Redes de Datos 2

2º Parcial - 13/07/2024

Solución

Esta es una posible solución a las preguntas planteadas. Por razones didácticas normalmente contiene bastante más información que la mínima necesaria para responder la pregunta.

Pregunta 1 (6 puntos)

En las redes MPLS:

a) Describa el mecanismo de encaminamiento, explique qué son las etiquetas (labels) y cómo se utilizan.
En las redes MPLS el encaminamiento de paquetes se basa en utilizar caminos establecidos lógicamente en la red, de una forma muy similar a las redes de circuitos virtuales.
Las etiquetas de MPLS son valores (números de 20 bits) que identifican los caminos lógicos en la red. Estas etiquetas tienen significado local, es decir que un determinado valor solo tiene significado para el equipo que lo asignó a una FEC (ver parte (c)) y el o los equipos que deban enviarle paquetes destinados a esa FEC.
También vimos que informalmente se llama etiqueta al encabezado del protocolo MPLS aunque su nombre correcto es "shim header".
El primer enrutador MPLS en el camino del paquete asigna la o las etiquetas necesarias para el encaminamiento del paquete. Los demás enrutadores del camino utilizan la etiqueta externa para encaminar el paquete, en general cambiando la etiqueta (swap) por la que corresponda al enlace de salida, pudiendo además quitar o agregar etiquetas (pull/push) según la información en su tabla de encaminamiento. El último enrutador (el LER de salida) quitará la o las etiquetas y encaminará el paquete de acuerdo al protocolo del paquete contenido.

b) Describa el concepto de stack de etiquetas.
En MPLS no estamos restringidos a asignar una única etiqueta a cada paquete, sino que en general a cada paquete se le asocia un stack de etiquetas, compuesto por uno o más valores de etiqueta. Los distintos niveles se ordenan en una pila (stack), y las decisiones se toman utilizando la etiqueta externa (la más cercana a capa 2 en la representación del stack). Solo al retirar (pop) una etiqueta se pasa a procesar el siguiente nivel.
Un bit en el encabezado indica si es la última etiqueta del stack o si hay más etiquetas.

c) Explique el concepto de FEC (Forwarding Equivalence Class, o Clase de equivalencia de encaminamiento)
Los posibles paquetes a encaminar en una red MPLS se clasifican en distintas FEC. Una FEC es el conjunto de posibles paquetes que serán encaminados de la misma manera. A cada una de estas FEC se les asocia una etiqueta, o en general un stack de etiquetas, que podemos ver como un identificador del comportamiento que la red debe tener con los paquetes que se clasifican en dicha FEC.

d) Para el caso particular de LDP, explique cómo se definen las FEC, y cómo se distribuyen.
En la utilización básica (tradicional) de LDP se dividen los paquetes en FECs de acuerdo a las rutas existentes en la tabla de encaminamiento del enrutador, provenientes del o los protocolos de enrutamiento IP internos (incluyendo aquellas rutas estáticas o aprendidas de redes directamente conectadas). A cada una de estas entradas en la tabla de rutas le asigna un valor de etiqueta, el cual es propagado a los vecinos LDP.
LDP descubre a sus vecinos directamente conectados utilizando paquetes de Hello, y establece una sesión TCP con cada uno de ellos para el intercambio de información de etiquetas. Es utilizando estas sesiones que se realiza la distribución de la información de la relación FEC- etiqueta, mediante mensajes de Label Mapping (anuncio de relación FEC-etiqueta) y Label Withdraw (anuncia que un mapeo dejó de ser válido).
El receptor utiliza estas relaciones, junto con la información de su propia tabla de rutas, para completar la tabla de encaminamiento.
(No se esperaba este párrafo en la respuesta a esta pregunta en el parcial) LDP es capaz de transportar mapeos de FECs arbitrarias, y también entre vecinos no directamente conectados (requiriendo algún mecanismo de descubrimiento o configuración manual). Vimos su uso en la aplicación de VPNs capa 2 sobre MPLS, donde las FEC corresponden a los paquetes del protocolo transportado que deben ser encaminadas por un determinado pseudocable (pseudo wire)

Pregunta 2 (6 puntos)

Para los servicios de VPN sobre MPLS vistos en el curso (VPNs capa 2 y capa 3):

a) Explique por qué se utiliza un stack de al menos dos etiquetas MPLS, explicando el significado de cada nivel de etiquetas.

Para el transporte de los servicios de VPNs sobre MPLS vistos en el curso se utiliza un stack de al menos dos etiquetas MPLS. Esto es para separar funciones, lograr escalabilidad, y evitar que todos los enrutadores del camino deban participar del establecimiento de los caminos para cada VPN.
La o las etiqueta(s) externa(s) (etiqueta de transporte, o IGP Label) se utiliza para el encaminamiento del paquete desde el enrutador PE de entrada hasta el PE de salida. Esta puede haber sido aprendida por LDP, por ingeniería de tráfico, por un mecanismo de Segment Routing, o en general por cualquier mecanismo que establezca un camino switcheado entre los enrutadores PE.
La etiqueta interna, o etiqueta de VPN, identifica el servicio al cual pertenece el paquete (por ejemplo la VRF en el caso de VPNs capa 3) y es utilizada por el PE de salida para encaminar el paquete utilizando el servicio adecuado.

b) ¿Qué protocolos se utilizan para distribuir las etiquetas correspondientes al servicio del cliente en cada tipo de VPN?

En las VPNs vistas en el curso se utilizan:
- BGP (con las extensiones multiprotocolo) para la distribución de relaciones FEC-etiqueta en las VPNs capa 3
- LDP (dirigido) para la distribución de relaciones FEC-etiqueta en las VPNs capa 2

c) En el caso de las VPNs capa 3 sobre MPLS vistas en el curso, ¿Qué atributos se utilizan para definir las VRFs que deben importar cada ruta? ¿Cómo se utiliza dicho atributo?

En las VPNs capa 3 vistas en el curso para distribuir el mapeo de etiquetas de servicio utilizando BGP multiprotocolo se utiliza una nueva familia de direcciones (VPNv4 o VPNv6 según corresponda), y atributos de tipo comunidad extendida de tipo Route Target para la importación y exportación de rutas entre los enrutadores PE (también se utiliza un atributo comunidad extendida de tipo Site of Origin, SOO, para evitar loops).
El atributo Route Target (RT de aquí en mas) tiene más de un formato posible (ASN + valor, IPv4 + valor, ASN32 + valor), pero independientemente del formato elegido es utilizado para instruir al receptor acerca de las VRFs que deben importar la ruta que acompaña dicho RT. En cada enrutador PE los anuncios BGP correspondientes a prefijos de cualquier VRF son acompañados por la lista de RT que la VRF tenga configurado para la exportación de rutas. En los enrutadores PE que reciben dichos anuncios la lista de RT que acompaña cada anuncio es comparada con los RT que importa cada una de las VRF, si algún RT coincide la ruta es importada en dicha VRF.

Pregunta 3 (4 puntos)

Para enrutamiento por segmentos (SR, Segment Routing) utilizando MPLS:

a) Durante el curso hablamos principalmente de dos tipos de identificadores de segmento (SID, Segment Identifiers): los SIDs de Nodo y los SIDs de adyacencia.
¿Qué representa cada uno de ellos? ¿Qué es un SID global?

Observación: al comienzo del parcial se aclaró que en lugar de "SIDs de Nodo" quisimos poner "SIDs de prefijo", siendo los primeros un caso particular de SIDs de Prefijo. Se toman como válidas las descripciones de cualquiera de los dos tipos de SID indicados.
En SR se definen distintos tipos de segmento. A cada segmento se le asigna un identificador (Segment IDentifier), que en el caso de MPLS se representa en los paquetes por el valor de una etiqueta MPLS. Para definir un camino se concatenan uno o más segmentos.
Los SIDs de tipo prefijo se asocian a un prefijo IP y representan el mejor camino en el protocolo interno para alcanzar el prefijo indicado. Usualmente los identificadores correspondientes son de tipo global. En particular cuando el prefijo corresponde con una IP de loopback de un nodo (IP/32) se denomina semento de tipo Nodo.
Los segmentos de tipo adyacencia representan una conexión entre dos nodos, permitiendo encaminar paquetes por una interfaz específica del nodo que anuncia dicha adyacencia.
Que un identificador (SID) sea global significa que todos los enrutadores del dominio SR reconocen la asociación de ese SID con el segmento correspondiente. Para ello en el dominio SR se define un bloque de etiquetas reservado para uso global, y se propaga el SID como el offset dentro del bloque de etiquetas reservado.

b) ¿Cómo se distribuye la información de los SIDs disponibles en la red?

La información de asociación de los SIDs con los segmentos correspondientes se distribuye utilizando los protocolos de enrutamiento ya conocidos, con extensiones para propagar la información de SR. Por ejemplo para los segmentos internos (los vistos principalmente en clase) se utilizan extensiones a los protocolos OSPF e IS-IS que permiten transportar dicha información. En el caso de OSPF se transporta la información en LSA de tipo opaco, mientras que en IS-IS se utilizan AVPs adecuados.

Otra opción es que un controlador externo (SDN) programe las asociaciones SID – segmento en los nodos de la red

c) ¿Qué equipo o equipos definen el camino que seguirá un paquete en la red SR y cómo lo hace(n)?
El camino que tomará cada paquete en la red SR es determinado por el LER de entrada a la red SR, ya sea mediante políticas configuradas localmente o mediante la consulta a un elemento centralizado. El camino está representado por una secuencia de SIDs, que se traducen en un stack de etiquetas MPLS que se impone al paquete. Los otros nodos de la red solamente aplicarán las reglas correspondientes al SID representado por la etiqueta exterior para encaminar el paquete.

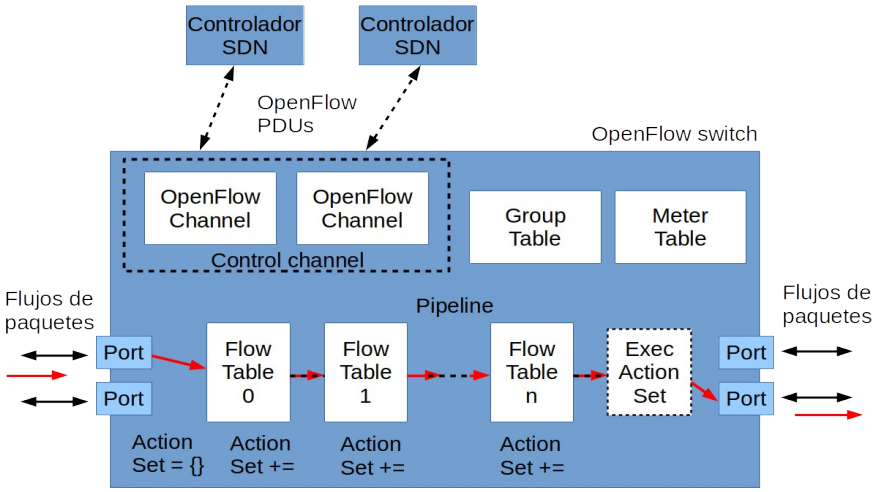
Pregunta 4 (5 puntos)

En la arquitectura SDN vista en el curso:

a) Explique por qué es necesario definir una arquitectura genérica de los switches.

Una de las ventajas de la arquitectura SDN es poder gestionar equipos de diferentes fabricantes y para eso es necesario poder identificar componentes dentro del switch de forma genérica. El controlador debe poder escribir y borrar entradas en la tabla X del switch y para eso es necesario que todos los switches tengan la misma arquitectura independientemente de cómo sea luego la implementación de cada uno internamente.

Openflow en particular define una arquitectura lógica de switches como se muestra en la siguiente figura:



b) Explique qué son las tablas de flujo de los switches Openflow y qué ventajas aporta la existencia de una potencial cadena de tablas.

Una tabla de flujo (flow table) en un switch OpenFlow, asocia cada paquete entrante a un flujo particular y especifica qué acciones se deben hacer con él (ver parte c).
La potencialidad de disponer de una cadena de tablas (pipeline) en un switch ofrece mayor flexibilidad para aplicar acciones a un paquete.
Los paquetes pueden recorrer una o más tablas de flujo una vez que ingresan al switch y en cada una de ellas, dependiendo de los "matches" en las entradas de cada tabla y las instrucciones que indiquen las entradas, se establecerán un conjunto de acciones a aplicar al paquete. Si el paquete no es descartado o enviado al controlador durante el procesamiento, será finalmente enviado por alguno de los puertos de salida del switch.
En la cadena de tablas de flujo se pueden identificar el procesamiento de ingreso (ingress processing) con una serie de tablas que se pueden aplicar antes de definir la interfaz de salida y el procesamiento de egreso (egress processing) que puede también incluir varias tablas y se ejecuta en el contexto del puerto de salida.

c) Explique la función de los principales campos que contienen las entradas de las tablas de flujo (flow entries).

Las tablas están compuestas por entradas, y cada entrada tiene:

- *Match Fields: criterios de selección de paquetes en base a campos de los encabezados (de protocolos de capa 2, 3, 4)*
- *Instructions: define qué hacer cuando hay un match (por ejemplo enviarlo a otra table, agregar acciones al instruction-set, aplicar acciones, actualizar metadata asociada al paquete)*
- *Counters: Cuenta de paquetes que matchean la entrada.*
- *Timeouts: temporizadores que eliminan la entrada por inactividad o al cabo de un tiempo (idle_timeout, hard_timeout)*
- *Cookie: identificador usado por el controlador para actualizar o borrar una entrada*
- *Priority: para ordenar las entradas en la tabla*
- *Flags: que permiten alterar el procesamiento estándar*

Pregunta 5 (6 puntos)

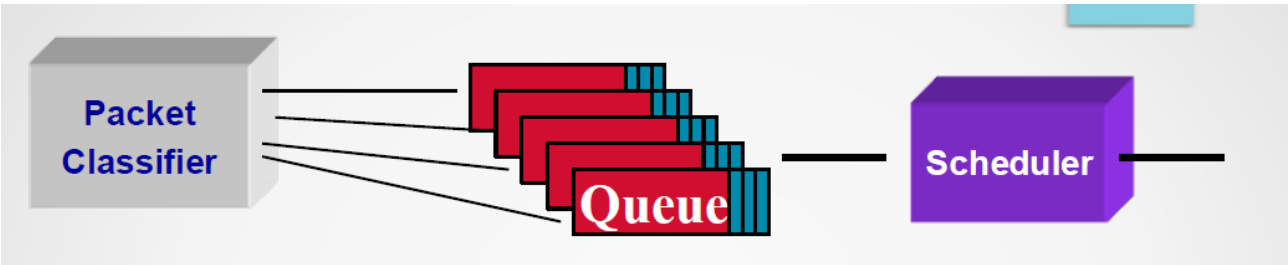
Bajo el modelo de QoS DiffServ:

a) Explique el funcionamiento de los bloques de Classifier, Queues y Scheduler en un router interior.

b) ¿Por qué es necesario implementar estos tres bloques para garantizar los objetivos de QoS? Justifique su respuesta

c) ¿Para qué sirve la marca en los DSCP bits del encabezado IP?

a) En el modelo DiffServ los nodos internos asumen que el tráfico ya fue correctamente marcado y controlado en los nodos externos, de acuerdo al contrato de tráfico, por lo cual lo único que deben hacer es identificar la marca en el paquete, asignar el paquete a la cola adecuada, y luego el scheduler impone las reglas de priorización de tráfico entre las colas.



*El bloque **classifier** revisa las marcas del tráfico en cada paquete, y de acuerdo a la asignación que tenga el paquete, lo asigna a una determinada **queue** o cola de procesamiento de paquetes.*

*El **scheduler** o despachador de paquetes, dispone de una política o estrategia de en que orden recorrer las diferentes queue, impone como distribuir la atención entre las diferentes colas, permitiendo implementar un comportamiento de prioridad estricta entre las clases, garantizar un mínimo de servicio a cada clase, o colas con bajo delay.*

b) El bloque que efectivamente implementa o garantiza la priorización del tráfico, decidiendo que paquete de que queue debe salir primero, es el scheduler. Priorizando el tráfico es lo que nos permite "garantizar" QoS, al menos para las clases de mayor prioridad.

Para poder priorizar, se requiere poder definir varias queue (al menos dos), de lo contrario no hay forma de priorizar. Para poder asignar a que queue irá el paquete, lo cual luego define con que prioridad se envía, es necesario revisar la marca en el paquete y asignarlo a la queue que corresponda.

La pregunta solo consideraba nodos internos, hay acciones que se pueden implementar en los nodos de ingreso o externos, por ejemplo que cumplan con el contrato de tráfico. Si el agregado del total de tráfico de una clase supera lo asignado por el scheduler, ocurren descartes, al menos en dicha clases y dependiendo del scheduler, como lo sería en strict-priority, podría haber descartes en las colas de menor prioridad también.

***Nota:** Ejemplo de dos estrategias del scheduler son WRR y stric priority*

WRR: *El scheduler garantiza un mínimo de atención a cada una de las colas. Se van recorriendo todas las colas en un orden prefijado asignándole un tiempo de servicio a cada una, y a diferencia del Round Robin sin pesos, el tiempo de servicio puede depender del peso de cada cola (o sea que el mínimo de atención puede depender de la cola). Como ejemplo podemos pensar en 4 colas con distintos pesos, y un intervalo de tiempo (ejemplo 1 segundo), que luego ranuramos (por ejemplo en 10 intervalos de 100 ms, de los cuales asignamos 4 intervalos a la cola con mayor peso, 3 intervalos a la segunda, 2 intervalos a la tercera, y 1 intervalo a la de menor peso). Luego vamos recorriendo las 4 colas, asignando a cada una de ellas los intervalos indicados. En caso de no haber paquetes en una cola se prosigue a a siguiente por lo que no se desperdicia capacidad, y hablamos de garantizar un mínimo de atención ya que en el peor de los casos en este ejemplo cada cola tendrá al menos el 40/30/20/10% de la capacidad cada segundo.*

Strict Priority: *Este esquema aplica prioridad estricta entre las clases, hasta no vaciar de paquetes la cola de prioridad mayor, el scheduler no se fija si hay paquetes en las restantes colas de prioridad menor.*

c) ¿Para qué sirve la marca en los DSCP bits del encabezado IP?

En el modelo Diffserv, la priorización del tráfico se hace por clase, a cada paquete IP se le define (se le marca) una clase utilizando los DSCP bits del encabezado IP. De esta forma las estrategias de QoS se realizan solo considerando la marcas, sin diferenciar a que flujo pertenece.

Esta marca, que se transporta junto con los datos porque es parte del paquete IP, es lo que requieren los nodos interiores. Son revisadas por los bloques classifier, asignado a una queue, que luego es considerada en la estrategia del scheduler.

Nota: *Formalmente Diffserv define un potencial de hasta 64 clases de tráfico diferentes. La filosofía es tratar al tráfico por clase sin importar el flujo específico dentro de la clase.*

Una clase define una forma de tratar el tráfico de dicha clase, que prioridad tiene un paquete IP de dicha clase respecto a otro paquete IP de otra clase.

Por diferentes razones, costos de implementar diferentes queues a grandes velocidades, dificultad de mapear la marca de clase en otras tecnologías como MPLS, Ethernet, etc; o el homogenizar el comportamiento entre equipos diferentes dentro de la red, lo usual es que en nodos internos se dispongan de 8 queue. Lo cual hace que en la práctica no tenga sentido utilizar más de 8 clases.

Pregunta 6 (6 puntos)

En MPLS-TE:

a) Explique el mecanismo de protección de nodo FRR en MPLS-TE.
   i. ¿Qué supuesto se realiza sobre la gestión de etiquetas?
   ii. ¿Cuándo se crean los túneles de protección?
   iii. ¿Cómo se detecta una falla en FRR?
   iv. ¿Qué sucede cuando se detecta la falla?
   v. ¿Qué tipo de fallas que ocurren en la red puede proteger?
   vi. ¿Por qué es necesario que se entere de la falla el extremo inicial del túnel (head-end)?

Nota: Para la explicación puede utilizar un ejemplo de restauración de túnel. Defina la topología y las etiquetas que sean necesarias en su explicación, no repita valores de etiquetas para evitar confusiones.

i. ¿Qué supuesto se realiza sobre la gestión de etiquetas?

El supuesto que se realiza es que las etiquetas tienen significado único al router (espacio de etiquetas local al nodo y no por interfaz), de esta forma, alcanza con llegar al router con el mismo valor de etiqueta, sin importar la interfaz de entrada, y el procesamiento de forwarding es el mismo.

El requerimiento para poder utilizar túneles de respaldo es que debo llegar al MP (Merge Point) con el mismo valor de etiqueta, tanto por el camino principal, como por el de respaldo.

ii. ¿Cuándo se crean los túneles de protección?

Los túneles de protección se crean "a priori" (antes que suceda alguna falla) como túneles normales, luego se declaran en los links que quiero proteger. Al momento de detectar una falla, el nodo PLR (Point of Local Repair) que detecta la falla, es quien decide que túnel de respaldo utilizar y que túneles respaldar. En el caso de protección de nodo, el túnel de respaldo debe finalizar al menos en el Next Next-HOP (siguiente al próximo nodo de acuerdo al camino actual) que oficia de MP.

***Nota:** En una red, en general no alcanza con un solo túnel de protección para resolver todas las fallas de nodos que componen el camino actual.*

iii. ¿Cómo se detecta una falla en FRR?
Las fallas se detectan principalmente por dejar de ver los mensajes de RSVP Hello en los links. El envío de mensajes de Hello periódicos fue un agregado al protocolo RSVP cuando se decidió utilizarlo para ingeniería de tráfico, dado que esperar a la pérdida de soft-state implicaba tiempos de 30 segundos, y el objetivo fue obtener una detección rápida de fallas.

Los Hello no son por sesión de RSVP o por túnel TE que utiliza el link, solo interesa saber que ocurrió una caída del link, pero no importa si hay un solo túnel por el link, o 150 túneles por el link.

Existen otros mecanismos que permiten detectar la caída de un link, por ejemplo BFD (Bidirectional Forwarding Detection), y que permiten acelerar el tiempo de detección.

iv. ¿Qué sucede cuando se detecta la falla?

Cuando se detecta una falla, el nodo que detecta la falla (PLR), para cada túnel de TE que pasa por el link que falló, notifica con un mensaje de RSVP ERROR al nodo Head-End (que inicia el túnel de TE) de cada túnel.
Para los túneles que solicitaron protección, si la misma está disponible, el PLR comienza a enviar los paquetes por el túnel de protección elegido.

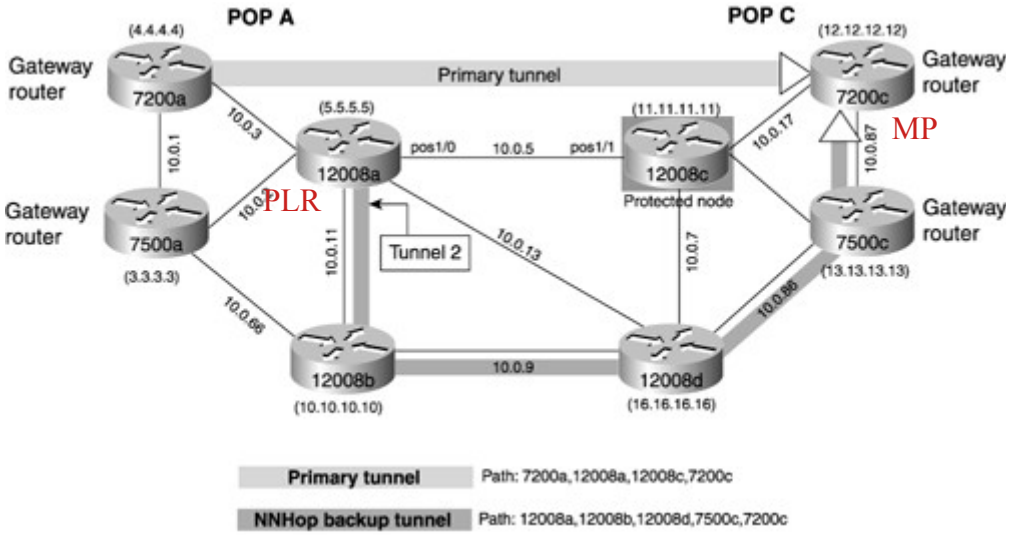
En esta lista de túneles, hay túneles que solicitaron protección y otros túneles que no lo solicitaron.
La solicitud de protección por FRR se define al momento de declarar el túnel en el Head-end, y viaja en los mensajes RSVP PATH ERROR.

Si el túnel no solicitó protección, el Head-End al recibir el RSVP PATH ERROR, inicia un nuevo proceso de búsqueda de un nuevo camino, revisando la topología, aplicando las restricciones de camino y requerimientos de BW del túnel y señalizando por RSVP el nuevo camino. Mientras tanto, el forwarding de tráfico se pasa a realizar por el forwarding clásico de IP o a lo sumo LDP, sin poder garantizar los requerimientos de BW.

Si el túnel solicitó protección, el nodo que detectó la falla percibe la situación, además de la falla, agrega al mensaje de RSVP PATH ERROR la información que la reparación se activó. Esto quiere decir que se comienza a utilizar el túnel de backup hasta nuevo aviso.

El Head-End al recibir el RSVP PATH ERROR, por más que identifica que se está reparando la falla, no puede asumir que el backup puede garantizar el BW del túnel original. Por lo cual el Head-End igual inicia un nuevo proceso de búsqueda de un nuevo camino, revisando la topología, aplicando las restricciones de camino y requerimientos de BW del túnel, y luego señalizando por RSVP el nuevo camino. Recién una vez definido un camino nuevo con garantías, el Head-End conmuta el tráfico para el nuevo túnel.

En caso de no encontrar un nuevo camino que cumpla las restricciones, se continúa utilizando el respaldo.



Recordar que el túnel de respaldo fue señalizado previamente, se dispone de las etiquetas necesarias. Cuando actúa el respaldo en MP, lo primero que se el hace es que etiqueta se modifica para la etiqueta de llegada del túnel primario al PLR, luego se hace un push de la etiqueta del túnel de respaldo (el cual comienza en el PLR y término en el MP).

v. ¿Qué tipo de fallas que ocurren en la red puede proteger?

Las fallas que protege la protección FRR de nodo, son tanto de link o del nodo detrás del link. El túnel de respaldo tiene que estar creado entre el nodo que detecta la falla PLR (Point of Local Repair), y el punto de reparación o MP (Merge Point) es el NNHOP (Next Next HOP).

El respaldo FRR de nodo puede proteger a varios túneles a la vez. La solicitud de protección de túnel se envía al momento de crear los túneles primarios, de esta forma el nodo PLR conoce la lista de túneles que requieren protección.

La cantidad final de túneles protegidos por un túnel de respaldo dependerá de la cantidad de túneles que pasen por el nodo y utilicen el link protegido, la cantidad de túneles que solicitaron respaldo, el tipo de respaldo (ancho de banda), y cuantos túneles de respaldo pueda tener en simultáneo. Este proceso es controlado por el nodo que detecta la falla.

vi. ¿Por qué es necesario que se entere de la falla el extremo inicial del túnel (head-end)?

El objetivo de la protección FRR es ofrecer un respaldo temporal, este respaldo puede ser preservando el ancho de banda reservado por el túnel principal, o no. En caso de preservar ancho de banda, es posible que no alcance para todos los túneles a respaldar. Si el respaldo es sin preservar el ancho de banda, se desconoce el ancho de banda que realmente se dispone mientras actúa el respaldo.

Por este motivo, siempre se busca que llegue el mensaje de RSVP PATH ERROR al Head-End, para que este se entere que hubo una falla y que actúa el respaldo. El Head-End puede entonces buscar un camino (si es que existe ya que cambió la topología) que cumpla con las restricciones del túnel original. En caso de encontrarlo, lo señaliza y luego transfiere el tráfico original al nuevo túnel.

Pregunta 7 (5 puntos)

Dentro del paradigma de NFV:

a) ¿Cuál es la función del hipervisor?

b) ¿A qué se denomina Máquina Virtual (VM)?

c) ¿Qué es una Virtual Network Function (VNF)?

d) Detalle los diferentes elementos que componen la Network Function Virtualized Infraestructure (NFVI).

a) En pocas palabras NFV propone desacoplar las funciones de red tradicionales (Hardware y Software propietario), para que de esta manera las funciones puedan ejecutarse en servidores de propósito general y Entornos Virtualizados (ejemplo Virtual Machine o contenedores).

La estandarización de NFV considera dos posibles tipos de virtualizaciones, virtualizaciones basadas en hipervisores (y VM sobre el hipervisor) o virtualización basada en contenedores.

Los hipervisores permiten realizar una abstracción del HW (servidores), particionar los recursos y controlar el acceso a dichos recursos HW, para luego desplegar VM (máquinas virtuales) las cuales implementan los componentes software necesarios. Desde el punto de vista de una VM requiere un sistema operativo tradicional y es incapaz de distinguir si se esta ejecutando directamente sobre un componente hardware real o sobre componentes virtualizados por el hipervisor. La separación se hace en la separación de las llamadas del kernel al HW.

b) Una máquina virtual VM (Virtual Machine) es una versión software de un computador/servidor, se ejecuta sobre un hipervisor que abstrae el HW físico y le asigna una porción del HW a dicha VM.

En un servidor tradicional, se instala un sistema operativo que gestiona el acceso al HW, y luego sobre el sistema operativo se instalan diferentes aplicaciones. Una VM comparte la característica de tener un sistema operativo base, aplicaciones sobre dicho sistema operativo, pero el acceso al HW no es directo, sino mediante la abstracción que realiza el hipervisor.

Siendo más formales, una VM se define como una imagen del sistema (un archivo muy grande que contiene el file system del sistema operativo, aplicaciones, etc), y un archivo de metada que define las cracterísticas del HW virtualizado (memoria, procesamiento, interfaces de red) que se requiere. El hipervisor es quien dada las características del vHW y la imágen del sistema, logra crear la VM, reservando los recursos e iniciando el proceso de arranque o booteo normal de un servidor, el cual una vez finalizado, no es capaz de discernir si es un servidor tradicional o una VM.

c) *Una VNF (Virtual Network Function) es una implementación de una Función de Red tradicional (router, firewall, CGSN, LB, PGW, PCRF, etc) desplegada dentro de un entorno virtualizado NFVI (Network Function Virtualization Infraestructure) de acuerdo al paradigma NFV.*

Originalmente la ETSI utilizó el concepto VNF de forma genérica para cualquier entorno de virtualización, pero las implementaciones fueron basadas VM, cuando se avanzó en la estandarización para el uso de contenedores, se comenzó utilizar la sigla CNF (Containerized Network Function) cuando el esquema de virtualización es basado en contenedores para poder distinguirlo.

d) La capa NFVI es parte del modelo de NFV, la misma contiene los elementos más cercanos al mundo físico, sobre la cual luego se despliegan las VNF/VM (ver Nota).

En grandes rasgos contiene los siguientes elementos:

- *HW:*
  - *Nodos de cómputo (servidores)*
  - *storage (almacenamiento fuera del nodo de cómputo)*
  - *infraestructura de red (switches físicos, switches embebidos, routers)*

- *hipervisor: El cual no solo abstrae los nodos de computo, sino el acceso al storage centralizado.*

- *Equivalentes virtualizados del HW:*
  - *Virtual Computing: equivalente virtual de CPU, memoria*
  - *Virtual Storage: equivalente virtual de almacenamiento centralizado*
  - *Virtual Network: equivalente virtual de una red LAN*

Nota: La relación entre VNF y VM no tiene porque ser uno a uno, formalmente una VNF puede descomponerse en VNFC (CNF Component), y luego cada VNFC si tiene una asociación uno a uno con una VM.