabe” llegar con las entradas que aparecen por las interfaces. La tabla de ruteo completa para esta router aparece a continuación, y en el campo “Próximo salto”, simplemente se verá el nombre de la interfaz conectada a cada red.

Destino	Próximo Salto
192.168.1.0/24	eth0 (por ejemplo)
200.40.30.0/24	eth1 (por ejemplo)

Router C. El router C debe ser capaz de transmitir los paquetes de la máquina 1 hacia internet, y a su vez enviarle a ésta las respuestas. Por lo tanto, la tabla de ruteo será:

Destino	Próximo Salto
192.168.1.0/24	200.40.30.3
0.0.0.0/0 (default)	200.40.20.2

M´quina 1. La máquina 1 simplemente deberá enviar todos sus paquetes dirigidos a internet al router A. Por lo tanto, la tabla de ruteo ser´:

Destino	Próximo Salto
0.0.0.0/0 (default)	192.168.1.1 (Router A)

b. Cuando la máquina 1 quiera realizar una navegación a internet, enviará sus paquetes a alguno de los routers, A o B. Al recibir los datos, el router A por ejemplo, realizar´ NAT, asignándole a los paquetes de la máquina 1 una IP pública y luego enviará los mismos.

El problema aparece cuando los paquetes de dicha navegación vuelven, debido a la asignación dinámica de la IP pública. En caso que la vuelta sea por el mismo router A, no habrá problemas, pero si la vuelta es por B (en el caso en que por ejemplo A haya caído o que las métricas del protocolo de ruteo no sean simétricas, o que las distancias se hayan modificado), éste no sabrá a donde enviar los paquetes, ya que no tendrá conocimiento de la asignación. El NAT deberá realizarse en el Router C para evitar estos problemas.