Redes de Computadoras

Solución Segundo Parcial – 29 de noviembre de 2021

Nota
El parcial se tomó en modalidad virtual a través de la plataforma EVA.
Consistió en 6 preguntas de un set de preguntas que se sortearon de forma aleatoria para cada estudiante.
Tuvo una duración de 90 minutos.

Pregunta 1 (10 puntos)

V1
a) Describa detalladamente la principal funcionalidad del Plano de Datos y del Pano de Control de la Capa de Red.
b) ¿Cómo se relacionan entre si?

a)La función principal de la capa de red es transportar paquetes desde un host emisor a un host receptor. En la realización de esta tarea podemos identificar dos importantes funciones de la capa de red:
• A nivel del Plano de Datos, la funcionalidad principal consiste en el reenvío (forwarding). Cuando un paquete llega al enlace de entrada de un router, este tiene que pasar el paquete al enlace de salida apropiado. El reenvío es solo una de las funciones (¡aunque la más importante y común!) implementadas en el plano de datos. Entonces, el reenvío 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 y pertenece a lo que se denomina plano de datos.

• A nivel del Plano de Control , la funcionalidad principal consiste en el enrutamiento (routing). La capa de red tiene que determinar la ruta o camino que deben seguir los paquetes a medida que fluyen de un emisor a un receptor. Los algoritmos que calculan estas rutas se conocen como algoritmos de enrutamiento. El enrutamiento se implementa en el plano de control de la capa de red. Precisamente, el enrutamiento 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. Este proceso tiene lugar con escalas de tiempo mucho más largas (normalmente de segundos) y, como, suele implementarse en software

b) Se relacionan mediante la tabla de forwarding, es decir, los algoritmos de enrutamiento calcularán las rutas extremo a extremo y luego las instalarán en la tabla de forwarding el dispositivo de red (router) para que el proceso de reenvío pueda operar.

V2
Considere una red de datagramas que utiliza direcciones de host de 32 bits. Suponga que un router tiene cuatro enlaces, numerados de 0 a 3 y que los paquetes son reenviados a las interfaces de los enlaces como sigue:

Rango de direcciones destino	Interfaz de enlace
11100000 00000000 00000000 00000000 hasta 11100000 00011111 11111111 11111111	0
11100000 00100000 00000000 00000000 hasta 11100000 00100000 11111111 11111111	1
11100000 00100001 00000000 00000000 hasta 11100001 00111111 11111111 11111111	2
en otro caso	3

a) Describa cómo funciona la regla de coincidencia con el prefijo más largo (Longest Prefix Match).
b) Proporcione una tabla de reenvío (con la menor cantidad de entradas posibles) que utilice la regla de coincidencia con el prefijo más largo y que reenvíe los paquetes a las interfaces de enlace correctas para los datos proporcionados en la tabla anterior.

a) Los x bits más significativos de una dirección en el formato a.b.c.d/x constituyen la parte de red de la dirección IP y a menudo se los denomina prefijo. Dada la existencia de una tabla de reenvío, la búsqueda de un prefijo en ella resulta conceptualmente simple: la lógica hardware simplemente recorre la tabla de reenvío, buscando la coincidencia con el prefijo más largo. 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.

b) Para los dos primeros rangos, se pueden usar una única entrada en la tabla por cada rango que cubra cada al rango completamente y de forma ajustada: 224.0.0.0/11 y 224.32.0.0/16

Para el tercer rango, todas las IPs en el rango 11100000 00100001 00000000 00000000 (224.33.0.0) hasta 11100001 00111111 11111111 11111111 (225.63.255.255) deben ir por la if2. Esto no se puede ajustar a un único rango.

Por lo tanto, se deben agregar 2 entradas: 224.0.0.0/8 y la 225.0.0.0/10, ambas por la if 2. Por LPM las IPs en el rango 224.0.0.0 hasta 224.32.255.255 salen por if0 e if1 por las entradas anteriores.

Prefijo	Interfaz
11100000 000 (224.0.0.00/11)	0
11100000 00100000 (224.32.0.0/16)	1
11100000 (224.0.0.0/8)	2
11100001 00 (225.0.0.0/10)	2
En otro caso (0.0.0.0/0)	3

Otra opción, con mas entradas (pero que se acepta) consiste en agregar una entrada 224.0.0.0/7 por la if2 y luego excluir todo el rango desde 225.64.0.0 hasta 225.255.255.255 usando LPM. Eso no se puede hacer con una sola entrada en la tabla. Es necesario poner la 225.64.0.0/10 y la 225.128.0.0/9

Prefijo	Interfaz
11100000 000 (224.0.0.00/11)	0
11100000 00100000 (224.32.0.0/16)	1
1110000 (224.0.0.0/7)	2
11100001 01 (225.64.0.0/10)	3
11100001 1 (225.128.0.0/9)	3
En otro caso (0.0.0.0/0)	3

V3
Considere un *router* R1 que cuenta con la siguiente tabla de *forwarding*:

Prefijo	Gateway	Interfaz
140.100.0.0/21	100.100.100.100	eth0
140.100.4.0/23	100.100.101.101	eth1
140.100.8.0/24	100.100.102.102	eth2
220.100.200.0/24	DC	eth3

Suponga que el *router* R1 recibe en su interfaz eth3 un paquete con
Dir IP Origen: 220.100.200.2
Dir IP Destino: 140.100.5.2
Explique detalladamente las acciones que tomará el *router* hasta poder reenviar el paquete por la interfaz de salida que corresponda. Justifique.

b) Escriba la tabla de *forwarding* de R1 resultante, si se realizan los siguientes agregados:
(Debe agregar la menor cantidad de entradas posibles a la tabla)
- El *router* vecino con IP 100.100.100.100 conectado a la interfaz eth0 ahora también debe ser usado como *default gateway*.

- Se agrega una nueva conexión entre la interfaz eth4 con la interfaz 124.25.3.2 de un nuevo *router* vecino. A través de este *router* se pueden alcanzar los prefijos 76.32.24.0/23, 76.32.26.0/23 y 76.32.23.0/24.

Notar que en la propuesta, en la tabla de forwarding faltan entradas para las redes DC en las interfaces 0, 1 y 2. Esto se tuvo en cuenta en la corrección en caso de corresponder.
a) Observando la tabla de forwarding, la IP Destino "matchea" con las dos primeras entradas, 140.100.0.0/21 y 140.100.4.0/23
con la segunda de ellas es que tiene el LPM (Longest Prefix Match), por lo tanto reenviará el paquete a través de la interfaz eth1 y dirigido a la interfaz con IP 100.100.101.102, lo que implicará disponer de la dirección MAC asociada (utilizando ARP de ser necesario) a los efectos de armar la trama correspondiente y enviarla.

b) Para el primer punto, es necesario agregar una nueva entrada con prefijo "default" y cuya interfaz de salida y gateway sean eth0 y 100.100.100.100 respectivamente.
Para el segundo punto correspondería agregar tres nuevas entradas, una por prefijo, con gateway 124.25.3.2 e interfaz eth4. Sin embargo, los prefijos 76.32.24.0/23, 76.32.26.0/23 se pueden sumarizar en un único prefijo 76.32.24.0/22.

Por lo tanto tenemos:

Prefijo	Gateway	Interfaz
140.100.0.0/21	100.100.100.100	eth0
100.100.100.100/30	DC	eth0
140.100.4.0/23	100.100.101.101	eth1
100.100.102.100/30	DC	eth1
140.100.8.0/24	100.100.102.102	eth2
100.100.101.100/30	DC	eth2
220.100.200.0/24	DC	eth3
76.32.24.0/22	124.25.3.2	eth4
76.32.23.0/24	124.25.3.2	eth4
124.25.3.0/30	DC	eth4
0.0.0.0/0	100.100.100.100	eth0

Pregunta 2 (10 puntos)

V1
a) Describa cómo funciona el entramado de conmutación vía bus de un *router*.
b) ¿Cuáles son sus principales desventajas?
c) Comente otro entramado que mitigue estas desventajas (indicando cuales).

a) Conmutación vía bus. Con esta técnica, el puerto de entrada transfiere directamente un paquete al puerto de salida a través de un bus compartido, sin intervención del procesador de enrutamiento. Esto se suele realizar haciendo que el puerto de entrada añada al paquete como prefijo, antes de transmitir el paquete hacia el bus, una etiqueta interna al conmutador (cabecera), que indica el puerto local de salida al que se está transfiriendo dicho paquete. Todos los puertos de salida reciben el paquete, pero solo el puerto que se corresponda con la etiqueta lo conservará. Después, la etiqueta es eliminada en el puerto de salida, ya que dicha etiqueta solo se usa dentro del conmutador para atravesar el bus.

b)Si llegan múltiples paquetes al router simultáneamente, cada uno a través de un puerto de entrada distinto, todos los paquetes menos uno deberán esperar, ya que los paquetes deben atravesar el bus de uno en uno. Puesto que todos los paquetes tienen que atravesar el único bus, la velocidad de conmutación del router está limitada por la velocidad del bus; en nuestra analogía de la rotonda, es como si la rotonda solo pudiera contener un único vehículo cada vez. De todos modos, la conmutación vía bus suele ser suficiente para los routers que operan en redes de área local o redes empresariales de pequeño tamaño.

c) Conmutación vía una red de interconexión. Una forma de soslayar la limitación del ancho de banda de un único bus compartido, consiste en emplear una red de interconexión más sofisticada, como las que se han empleado en el pasado para interconectar procesadores en una arquitectura de computadora multiprocesador. Un conmutador de malla (crossbar switch) es una red de interconexión que consta de 2N buses que conectan N puertos de entrada a N puertos de salida. Cada bus vertical intersecta con cada bus horizontal en un punto de cruce, que puede ser abierto o cerrado en cualquier momento por el controlador del entramado de conmutación (cuya lógica forma parte del propio entramado de conmutación). Cuando un paquete llega a través del puerto A y necesita ser reenviado al puerto Y, el controlador del conmutador cierra el punto de cruce situado en la intersección de los buses A e Y, y el puerto A envía a continuación a través de su bus el paquete, que será transferido (solo) al bus Y. Observe que puede reenviarse al mismo tiempo un paquete del puerto B hacia el puerto X, ya que los paquetes de A a Y y de B a X utilizan diferentes buses de entrada y de salida. Por tanto, a diferencia del anterior, los conmutadores de malla son capaces de reenviar múltiples paquetes en paralelo. Un conmutador de malla es no bloqueante: un paquete que esté siendo reenviado hacia un puerto de salida no se verá bloqueado para alcanzar ese puerto, siempre y cuando no haya ningún otro paquete que esté siendo reenviado actualmente hacia ese puerto de salida. Sin embargo, si dos paquetes de dos puertos de entrada distintos están destinados a un mismo puerto de salida, entonces uno de ellos deberá esperar a la entrada, ya que solo se puede enviar un único paquete en cada momento a través de un bus determinado.

V2
Describa y ejemplifique el fenómeno de “*Head of Line Blocking*” o “bloqueo de cabeza” en un conmutador con colas de entrada, matriz de conmutación y colas de salida. ¿Dónde se produce?

El fenómeno de bloqueo de cabeza o bloqueo HOL (Head-of-the-line, cabeza de línea) se produce en la cola de entrada, cuando un paquete tiene que esperar a ser transferido a través de la matriz de conmutación, aunque su puerto de salida esté libre, porque está bloqueado por otro paquete que se encuentra en la cabeza de la cola de entrada, esperando que se libere otro puerto de salida.

Ejemplo: se considera un conmutador con tres colas de entrada E1, E2, E3, la matriz de conmutación M, y tres colas de salida S1, S2, S3. En un instante dado, con todas las colas de salida libres, al frente de las colas E1 y E3 se encuentran paquetes de color rojo destinados a la cola de salida S1, y en la cola E3, en segundo lugar, hay un paquete amarillo destinado a la cola de salida S2. Los paquetes rojos van a competir por la cola S1 (contención), y aunque el paquete amarillo tenga su cola de salida S2 libre, deberá esperar a que se transfiera el paquete rojo que está al frente para poder ser transferido.

Redes de Computadoras

V3
Se sabe que en un *router* el retardo máximo de cola es *(n − 1)D* si el entramado de conmutación es *n* veces más rápido que las velocidades de las líneas de entrada. Suponga que todos los paquetes tienen la misma longitud, que *n* paquetes llegan simultáneamente a los *n* puertos de entrada de un router y que los *n* paquetes desean ser reenviados a diferentes puertos de salida.

¿Cuál será el retardo máximo de un paquete para los entramados de conmutación (a) de memoria, (b) de bus y (c) de malla?. Justifique su respuesta en base a la descripción de los entramados.

a) Los puertos de entrada y de salida funcionan como dispositivos de E/S tradicionales. En este sentido no pueden reenviarse dos paquetes al mismo tiempo, incluso aunque tengan diferentes puertos de destino, ya que solo puede realizarse una lectura/escritura de memoria cada vez a través del bus compartido del sistema. Por lo tanto el retardo será (n-1)D.

b) Con esta técnica, el puerto de entrada transfiere directamente un paquete al puerto de salida a través de un bus compartido, sin intervención del procesador de enrutamiento. Esto se suele realizar haciendo que el puerto de entrada añada al paquete como prefijo, antes de transmitir el paquete hacia el bus, una etiqueta interna al conmutador (cabecera), que indica el puerto local de salida al que se está transfiriendo dicho paquete. Todos los puertos de salida reciben el paquete, pero solo el puerto que se corresponda con la etiqueta lo conservará. Si llegan múltiples paquetes al router simultáneamente, cada uno a través de un puerto de entrada distinto, todos los paquetes menos uno deberán esperar, ya que los paquetes deben atravesar el bus de uno en uno. Por lo tanto, nuevamente tenemos (n-1)D.

c) Un conmutador de malla (crossbar switch) es una red de interconexión que consta de 2N buses que conectan N puertos de entrada a N puertos de salida. Cada bus vertical intersecta con cada bus horizontal en un punto de cruce, que puede ser abierto o cerrado en cualquier momento por el controlador del entramado de conmutación. Al tener n paquetes en los n puertos de entrada para n puertos de salida diferentes, el retardo en este caso es 0.

Pregunta 3 (10 puntos)

V1
a) Liste en orden los campos del cabezal de IPv4, e identifique los campos relacionados con la fragmentación de paquetes.
b) Describa detalladamente un ejemplo donde sea necesario fragmentación y re-ensamblado.
c) ¿Cuáles son las diferencias en este tema para Ipv6?

a)
Versión| Largo Cabecera |Tipo de Servicio| Longitud del datagrama (bytes) | Identificador de 16 bits |Flags|Desplaz. de fragmentación de 13 bits| Time to Live | Protocol | Header Checksum | Dir. Origen | Dir. Destino | Opciones |

Relacionados con la fragmentación:
| Identificador de 16 bits |Flags|Desplaz. de fragmentación de 13 bits|

b)Ejemplo:

Un datagrama de 4.000 bytes (20 bytes de encabezado IP más 3.980 bytes de carga útil IP) llega a un enrutador y debe reenviarse a un enlace con una MTU de 1.500 bytes. Esto implica que los 3.980 bytes de datos del datagrama original deben asignarse a tres fragmentos separados (cada uno de los cuales también es un datagrama IP). Suponga que el datagrama original está marcado con un número de identificación de 777. La cantidad de datos de carga útil original en todos los fragmentos excepto el último debe ser un múltiplo de 8 bytes (porque el campo Desplaz. de fragmentación tienen 13 bits), y que el valor de desplazamiento se especifique en unidades de fragmentos de 8 bytes.
En el destino, la carga útil del datagrama se pasa a la capa de transporte solo después de que la capa IP haya reconstruido completamente el datagrama IP original. Si uno o más de los fragmentos no llegan al destino, el datagrama incompleto se descarta y no pasa a la capa de transporte.

Las características de los tres fragmentos son las siguientes:

1er fragmento:
Carga útil: 1480 bytes (con los 20 bytes de cabecera completa la MTU de 1500 bytes), Identificador: 777, Desplazamiento: 0 (es el primer fragmento), Flag: 1 (porque hay más fragmentos).

2o fragmento:
Carga útil: 1480 bytes, Identificador: 777, Desplazamiento: 185 (1480/8), Flag: 1 (porque hay más fragmentos).

3er fragmento:
Carga útil: 1420 bytes, Identificador: 777, Desplazamiento: 270 (1480*2/8), Flag: 0 (porque este es el último fragmento).

c)En IPv6, no se permite la fragmentación y reensamblado en enrutadores intermedios; estas operaciones solo pueden ser realizadas por la fuente y destino.

V2
A lo largo del curso se ha dicho en múltiples ocasiones que IPv4 *viola el objetivo de separación en capas.*

a) – explique como se diseñan los protocolos, conceptualmente orientados a capas. Explique ventajas de esta aproximación
b) – brinde un ejemplo conceptual de violación de la separación en capas
c) – muestre dos ejemplos de cómo IPv4 viola la separación en capas
d) – muestre un ejemplo de cómo ICMPv6 es más ordenado respecto a la aislación de capas.

a) Según Kurose, 7a Edición 1.5: “los diseñadores de redes organizan los protocolos en capas. Cada protocolo pertenece a una de las capas (...) Estamos interesados en los servicios que ofrece una capa a la capa que tiene por encima, lo que se denomina modelo de servicio de la capa. (...) cada capa proporciona su servicio (1) llevando a cabo ciertas acciones en dicha capa y (2) utilizando los servicios de la capa que tiene directamente debajo de ella. (…) Las capas de protocolos presentan ventajas conceptuales y estructurales [RFC 3439]. Como hemos visto, las capas proporcionan una forma estructurada de estudiar los componentes del sistema. Además, la modularidad facilita la actualización de los componentes del sistema.”

b) Cuando un protocolo de capa n, en vez de utilizar servicios o abstracciones de las capas n+1 o n-

Redes de Computadoras

1, utiliza servicios de una capa no-adyacente, es decir, n+2 o n-2, por ejemplo. También se puede ver cuando una capa accede y/o utiliza información correspondiente a otra capa por fuera de la interfaz que esta provee o resuelve algún problema perteneciente a otra capa.

c) Un ejemplo se da en el formato de la URL en HTTP, donde las URL permiten se especifique una dirección de red, con el formato: http://192.168.1.1. En este caso, un protocolo de capa de aplicación (5) utiliza elementos de la capa de red (3) para identificar recursos.
Otro ejemplo puede verse con NAT, donde se tiene dispositivos intermedios (por ejemplo routers) que modifican información tanto de capa 3 como de capa 4
También con NAT y algunos protocolos, por ejemplo FTP.. A efectos que el FTP clásico funcione a través de un router NAT es necesario modificar el stream de datos de control y “cambiar” la IP origen del cliente FTP por el servidor NAT cuando se realiza el connect, más allá de los cambios de IP y puerto estudiados

d) En IPv6 ICMPv6 reúne el funcionamiento que en IPv4 realizan ARP, ICMP e IGMP. Todas las comunicaciones se realizan utilizando IPv6 gracias a la existencia de SLAAC (Stateless Address Auto Configuration), lo que permite que desde la inicialización, todas las interfaces de red cuentan con direcciones IPv6 válidas. ARP utiliza el broadcast a nivel Ethernet, cuando el protocolo Neighbor Discovery de IPv6 utiliza direcciones IPv6 específicas para dirigirse a algunos grupos (all hosts, all routers, entre otros). ARP es una adaptación, que se hace como protocolo sobre Ethernet y queda al mismo nivel que IPv4, mientras que ND en IPv6 es un protocolo dentro de ICMPv6 y utiliza direcciones IPv6 válidas en todas las comunicaciones.

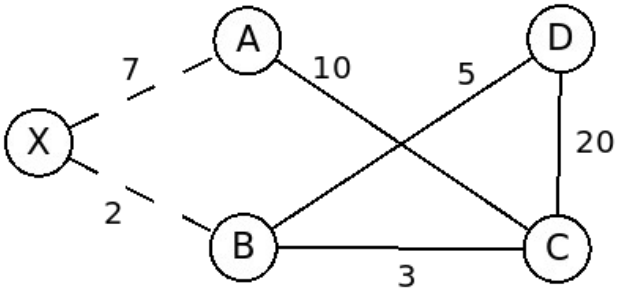
V3
Una de las características introducidas por IPv6, que lo diferencian de IPv4, es que IPv6 no permite la fragmentación.
a) Asumiendo que usted debe transferir 1GB de datos por una conexión TCP, ¿puede hacerlo sobre IPv6? ¿cómo se resuelve este problema en IPv6?
b) Describa el propósito y el funcionamiento del *PathMTU Algorithm* en IPv6, mensajes intercambiados, y los protocolos involucrados.

a) Si, es posible utilizar IPv6 para transferir contenido de tamaño arbitrario, sin inconvenientes. IPv6 no permite la fragmentación realizada por routers intermedios pero ofrece la posibilidad de determinar el MTU del camino (pathMTU) además de garantizar un MTU mínimo.
Con esta información se configura el MSS de TCP par que este realice la segmentación y reensamblado (en una capa superior a IPv6). Es decir, se determina un tamaño seguro de mensajes que no van a ser descartados y se transmite con mensajes de dicho tamaño máximo (tantos como sean necesarios para transferir cualquier volumen de datos)

b) Path MTU Discovery Algorithm es usado en la adaptación de TCP a IPv6 a efectos de determinar un MTU seguro y segmentar de forma acorde. La especificación de IPv6 garantiza que cualquier implementación debe poder enviar un datagrama IP de 1280 bytes sin ser fragmentado, por lo que, se conoce un MTU seguro. A través de diferentes algoritmos se puede probar con datagramas de diferentes tamaños hasta encontrar el óptimo para el destino específico. Cuando un router descarta un mensaje, genera un mensaje ICMP Packet Too Big (Tipo 2) conteniendo su MTU, con destino al emisor del mensaje descartado. Éste puede ajustar su MTU al informado, sin garantías que éste no deba ser decrementado en otro hop posterior.

Pregunta 4 (10 puntos)

V1
Considere la red de la figura donde se está ejecutando un algoritmo de Vector de Distancias y se agrega el nodo X y los dos enlaces punteados.



Considerando que cuando se agrega el nodo X el algoritmo de enrutamiento ya había convergido en los demás nodos:
a) Formule los vectores distancia que recibirá X al unirse a la red.
b) A partir de los vectores distancia anteriores calcule el vector distancia de X, explicando para cada valor el cálculo realizado.
c) Recalcule los vectores distancia de A y B al recibir el vector distancia de X calculado en la parte anterior. Explique los cálculos de los valores que se modifiquen.

a)Vector de A: X=7, B=13, C=10, D=18

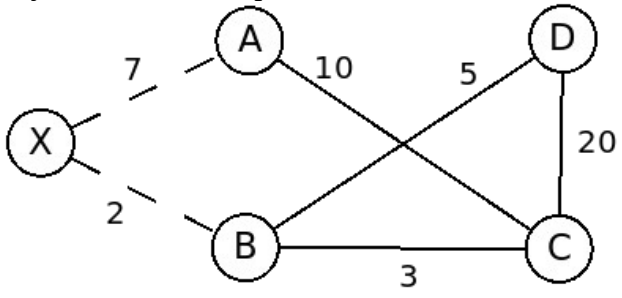
Vector de B: X=2, A=13, C=3, D=5

b) Vector de X.
A=7, el mínimo entre, el costo de ir directo (7) y el costo de ir a B (2) mas la distancia informada por B a A (13)
B=2, el mínimo entre, el costo directo (2) y el costo de ir a A (7) mas la distancia de A a B (13).
C=5, el mínimo entre, el costo de ir a A (7) mas la distancia de A a C (10) y el costo de ir a B (2) mas la distancia de B a C (3)
D=7, el mínimo entre, el costo de ir a A (7) mas la distancia de A a D (18) y el costo de ir a B (2) mas la distancia de B a D (5)

c) Vector de A: X=7, B=9, C=10, D=14. Se modifica B y D ya que hay un camino de menor costo yendo por X

Vector de B: X=2, A=9, C=3, D=5. Se modifica A ya que hay un camino de menor costo por X

V2
Considere la red de la figura donde se está ejecutando un algoritmo de Estado de Enlace y se agrega el nodo X y los dos enlaces punteados.



a) Describa la información (mensajes) que recibe el nodo X al unirse a la red.
b) Describa el algoritmo que deberá ejecutar X para calcular los caminos de menor costo a todos los nodos de la red.

a) Recibe mensajes de todos los nodos de la red. Cada nodo difunde en toda la red la información relacionada con sus enlaces. Por lo tanto, reciba mensajes del estilo:
A: X=7, C=10
B: X=2, C=3, D=5
C: A=10, B=3, D=20
D: B=5, C=20

b)Describir el algoritmo de Dijkstra.

Supongamos que se desea calcular las rutas de coste mínimo desde un nodo u a todos los destinos posibles. El algoritmo de Dijkstra consiste de un paso de inicialización y de un loop.
• En el paso de inicialización, se inicializan las distancias de coste mínimo actualmente conocidas desde u a sus vecinos conectados directamente (D(v) = costo(u,v) para todo vecino v). Las distancias a nodos no directamente conectados se hacen igual a infinito (D(v)=inf si v no es vecino). Se agrega u al conjunto N de nodos visitados.
• En cada iteración, buscamos entre aquellos nodos que todavía no se han añadido al conjunto N y localizamos el nodo que tiene el coste mínimo después de finalizar la iteración previa (w tal que D(w) sea mínimo). Se agrega w a N. Se actualiza D(v) para cada vecino v de w que no pertenezca a N (D(v) = min(D(v), D(w)+ c(w,v))

Pregunta 5 (10 puntos)

V1
a) Describa el concepto de Sistema Autónomo (AS) e indique como se identifican.
b) Describa las principales motivaciones de tener enrutamiento Intra-AS e Inter-AS.
c) Para el caso del protocolo Inter-AS BGP, explique el procedimiento que utiliza para seleccionar la mejor ruta en caso de tener opciones para un cierto prefijo.

a) Un sistema autónomo puede verse como un conjunto de routers que se encuentran bajo el mismo control administrativo. Cada sistema autónomo se identifica mediante su número de sistema autónomo, que es único globalmente.

b) Para esto hay tres grandes lineamientos, el primero es un tema de escala ya que el enrutamiento jerárquico reduce el tamaño de las tablas y de información que debe propagarse en la red para mantener actualizadas las tablas de los nodos pertenecientes a la red.
Un tema importante es el tema de performance, ya que los algoritmos Intra-As están enfocados en términos de performance de la red (Mensajes, algoritmos) mientras que los Inter-AS se basan más en políticas los cuales tienen un mayor peso que la performance de la red.
Otra motivación es el tema político ya que los Inter-AS los administradores controlan cómo se debe enrutar su tráfico, e indican quienes pueden utilizar su sistema autónomo como camino a otro

destino, mientras que en los Intra-AS la administración es única y no es requerido políticas externas.

c) Se presenta el procedimiento descrito por Kurose (mas simple que el especificado en la RFC correspondiente:
- Se elige la ruta con preferencia local mas alta.
- Si hay mas de una, el siguiente criterio es la que tenga el AS-PATH mas corto
- Si hay mas de una, el siguiente criterio es utilizar el enrutamiento de papa caliente.
- Por último, se utiliza el identificador BGP (por ejemplo el menor)

V2
Considere el protocolo BGP,
a) Describa para que se utiliza y los principales servicios que brinda.
b) Mencione y explique cuales son los tipos de mensajes y su principal función.
c) ¿Cual es protocolo de capa de transporte que utiliza para sus mensajes? ¿Utilizar un protocolo de la Capa de Transporte para enviar mensajes de control de la Capa de Red implica una contradicción conceptual o un no respeto del Modelo de Capas seguido en el curso? Justifique.

a) BGP es un protocolo de enrutamiento entre sistemas autónomos. BGP proporciona a cada router mecanismos para obtener de los AS vecinos información acerca de la alcanzabilidad de los prefijos y determinar las mejores rutas hacia esos prefijos.
En particular, BGP permite a cada subred de un AS anunciar su existencia al resto de Internet.
b) Open: Establece una conexión TCP con un par BGP y auténtica al emisor BGP.
Update: Notifica de un cambio en un path, ya sea nuevo o una caída del mismo.
Keepalive: Mantienen la conexión abierta en la ausencia de Updates
Notification: Reporta errores en previos mensajes, también se puede utilizar para el cierre de conexión.
c) BGP utiliza TCP para las conexiones a sus pares.
No, no implica una contradicción ni un no respecto del Modelo de Capas. Mientras que la Capa de Red es la capa donde actúa el protocolo, la Capa de Transporte se utiliza en la implementación de la “aplicación” bgp (o el demonio “bgpd” si lo visualizamos en el contexto de Quagga por ejemplo) y qué servicio utiliza para intercambiar sus mensajes, de acuerdo a las decisiones de diseño que tomaron oportunamente. En tal sentido, se consideró necesario disponer de los servicios de TCP, mientras en otros casos, como por ejemplo OSPF, se decidió que sus mensajes fueran carga útil de paquetes IP. En todos los casos, las decisiones están fuertemente condicionadas por sus principios de funcionamiento.

V3
Describa el funcionamiento de un algoritmo de enrutamiento de tipo Vector de Distancias (Distance Vector) y de uno de tipo Estado de Enlace (Link State). Compárelos e indique las diferencias entre ambos algoritmos.

Los algoritmos de tipo link-state calculan la ruta de menor costo de un nodo a todos los otros nodos de la red utilizando el algoritmo de Dijkstra. Para esto el algoritmo necesita conocer toda la

topología de la red. A partir de los caminos de menor costo se puede computar la tabla de forwarding del nodo.
En cambio, en un algoritmo de tipo Distance Vector, los nodos calculan la distancia a los nodos de la red a partir de los vectores de distancia que le envían sus vecinos. A partir de la información de sus vecinos y el costo del enlace a cada vecino el nodo puede calcular la distancia a los demás nodos de la red. Para este cálculo se utiliza la ecuación de Bellman-Ford en la cual la distancia a un nodo x es el mínimo entre el costo a cada vecino mas la distancia de ese vecino a z.
La principal diferencia entre ambos algoritmos es que los algoritmos link state deben conocer la topología en su totalidad, mientras que los algoritmos distance vector solo la información local del nodo más la de los vecinos. En el primer caso cada nodo difunde a toda la red los costos a sus vecinos mientras que en DV cada nodo solo transmite su vector distancia a sus vecinos.

Pregunta 6 (10 puntos)

V1
Mencione los principales aportes que podría significar la incorporación de la tecnología MPLS a una red ya implementada con el stack TCP/IP. Justifique.

MPLS es una tecnología de Capa de Enlace que permite:
• Disponer de una red de circuitos virtuales por donde las unidades de datos viajan en función del valor de etiquetas que se encuentran en el Encabezado MPLS (ubicado entre el encabezado de Capa de Enlace y el Encabezado de Capa de Red)
• Conmutar las unidades de datos utilizando para ello el valor de la etiqueta contenida en el Encabezado MPLS; inicialmente (hace ya varios años) ello permitió lograr una conmutación más rápida en comparación con la “tradicional” basada en la dirección IP destino del paquete involucrado
• Contar con algunas configuraciones asociadas a la ingeniería de tráfico donde la decisión de hacia dónde enviar determinado paquete no sólo por la dirección IP destino (como ocurre en las redes IP tradicionales), sino que también se puede considerar, la dirección de IP origen y la interfaz de entrada, entre otras opciones.
• Implementar Redes Privadas Virtuales (VPN). Mediante MPLS, un ISP por ejemplo, puede interconectar redes privadas de diferentes clientes a través de su red pública, y de manera transparente para ellos.

V2
a) Explique las diferencias, a nivel de su principal objetivo, entre la Capa de Red y la Capa de Enlace.

b) ¿Cual es el problema que se genera en los enlaces de difusión compartidos (múltiples nodos emisores y receptores, todos conectados al mismo y único canal)?. Nombre las tres categorías de protocolos que existen para mitigar este problema, y describa brevemente uno de ellos.

a) El servicio básico de cualquier capa de enlace es mover un datagrama desde un nodo hasta otro adyacente a través de un único enlace de comunicaciones, mientras que la capa de red tiene por

objetivo mover datagramas entre dos sistemas terminales (hosts) lo que puede implicar atravesar muchos enlaces que componen el camino de origen a destino.

b) Lo que se conoce con el nombre de problema de acceso múltiple: cómo coordinar el acceso de múltiples nodos emisores y receptores a un canal de difusión compartido de forma de evitar o mitigar las colisiones. Esto se logra mediante protocolos (denominados protocolos de acceso múltiple) los cuales se encargan de regular las transmisiones de los nodos hacia el canal de difusión compartido. Puesto que todos los nodos son capaces de transmitir tramas, podría darse el caso de que más de dos nodos transmitieran tramas al mismo tiempo. Cuando esto sucede, todos los nodos reciben varias tramas simultáneamente; es decir, las tramas transmitidas colisionan en todos los receptores. Por tanto, todas las tramas implicadas en la colisión se pierden y el canal de difusión está desaprovechado durante el intervalo de colisión.

Categorías: protocolos de particionamiento del canal, protocolos de acceso aleatorio y protocolos de toma de turnos.
Protocolos de particionamiento del canal multiplexación por división en el tiempo (TDM) y la multiplexación por división de frecuencia (FDM). TDM divide el tiempo en marcos temporales y luego subdivide cada marco temporal en N particiones de tiempo. Cada partición de tiempo se asigna entonces a uno de los N nodos. Cada vez que un nodo tenga un paquete para enviar, transmite los bits del paquete durante su partición de tiempo asignada, dentro del marco TDM que se repite de forma cíclica. TDM resulta muy atractivo porque elimina las colisiones y es perfectamente equitativo: cada nodo obtiene una tasa de transmisión dedicada igual a R/N bps durante cada marco temporal. Inconvenientes: cada nodo está limitado a una tasa promedio de R/N bps aunque sea el único nodo que tiene paquetes para transmitir. El segundo inconveniente es que un nodo siempre tiene que esperar a que le llegue el turno dentro de la secuencia de transmisión; aún cuando sea el único nodo que tenga una trama que enviar.

V3
a) Explique las diferencias, a nivel de su principal objetivo, entre la Capa de Red y la Capa de Enlace.

b) ¿Qué son, para qué se utilizan y cual es el principio de funcionamiento de las técnicas de detección y corrección de errores en capa de enlace?

c) ¿Es posible garantizar la confiabilidad de una transferencia de datos entre hosts con la utilización de las técnicas anteriores? Justifique.

a) El servicio básico de cualquier capa de enlace es mover un datagrama desde un nodo hasta otro adyacente a través de un único enlace de comunicaciones, mientras que la capa de red tiene por objetivo mover datagramas entre dos sistemas terminales (hosts) lo que puede implicar atravesar muchos enlaces que componen el camino de origen a destino.

b) La capa de enlace en general ofrece el servicio de detección y corrección de errores en su hardware. El objetivo de este servicio es detectar (y corregir si es posible) errores de bit introducidos debido a la atenuación de las señales y al ruido electromagnético durante la

transmisión por el medio. Esto se lleva a cabo haciendo que el nodo transmisor incluya bits de detección de errores en la trama y que el nodo receptor realice una comprobación de errores.

c) No es posible garantizar la confiabilidad extremo a extremo dado que existen otras fuentes de error o pérdida mas allá de los errores de bit en los enlaces. Por ejemplo, un router puede decidir descartar un paquete si sus buffers están llenos.

V4

a) Se dice que los switches de capa de enlace son transparentes para los hosts y routers de la subred. Explique la razón por la que se los llama “transparentes”.

b) Argumente (a favor o en contra) de la siguiente afirmación: "*cuando llega una trama a un switch cuya dirección de destino es desconocida, el switch realiza un ARP y luego completa su tabla de conmutación con los datos de la respuesta*".

c) Suponga que tiene 2 switches con soporte para VLANs y se quieren configurar 3 VLANs en ellos, de forma que sea posible que en ambos switches puedan existir usuarios conectados en cualquiera de las 3. Explique como lo haría.

a) La función de un switch es recibir las tramas de la capa de enlace entrantes y reenviarlas a los enlaces de salida. El propio switch es transparente para los hosts y los routers de la subred; es decir, un host/router dirige una trama a otro host/router (en lugar de dirigirla al switch) y la envía a la red LAN, sin ser consciente de que un switch recibirá la trama y la reenviará.

b) Los switches de capa de enlace no manejan el protocolo ARP, ni tampoco la tabla ARP. La tabla que sí manejan es la de conmutación, pero la misma se auto-completa mediante un mecanismo de auto-aprendizaje, y no mediante un protocolo propiamente dicho.

c) Se deberán definir 3 VLANs en cada switch utilizando la misma identificación. Para lograr que el tráfico de las VLANs pase de un switch al otro, se deberá utilizar troncalización VLAN (VLAN Trunking), mediante el uso de algún protocolo de señalización como por ejemplo 802.1Q. Cuando los datos atraviesan el trunk es necesario etiquetar las tramas utilizando la VLAN ID correcta, de manera de no mezclar tráfico de una VLAN con otra. Finalmente sería necesario asignar qué puertos de cada switch pertenecen a qué VLAN.

V5

Considere la topología de la figura.

La Computadora 1 obtendrá su configuración de red vía DHCP. El Servidor 1 ya tiene su configuración de red.

Si el primer tráfico que se observará en la red es todo el motivado por ejecutar el siguiente comando en la Computadora 1: *ping 10.10.1.254*, describa la secuencia de tramas que se observarán en la interfaz de red de la Computadora 1.

La descripción de cada trama deberá contener información de Capa de Enlace de Datos, de Capa de Red y toda otra que considere relevante.

En la situación planteada, ocurrirán 3 eventos principales que se ordenan en el tiempo de la siguiente forma: primero la Computadora 1 obtiene su configuración de red (mediante el protocolo DHCP), segundo, y luego de determinar que debe enviar ciertos mensajes ICMP (como carga útil de paquetes IP) al Servidor 1 (pues a partir de analizar la dirección IP destino, determina que está en su mismo segmento de red), debe conocer la dirección MAC de la tarjeta de red de éste (mediante el protocolo ARP) y tercero y finalmente, enviar y recibir los mensajes del protocolo ICMP asociados al comando ping ejecutado.
Para cada mensaje que se detallará a continuación, se indica si es entrante (in) a la tarjeta de red de la Computadora 1 o saliente (out).
Los mensajes que se observarían son (para cada uno, se agrega la información relevante solicitada):

Respecto a DHCP **(esta parte no se exige en la corrección al no quedar claro que era requerido)**
DHCP Discover (out) – por tratarse de un broadcast el mismo no avanza más allá del router, por lo tanto, no es recibido por el Servidor DHCP 2
Capa de Enlace de Datos
MAC Origen: MAC_Computadora1, MAC Destino: FF:FF:FF:FF:FF:FF
Capa de Red
IP Origen: 0.0.0.0, IP Destino: 255.255.255.255
Capa de Transporte
UDP - Puerto Origen: 68, Puerto Destino: 67
Carga útil: descubrimiento de servidores DHCP para solicitar información de configuración de red

DHCP Offer (in) – sólo lo envía el Servidor DHCP 1, por lo expresado para el mensaje anterior
Capa de Enlace de Datos
MAC Origen: MAC Servidor DHCP 1, MAC Destino: MAC_Computadora1
Capa de Red
IP Origen: 10.10.1.254, IP Destino: 10.10.1.100
Capa de Transporte
UDP - Puerto Origen: 67, Puerto Destino: 68
Carga útil: información de configuración de red ofrecida vía DHCP (especialmente dir IP) y condiciones del servicio.

DHCP Request (out)
Capa de Enlace de Datos
MAC Origen: MAC_Computadora1, MAC Destino: FF:FF:FF:FF:FF:FF
Capa de Red
IP Origen: 0.0.0.0, IP Destino: 255.255.255.255
Capa de Transporte
UDP - Puerto Origen: 68, Puerto Destino: 67
Carga útil: información de selección de configuración de red por el cliente, vía DHCP. Incluye la dir IP del servidor seleccionado y la dir IP seleccionada. Puede incluir solicitud de información de DNS y NTP.

DHCP Ack (in)
Capa de Enlace de Datos

MAC Origen: MAC Servidor DHCP 1, MAC Destino: MAC_Computadora1
Capa de Red
IP Origen: 10.10.1.254, IP Destino: 10.10.1.100
Capa de Transporte
UDP - Puerto Origen: 67, Puerto Destino: 68
Carga útil: confirmación de la aceptación por parte del servidor seleccionado por el cliente y eventualmente, otra información de red solicitada por el cliente.

Respecto a ARP,
ARP Request (out)
Capa de Enlace de Datos
MAC Origen: MAC_Computadora1, MAC Destino: Broadcast (FF:FF:FF:FF:FF:FF)
Carga útil
MAC Origen: MAC_Computadora1, IP Origen: 10.10.1.100
MAC Destino: 00:00:00:00:00:00, IP Destino: 10.10.1.254

ARP Response (in)
Capa de Enlace de Datos
MAC Origen: MAC_Servidor1, MAC Destino: MAC_Computadora1
Carga útil
MAC Origen: MAC_Servidor1, IP Origen: 10.10.1.254
MAC Destino: MAC_Computadora1, IP Destino: 10.10.1.100

Respecto al comando ping (suponiendo que la ejecución del comando ping implica el envío de 3 mensajes ICMP del tipo “echo”),

ICMP Echo (request) (out)
Capa de Enlace de Datos
MAC Origen: MAC_Computadora1, MAC Destino: MAC_Servidor1
Capa de Red
IP Origen: 10.10.1.100, IP Destino: 10.10.1.254
Carga útil
Mensaje ICMP Echo (request) – Tipo 8

ICMP Echo Response (in)
Capa de Enlace de Datos
MAC Origen: MAC_Servidor1, MAC Destino: MAC_Computadora1
Capa de Red
IP Origen: 10.10.1.254, IP Destino: 10.10.1.100
Carga útil
Mensaje ICMP Echo Response – Tipo 0
ICMP Echo (request) (out)
ICMP Echo Response (in)
ICMP Echo (request) (out)
ICMP Echo Response (in)

Redes de Computadoras

Cada uno de 4 últimos mensajes tienen un contenido similar al mostrado para los 2 primeros, respectivamente.