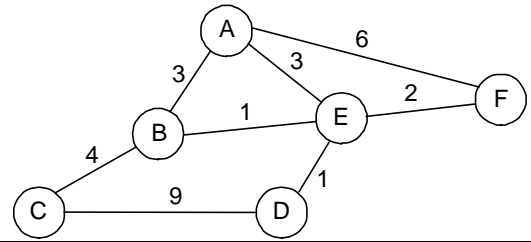


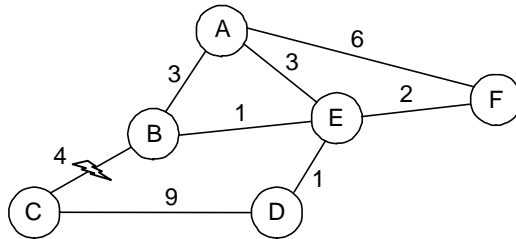
1.- Supongamos que en algún momento se reinicia el nodo A de una red que usa el algoritmo de estado de enlace para cálculo de rutas.

a) Muestre cómo se construye la tabla de ruta del nodo A desde la condición inicial indicada y luego de la llegada de los paquetes de estado de enlace de sus vecinos.



Paso	Conformado	Tentativo
1	(A,0,-)	
2	(A,0,-)	(B,3,B) (E,3,E) (F,6,F)
3	(A,0,-) (B,3,B)	(E,3,E) (F,6,F)
4	(A,0,-) (B,3,B)	(E,3,E) (F,6,F) (C,7,B)
5	(A,0,-) (B,3,B) (E,3,E)	(F,5,E) (C,7,B) (D,4,E)
6	(A,0,-) (B,3,B) (E,3,E) (D,4,E)	(F,5,E) (C,7,B)
7	(A,0,-) (B,3,B) (E,3,E) (D,4,E) (F,5,E)	(C,7,B)
8	(A,0,-) (B,3,B) (E,3,E) (D,4,E) (F,5,E) (C,7,B)	

b) Si la ruta entre B y C se corta, muestre cuál es la nueva situación de estado estacionario (después del corte) para cada nodo.



Nodo A			Nodo B			Nodo C			Nodo D		
Destino	Costo	Next Hop	Destino	Costo	Next Hop	Destino	Costo	Next Hop	Destino	Costo	Next Hop
A	0	-	A	3	A	A	13	D	A	4	E
B	3	B	B	0	-	B	11	D	B	2	E
C	13	E	C	11	E	C	0	-	C	9	C
D	4	E	D	2	E	D	9	D	D	0	-
E	3	E	E	1	E	E	10	D	E	1	E
F	5	E	F	3	E	F	12	D	F	3	E

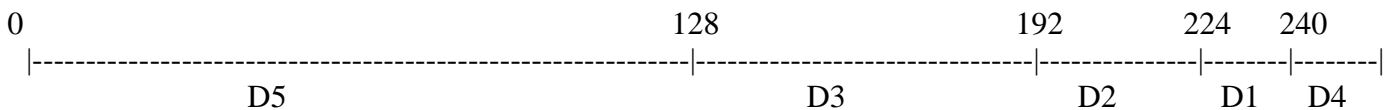
Nodo E			Nodo F		
Destino	Costo	Next Hop	Destino	Costo	Next Hop
A	3	A	A	5	E
B	1	B	B	3	E
C	10	D	C	12	E
D	1	D	D	3	E
E	0	-	E	2	E
F	2	F	F	0	-

2.- Una organización tiene cinco departamentos D1, D2, D3, D4, D5.

a) Indicar la distribución de subredes que usted sugeriría si se le asigna la Clase C 200.1.17.0 dados los requerimientos de direcciones IP indicados.

No hay solución única para este problema. Una solución posible es:

Departamento	D1	D2	D3	D4	D5
Nº IPs	10	29	57	13	125
Máscara Subred	255.255.255.240	255.255.255.224	255.255.255.192	255.255.255.240	255.255.255.128
Dirección Subred	200.1.17.224	200.1.17.192	200.1.17.128	200.1.17.240	200.1.17.0

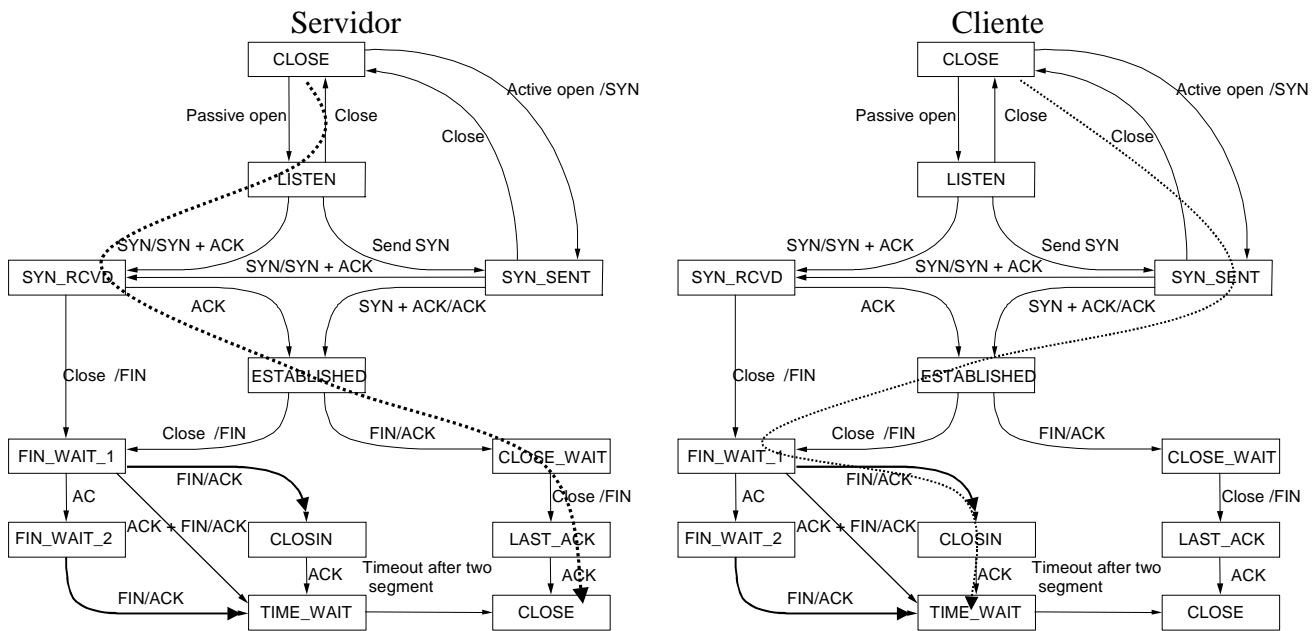


b) Es posible superponer dos o más subredes en la misma red física. Por ejemplo, conectando los equipos de D4 y D1 a los mismos switches Ethernet. Justifique su respuesta.

Sí, es posible. No hay impedimento en los protocolos de las capas 1 y 2 para efectuar esto. Se debe notar si que no porque exista conexión directa entre las máquinas de distintas subredes, ellas pueden comunicarse directamente. El paso por un router es necesario para que máquinas de ambas redes se puedan comunicar. Este router debe tener una IP asignada a en cada subred.

3.-

a) ¿Es posible tener la secuencia de estados indicada en los diagramas adjuntos para el cliente y el servidor?. Si no. Justifique. Si es posible justifique haciendo un diagrama temporal que muestre la secuencia de eventos (mensajes enviados, recibidos, y acciones de la aplicación - open/close) que generan el paso por los estados que se indican.



No es posible. Si podemos llegar en ambos lados al estado ESTABLISHED; sin embargo, la secuencia de estados del servidor para el cierre de la conexión con corresponde al escenario que hace que el cliente transite por los estados mostrados.

La situación muestra que el servidor recibe un FIN, con lo cual él responde con un ACK. Luego no es posible que el cliente reciba un FIN en lugar del ACK como se indica.

Como en TCP los acks con acumulativos la llegada del FIN en el cliente transportaría también el ack de su FIN, con lo cual una eventual pérdida del ACK enviado por el servidor podría conducir a un diagrama en el cliente pasando directamente del FIN_WAIT_1 al TIME_WAIT, pero nunca a través de CLOSING.

b) ¿Bajo qué circunstancias una conexión TCP bloquea a la aplicación transmisora?

Cuando debido a congestión o reducción de la ventana del receptor a valores muy pequeños, el transmisor no logra transmitir todos los datos que la aplicación le esta pidiendo transmitir y su buffer local se empieza a llenar. Cuando la capacidad de almacenamiento local se alcanza, la capa de transporte bloquea a la aplicación generadora de los datos siendo transmitidos.

c) ¿Qué problema se resuelve con el algoritmo de Nagle?

El problema es ¿Cuánto es razonable esperar en la capa transporte por datos de la aplicación antes de enviar un segmento TCP? Si la capa de transporte envía un segmento tan pronto le llega algo de la aplicación, podría estar enviando paquetes muy pequeños (sólo un byte de datos si yo estoy escribiendo por ejemplo) y se paga un alto overhead. Si por el contrario se espera, ¿cuánto debo esperar? dado que la capa transporte no sabe siquiera si la aplicación enviará más datos.

Nagle sugiere que continuemos llenando un segmento hasta que no tengamos ACK pendientes.

4.- a) Mencione y explique algún mecanismo de uso en Internet que apunte a mejorar la equidad del uso del ancho de banda entre los distintos flujo de datos.

El algoritmo FQ (Fair Queuing) es uno de ellos. La idea es crear en los routers diferentes colas de salida, una por cada flujo. Luego se da acceso al canal de salida de forma tal que todos los flujos presentes puedan avanzar en promedio con la misma tasa de bit por segundo. Para ello se considera el tiempo de llegada de cada paquete y su tamaño para programar su salida.

b) Mencione y explique un mecanismo introducido en Internet para abolir la congestión. Sea breve y conciso en sus respuestas.

RED (Random Early Detection) es uno de tales mecanismos. Una vez que la capacidad de almacenamiento del router supera cierto límite mínimo, éste descarta los próximos paquetes según alguna probabilidad. Esto permite que transmisor identifique una pérdida de segmento y luego reduzca su ventana de congestión, y por ende reduciendo su ventana efectiva. Se logra así reducir el flujo de la fuentes y mantenerse dentro de la capacidad de almacenamiento del router.

5.- En telefonía sobre IP la información de audio de ambas partes es normalmente enviada usando UDP.

a) ¿Podría usted dar alguna razón para elegir este protocolo? En otras palabras qué ventaja tiene UDP sobre TCP en este caso?

b) ¿Sigue siendo válido su argumento si se trata de la recepción del audio proveniente de una Radio Emisora en Internet?

a) Con uno de los argumentos siguientes basta:

i) Los paquetes de audio deben ser no muy grandes por razones de retardo (no conviene acumular muchas muestras antes de enviar el paquete). UDP impone un menor overhead que TCP, esto es especialmente notorio en paquetes pequeños (pensar en relación datos versus tamaño de encabezado UDP o TCP).

ii) Otra razón es el retardo de extremo a extremo de UDP. En TCP este es mayor por el procesamiento de ACKs y eventuales retransmisiones que retardan la entrega de los segmentos recibidos por el receptor. En UDP si no llega un datagrama, el receptor no espera por retransmisiones haciendo que la comunicación posea menor retardo. El costo pagado es la recepción con pérdidas, pero debido a la redundancia de la comunicación oral, esto no constituye problema; en el peor caso diremos “podrías repetírmelo, porfa’ ”.

iii) Finalmente podríamos argumentar que UDP requiere menos recursos de almacenamiento y CPU por parte del transmisor y receptor. No se requiere almacenar paquetes hasta la llegada de sus ACKs, ni la llegada de paquetes fuera de orden en el receptor. UDP tampoco tiene que llevar control de ACKs ni números de secuencia haciéndolo más liviano para la CPU.

b) Dependiendo de cual fue su argumento, esta parte sería

Argumento i) sigue siendo válido. El overhead de TCP es siempre mayor que el de UDP. Aún cuando los paquetes podrían ser de mayor tamaño en las transmisiones de radio (el retardo no es relevante), estos no superarán valores razonables de MTU (1500 bytes por ejemplo) para evitar fragmentación. Nunca usaremos valores cercanos a los admitidos por el datagrama UDP.

Argumento ii) No es aplicable, dado que en el caso de una transmisión en un sentido, el retardo no es relevante. Más aún es recomendable en este caso acumular unos cuantos segundos de datos antes de empezar a reproducir y así protegerse de futuras situaciones de congestión donde la tasa de recepción de datos pueda bajar.

Argumento iii) sigue siendo válido. La arquitectura de hardware de la radio e emisora IP es más simple y económica si usamos UDP.