Redes de Datos 2

1º Parcial – 27/04/2023

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 (3 puntos)

a) Explique los cambios que introduce IPv6 respecto a IPv4 en cuanto a la fragmentación de paquetes ¿Qué beneficio brinda la forma en que se implementa en IPv6?

La diferencia esencial es que en IPv6 no se admite la fragmentación en equipos intermedios.
En IPv4 cualquier enrutador puede fragmentar los paquetes si la MTU del siguiente enlace es menor al tamaño del paquete a enviar (a menos que el paquete tenga prendida la bandera DF, don't fragment).
En IPv6 esto no está permitido, si es necesario fragmentar un paquete, debe hacerse en el origen.
La ventaja de esta decisión es que simplifica el procesamiento en los routers de la red (pasando la complejidad a los extremos) y por tanto hace más eficiente el forwarding.

b) Explique el mecanismo "Path MTU Discovery" usado en IPv6 ¿Por qué este mecanismo es particularmente importante en IPv6?

Como se dijo en la parte a), la fragmentación en IPv6 solamente se puede hacer en el origen. Esto implica que, para hacer un uso eficiente de la red, el origen debe tratar de enviar paquetes lo más grandes posible (para que los encabezados pesen menos), pero asegurándose que no serán descartados en equipos intermedios por limitación de la MTU.
El mecanismo "Path MTU Discovery" le permite al equipo origen determinar cuál es el tamaño máximo de paquete que puede utilizar hacia un determinado destino.
El mecanismo consiste en que el equipo comienza enviando un paquete del tamaño máximo que le permita su primer enlace de salida. Si en algún punto de la red, ese paquete no puede seguir porque aparece una limitación de la MTU, el paquete será descartado y el enrutador que lo descarta enviará un mensaje ICMPv6 al nodo origen, indicando el motivo del descarte (MTU excedeed) y el valor de la MTU máxima admitida.
A partir de ese mensaje ICMP, el nodo origen enviará un paquete del tamaño indicado en la respuesta anterior. El procedimiento se repite (bajando el tamaño a los valores indicados en los mensajes ICMP) hasta que el paquete llega al destino.
Una vez determinado el "path MTU", en caso de necesitar fragmentarse un paquete, se realizará utilizando encabezados de extensión específicos.

Pregunta 2 (3 puntos)

a) Explique el mecanismo de auto-configuración de IPv6 llamado "Stateful Address Auto Configuration", indicando brevemente los pasos que realiza un equipo para auto-configurarse.

El mecanismo de auto-configuración stateful es cuando las direcciones IP del son provistas por un servidor DHCP.
Cuando un nodo IPv6 se conecta a una red con este mecanismo de auto-configuración, realizará los siguientes pasos (se asume que en la red hay un enrutador y un servidor o relay DHCP):

i. *El equipo comienza generando una dirección IPv6 link-local. Para esto debe generar un IID (Interface ID) (64 bits) mediante el mecanismo EUI-64 o algún otro método. A ese IID se le agrega el prefijo fe80::/10, predefinido para las direcciones link-local.*

ii. *Una vez creada esa dirección link-local, el equipo deberá verificar su unicidad mediante el mecanismo DAD (Duplicate Address Detection). Este mecanismo consiste en enviar un mensaje ICMPv6 de tipo Neighbor Solicitation (NS) destinado a la dirección Solicited Node correspondiente a la IP link-local autogenerada en el paso i. Como es de esperar que la dirección no esté en uso, no recibirá respuesta a ese NS. Si está en uso, deberá generar otro IID.*
iii. *El equipo procede entonces a configurar esa IP autogenerada en la interfaz.*
iv. *Con esa IP como origen, envía un mensaje ICMP de Router Solicitation (RS) para determinar si hay un router en la red. El destino del mensaje es una IP de multicast predefinida donde escuchan todos los enrutadores.*
v. *Si hay un enrutador, recibirá de el un mensaje ICMP de Router Advertisement (RA) que contendrá las banderas M y O ambas en 1. Esto significa que deberá usarse un servidor DHCP tanto para configurar las direcciones (bandera M) como los otros parámetros (servidores de DNS).*
vi. *El equipo enviará entonces un mensaje DHCP "solicit" a la dirección multicast bien conocida que identifica a todos los servidores o relays DHCP. Este mensaje será respondido con un mensaje DHCP "advertise". Si el equipo desea aceptar la dirección asignada, le enviará al servidor DHCP un mensaje de "request", que será respondido con un mensaje DHCP "reply" por parte del servidor, confirmando la dirección asignada. Además enviará las direcciones de los servidores DNS a utilizar y eventualmente otros parámetros.*
vii. *Verificará por DAD la unicidad de la dirección asignada igual que en ii).*
viii. *Esa dirección asignada se configura en la interfaz del equipo, así como las direcciones de los servidores DNS.*
ix. *Además el equipo necesitará una ruta por defecto la cual configurará usando como próximo salto la dirección link-local del enrutador que le envió el RA en el paso v.*

b) Explique las principales diferencias con el mecanismo equivalente en IPv4.

El DHCP en IPv6 es similar al DHCP en IPv4. En ambos casos hay 4 mensajes involucrados y en ambos casos existe la posibilidad de tener un relay DHCP en caso que el servidor DHCP no esté en la misma LAN. Se utiliza UDP como transporte en ambos casos, con números de puertos diferentes. Los mensajes a los servidores o relays en DHCPv6 se realizan por multicast en vez de broadcast como en DHCPv4.
La principal diferencia entre IPv4 e IPv6 es que en IPv6 el servidor DHCP no envía la ruta por defecto, que debe obtenerse de los RA de los enrutadores.
Otra diferencia es que en DHCPv4 los identificadores de los equipos son las direcciones MAC y en DHCPv6 se utilizan identificadores propios llamados DUID (DHCP Unique ID).

Pregunta 3 (7 puntos)

a) ¿Cómo implementa OSPF el enrutamiento jerárquico utilizando áreas?
¿Qué limitaciones tiene? (cantidad de niveles, topología, restricciones del backbone, etc.)

Como vimos OSPF permite dividir la red en "áreas", una división administrativa de ennrutadores y enlaces. Dentro de un área los enrutadores intercambian información de la topología, es decir los vértices (enrutadores y redes de tránsito) y las aristas que los interconectan (los links), mientras que entre áreas solamente se intercambia información de rutas alcanzables (prefijos internos y externos alcanzables a través de los enrutadores de borde de área y la métrica de alcanzarlos). Los links pertenecen a un área, mientras que los enrutadores pueden estar conectados a varias áreas.
Las áreas se identifican por números. Puede haber tantas áreas como el administrador considere conveniente, pero este enrutamiento jerárquico tiene solo dos niveles. El área número "0" es conocida como backbone, y debe interconectar a todas las demás áreas (un enrutador que pertenezca a más de un área necesariamente debe pertenecer al área 0)

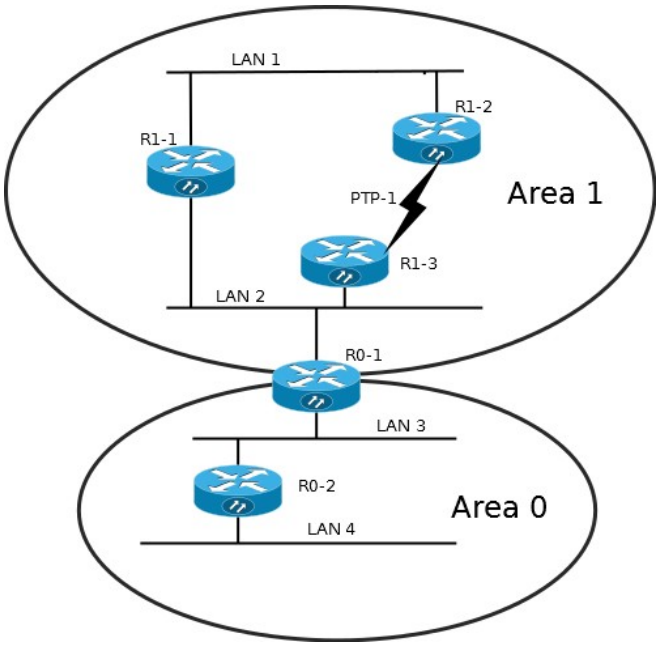
Como cada enrutador solo conoce la información de topología de la/las área/s a la/s que pertenece esto reduce la dificultad del trabajo de cada enrutador.
Otra simplificaciòn, es que el enrutador de borde de área puede ser configurado para realizar sumarización (agregar en uno o varios anuncios la información de varios prefijos del área)
El tráfico entre áreas debe necesariamente pasar por el área 0.
El backbone debe ser conexo (debe haber "una sola área 0")

b) ¿Cómo evita OSPF los loops de información de enrutamiento al utilizar enrutamiento jerárquico?
Dentro de un área todos los enrutadores reciben toda la información de topología, por lo que cada enrutador conoce los caminos posibles y no habrá posibilidad de loops.
En cambio entre áreas solo se propaga la información de destinos alcanzables y la métrica para alcanzarlos. Esta información no es suficiente para saber que no se está eligiendo un camino libre de loops en el caso general. Por ello OSPF restringe la topología entre áreas, solo permitiendo tránsito entre áreas a través del backbone, efectivamente impidiendo los loops entre áreas.

c) Explique las semejanzas y diferencias vistas en clase entre el enrutamiento jerárquico del protocolo OSPF, y el del protocolo IS-IS
Algunas semejanzas vistas en el curso son la cantidad de niveles (dos en ambos protocolos), la existencia de un backbone que interconecta la jerarquía, la posibilidad de realizar sumarización en la frontera de la jerarquía
Respecto a las diferencias vimos que mientras que en OSPF la división jerárquica está dada por las áreas, en IS-IS la jerarquía está dada por el "nivel". Un mismo enrutador sobre el mismo link puede estar intercambiando información de ambos niveles. El Backbone es el conjunto de enrutadores que intercambian información de nivel 2. A diferencia de OSPF, en ISIS los enrutadores siempre pertenecen a una única área, la interconexión de áreas se da en los links

Pregunta 4 (7 puntos)

En la red de la figura se utiliza enrutamiento dinámico con el protocolo OSPF.
Las áreas son las indicadas en la figura. Las redes LAN son todas Gigabit Ethernet, y el enlace punto a punto funciona a 100 Mbps.



a) Asumiendo que es OSPFv2 (OSPF para IPv4), indique todos los LSA que espera ver en el Área 1. Para cada uno indique el tipo de LSA y su contenido (puede indicar con sus propias palabras la información que espera encontrar). No olvide la información aprendida de otras áreas
En el área 1 tenemos 4 enrutadores y dos redes multiacceso de tránsito. Además esperamos recibir LSA de tipo summary con la información de los prefijos externos al área. Entonces esperamos ver:
- LSA tipo router correspondiente a R1-1, con la identificación de R1-1 (router-id), y una lista con dos links de tránsito identificados por el DR de LAN1 y LAN2 respectivamente (cada uno con el costo de utilizar dicho link)
- LSA tipo router correspondiente a R1-2, con la identificación de R1-2 y una lista con dos links, un link de tránsito (identificado con la IP del DR de LAN1) y un link punto a punto (identificado con el router-id de R1-3). Si el link punto a punto está numerado, aparecerá además un link de tipo STUB hacia la red correspondiente

- LSA tipo router correspondiente a R1-3, con la identificación de R1-3 y una lista con 2 links, un link de tránsito identificado con la IP del DR de LAN2 y un link punto a punto identificado con el router-id de R1-2. Si el link punto a punto está numerado, aparecerá además un link de tipo STUB hacia la red correspondiente
- LSA tipo router correspondiente a R0-1, con la identificación de R0-1 y una lista con un único link correspondiente a la red de tránsito (LAN2), identificado por la IP del DR en LAN2
- LSA de tipo network, correspondiente a LAN1, identificado por la IP del enrutador designado en dicha red, y con la lista de los enrutadores (router-id) conectados a LAN1, en este caso R0-1, R1-1, R1-3
- LSA de tipo network, correspondiente a LAN2, identificado por la IP del enrutador designado en dicha red, y con la lista de los enrutadores (router-id) conectados a LAN2, en este caso R1-1, R1,2
- LSA de tipo summary enviado por R0-1 correspondiente al prefijo de LAN3 y el costo que R0-1 tiene para llegar a dicha LAN
- LSA de tipo summary enviado por R0-1 correspondiente al prefijo de LAN4 y el costo que R0-1 tiene para llegar a dicha LAN

b) Realice un esquema del grafo que representa la topología del área 1 en base a la información obtenida de los LSA
DIAGRAMA

c) Si en lugar de OSPFv2 tuviéramos OSPFv3, ¿esperaría ver diferencias en la topología? ¿esperaría ver nuevos LSA, y si es así cuáles?
Respecto a la topología no esperamos ver diferencias, ya que el manejo de topología es igual en ambas versiones del protocolo.
Sin embargo si esperamos ver nuevos LSA, correspondientes a la información de prefijos IPv6 (que en OSPFv3 se separa de la información de topología)
En particular esperamos encontrar:
- LSAs de tipo inter-area-prefix, correspondientes a los prefijos IPv6 presentes en LAN1, LAN2, y el enlace punto a punto.
- LSAs de tipo Link, correspondientes a los prefijos de los vecinos directamente conectados en los links correspondientes

Pregunta 5 (5 puntos)

El protocolo BGP versión 4 fue originalmente diseñado para intercambiar prefijos de direccionamiento IPv4 y utilizar un identificador de AS de 16 bits. Actualmente BGP permite intercambiar prefijos IPv6 y utilizar AS de 32 bits.

a) ¿Qué cambios hubo que realizar en el protocolo para poder utilizar AS de 32 bits?
En particular, explique el caso de dos enrutadores que soportan AS de 32 bits.

b) El intercambio de prefijos IPv6 utiliza las extensiones MP-BGP.
Explique los dos atributos nuevos que se utilizan y justifique su necesidad.

c) Las partes anteriores son ejemplos de extensiones del protocolo BGP, pero ambos vecinos BGP deben de acordar el soporte de estas extensiones.
¿Qué mecanismo utiliza BGP para acordar el soporte de cada extensión de protocolo?

a) Comencemos por el escenario más simple, que es que ambos vecinos BGP soporten AS de 32 bits. Si este es el caso todos los atributos que referencian a números de sistema autónomo deben pasar de 16 bits a 32 bits, esto no genera conflictos porque ambos enrutadores lo implementan. El caso más claro es el AS_PATH, secuencia de AS_PATH segments, y por más que actualmente sea poco utilizado y no recomendado, también ocurre con AGGREGATOR (número de AS que realiza agregación de rutas).

Por consistencia en las sugerencias de como utilizar las comunidades, que originalmente son números de 32 bits, como AS:N_{id}, que posee la ventaja de dejar de forma explícita que la comunidad es definida por el AS. Cuando pasamos de números de AS de 16 bits a 32 bits, se define un nuevo atributo llamado comunidad extendida de largo 64 bits, que permite seguir utilizando la misma interpretación como AS:$N_{id.}$

Resulta un poco mas complejo el escenario de transición, cuando uno de los vecinos solo entiende AS de 16 bits y el otro puede manejar AS de 32 bits; para este caso se definen dos nuevos atributos AS4_PATH y AS4_AGGREGATOR, cuyo objetivo es poder preservar la información que no se puede contemplar en los atributos anteriores. Se utiliza un número de AS especial llamado AS de transición 2345 (AS_TRANS), de forma de que en el AS_PATH se registre el AS_TRANS cada vez que se pase por un AS cuyo número sea de 32 bits, de esta forma se preserva el largo del AS_PATH para la toma de decisión de mejor camino de BGP.

En el atributo AS4_PATH se preserva la secuencia de AS no entendible por las entidades BGP que solo manejan número de AS de 16 bits, de forma de poder reconstruir la información por las entidad BGP que si entienden número de AS de 32 bits (y si lo desean poder realizar políticas de acuerdo a la secuencia de números de AS). El atributo AS4_PATH lo genera el primer router que entiende AS de 32 bits, y dialoga con otro router que solo entiende AS de 16 bits. Una vez que un enrutador agrega ese atributo, el mismo se mantiene en todo el camino, de forma que si luego atravieso varias islas de 16 bits, se preserve la información, en esos casos el AS4_PATH podría tener números de AS menores a 65335 o mayores, pero todos expresados en 32 bits.

Como a priori el enrutador no sabe si el vecino BGP entiende números de AS de 32 bits, si su propio número de AS es de 32 bits no puede indicarlo en el mensaje de OPEN. Esto se resuelve utilizando el valor de AS _TRANS cuando el número de AS es mayor a 65535 (para el campo dentro del mensaje OPEN).
Como todas las nuevas funcionalidades agregadas en BGP estas deben ser anunciadas en el mensaje de OPEN como una capability (support 4-octet AS number capability). Con esto además de declarar el soporte de números de AS de 32 bits, también contiene el verdadero número de AS (de otra forma no podría confirmar mis vecinos o distinguir entre eBGP e iBGP).

b)
Si bien las extensiones Multi Protocolo (MP) en BGP tienen un carácter genérico, se pueden utilizar para varios protocolos, resulta más intuitivo ver cómo se utilizan en IPv6.

El atributo bien conocido mandatorio NEXT-HOP carece de sentido si no es IPv4. En IPv4 para anunciar nuevos prefijos o retirarlos se utiliza el mensaje de UPDATE. Pero dentro del mensaje UPDATE el largo de los prefijos a lo sumo puede ser de 32 bits, no permite adaptaciones.

En BGP es posible adicionar nuevos atributos, para poder realizar las mismas acciones que en IPv4 en IPv6, anunciar prefijos y retirar los anuncios, se definen entonces dos nuevos atributos para dichas acciones ***MP_REACH_NLRI*** *(Multi Protocol Reach Network Layer Routing Information) que "reemplaza" los anuncios dentro del mensaje UPDATE y* ***MP_UNREACH_NLRI*** *(Multi Protocol Unreach Network Layer Routing Information) que "reemplaza" el witdrawn dentro del mensaje UPDATE.*

El nuevo atributo MP_REACH_NLRI contiene el prefijo (NLRI), como dirección de red y largo de prefijo (equivalente a como se realizaba para IPv4 en el UPDATE), entre otro campos también contiene el NEXT-HOP. Para el caso de IPv6 el NEXT-HOP puede ser más de una dirección IP (link-local y una o más globales).

El nuevo atributo MP_UNREACH_NLRI se utiliza para indicar prefijos anunciados previamente que ya no son alcanzables. Si solo se anuncia el mejor camino a un prefijo, alcanza con dar la información del prefijo (NLRI) para retirar el anuncio.

Por su carácter de ser genérico, ambos atributos llevan unos campos que indican a que protocolo se refiere el intercambio de información de ruteo.

c) BGP permite utilizar extensiones del protocolo como las desarrolladas en a) y b), podemos interpretar el mensaje OPEN de BGP como un inicio de conexión entre entidades de ruteo donde, entre otras cosas, intercambiamos las extensiones (las nuevas funcionalidades) que pueden soportar. El soporte de estas extensiones se declara como una nueva capability (campos dentro del mensaje OPEN) una por cada extensión a utilizar. Solo si ambos interlocutores declaran soportar dicha extensión puede ser utilizada en la sesión.

Una vez acordado que ambos han implementado y entienden dichas extensiones, entonces es posible utilizarlas (utilizar los nuevos atributos o el cambio de la interpretación de estos).

Pregunta 6 (5 puntos)

En el curso hemos visto algunos problemas de seguridad que presenta el protocolo BGP.

a) Explique el problema denominado secuestro de prefijo.

b) Explique el problema denominado secuestro de camino.

c) ¿Por qué pueden ocurrir ambos tipos de ataque?

a) Si recordamos que Internet es un conjunto de varios AS que se interrelacionan para intercambiar la información de sus prefijos (sus redes de clientes o servicios) y que puedan realizar intercambios de datos (TCP/UDP).

Un AS cualquiera debería de anunciar sus prefijos (aquellos que fueron gestionados por el RIR y asignados a la misma entidad que le asignaron el número de sistema autónomo).

El problema de secuestro de prefijo ocurre cuando otro AS decide anunciar prefijos que no le corresponden. El AS X anuncia los prefijos N del AS Y . Se llama secuestro de camino porque dichos anuncios N figuran como generados por el AS X, el AS X es el primer elemento del AS_PATH.

Para este problema vimos una solución en el curso, es lo que llamamos el ROA (Root of Authority), pero en el fondo se basa en que el mismo RIR que asigna el número de sistema autónomo asigna el rango de direcciones IP. Por lo que conceptualmente es posible consultarlo y verificar la consistencia, esto se hace fuera de banda, o dicho de otra forma por fuera del protocolo BGP.

b) A diferencia del secuestro de prefijo, en el secuestro de camino el AS X genera anuncios por los prefijos N del AS Y, pero donde el AS_PATH pase por él, manteniendo que el anuncio fue generado por el AS Y.

Para ejemplificar, genera anuncios para los prefijos N donde el AS_PATH sea "AS_X AS_Z AS_Y" o "AS_X AS_Y". De esta forma aun implementando el ROA propuesto en a) no sería posible detectarlo.

Con esto el AS X logra que parte (o todo) del tráfico destinado al AS Y, pase por él, beneficiándose de alguna forma, o bien por cobro de enlaces o porque es posible hacer análisis o ataques de men-in-the-middle.

Este problema es más complejo de resolver, porque la información de los AS-PATH válidos no se encuentra en los RIR, sino que depende de como se interrelacionan los diferentes AS en Internet. Algo sobre el cual no puedo dar garantías.

Validar un camino presenta la complejidad que se deben validar toda la cadena, a modo de ejemplo ilustrativo, para el AS_PATH "AS_X AS_Z AS_Y", que el AS_Z y el AS_Y tienen un sesión eBGP y ambos la declaran, y luego que el AS_Z permite que los anuncios del AS_Y se los envíe al AS_X.

c) En pocas palabras ambos problemas suceden porque no fueron previstos por el protocolo, no existe en el protocolo información que permita poder evitarlos.

El problema de secuestro de prefijo, puede contenerse si los AS de tránsito que ofrecen servicios a pocos AS, controlan la lista de prefijos que pueden anunciar. Pero cuando dos AS de tránsito con varios clientes intercambian información se vuelve complejo el implementar estos filtrados. Por lo que se depende de ese primer control.

En a) mencionamos la propuesta del ROA, que en el fondo conceptualmente se basa en que confiamos en los RIR, quienes asignan los números de AS y los rangos de direcciones IP.
El secuestro de camino en el fondo puede suceder porque el protocolo no tiene forma de verificar el AS_PATH de un anuncio, confía en lo entregado por el vecino y lo propaga. Aquí nuevamente sucede que se podrían realizar controles en los AS de tránsito, deberían hacerse filtros por prefijos y por lista de AS_PATH, pero son difíciles de mantener actualizados y se suelen generar problemas si no están previstos todos los caminos alternativos.Pensemos que si soy un AS de tránsito por el cual pasa información de ruteo de 20.000 sistemas autónomos, es difícil implementar todos los posibles caminos que pueden tener entre estos 20.000 clientes (debo de conocer todos los posibles caminos) sin que existan errores en las políticas. Sin contar con escenarios donde los gobiernos podrían estar involucrados, a modo de ejemplo, son varias las noticias de hackeo desde Rusia.

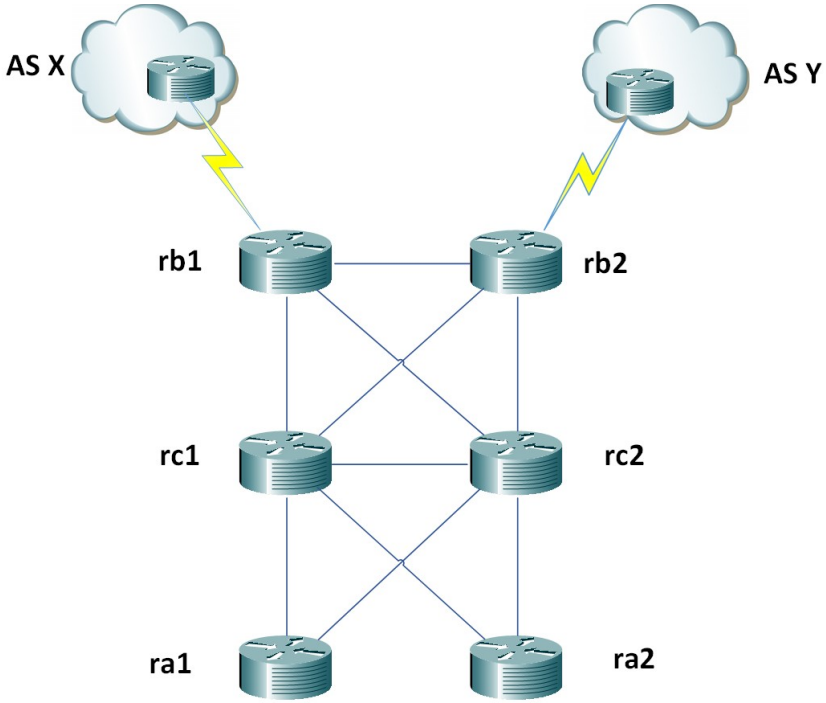
Pregunta 7 (7 puntos)

a) Para poder evitar loops internos iBGP no permite anunciar a otro vecino iBGP lo aprendido por una sesión iBGP. Asumiendo que la sincronización está apagada dentro del AS. ¿Qué consecuencias tiene dicha regla y que soluciones existen?

b) Se decide implementar reflectores de rutas dentro de un sistema autónomo de tránsito, el cual cuenta con 6 enrutadores, dos de ellos hablan eBGP (rb1 y rb2) con otros AS, dos de ellos son equipos de core (rc1 y rc2) y los restantes dos equipos que realizan agregación (ra1 y ra2).

Se decide definir a los router rc1 y rc2 como reflectores de rutas para ra1 y ra2, pero no para rb1 y rb2.

Considerando el diagrama de la figura, donde figuran los routers y los links físicos entre routers, y que se desea que la información de routeo se pueda propagar por iBGP. Realice un diagrama de todas las sesiones iBGP necesarias, indicando cuáles de ellas son iBGP normales y cuáles son sesiones de reflectores.



Tener presente que cada router debe tener al menos dos sesiones a distintos enrutadores para garantizar la alta disponibilidad (la caída de un router, no debería discontinuar el servicio).

a) Si se desea que todos los router dispongan de la información de ruteo aprendida por eBGP en el AS, la única alternativa es realizar una sesión iBGP entre cada par de routers. De esta forma lo aprendido por eBGP, se lo enseño a todos de primera mano por iBGP, lo aprendido por iBGP, lo puedo anunciar por eBGP.

*Pero el full-mesh de iBGP tiene un problema de escala cuando hay muchos vecinos iBGP, por ejemplo si ya hay **n** routers, agregar un router más, implica ir a cada uno de los **n** router y configurar una sesión iBGP con el nuevo router.*

La alternativa clásica es utilizar reflectores de rutas (RR), o jerarquías de reflectores de rutas cuando el número de enrutadores es muy alto. Los RR permiten definir un conjunto de vecinos como clientes de reflector (RRC). Si el RR recibe el anuncio de un prefijo por la sesión iBGP de un cliente, puede reflejar la ruta a otros clientes y no clientes (iBGP normales). Si un RR recibe un anuncio por una sesión de iBGP normal, puede reflejar la ruta a todos sus RRC.

A su vez las jerarquías se crean cuando un RR es reflector para un conjunto de routers, pero a su vez es cliente reflector de otros routers.

Existe otra alternativa menos utilizada llamada confederaciones. Estas permiten separar al AS en varios sub AS, dentro de los cuales deben de relacionarse full-mesh. El intercambio entre sub sistemas autónomos se realiza por las reglas de eBGP pero permitiendo preservar otros atributos como local-preference.

El escenario de confederaciones no plantea un camino simple de migración desde full-mesh o reflectores, por lo cual su aplicación es disruptiva.

b) La forma óptima para aprovechar los caminos de redundantes dentro del AS es definir IPs de loopback en cada enrutador, anunciar las interfaces de loopback en el IGP, y realizar las sesiones iBGP entre loopback.

Peer A	**Peer B**	**BGP Type**
rc1	rc2	iBGP normal son RR
ra1	rc1	iBGP RRC (ra1 RRC de rc1)
ra1	rc2	iBGP RRC (ra1 RRC de rc2)
ra2	rc1	iBGP RRC (ra2 RRC de rc1)
ra2	rc2	iBGP RRC (ra2 RRC de rc2)
rb1	rb2	iBGP normal
rb1	rc1	iBGP normal
rb2	rc1	iBGP normal
rb1	rc2	iBGP normal
rb2	rc2	iBGP normal

***Nota:** en las columnas Peer, se debe interpretar como sesiones iBGP desde las interfaces de loopback.*

Observación1: *Notar que entre rb1, rb2, rc1 y rc2 hay un full-mesh de iBGP.*

Observación2: *El clúster lo definen los dos RR rc1 y rc2, y los clientes ra1 y ra2.*

Observación3: *Si bien la mejor práctica es utilizar las loopback para las sesiones iBGP, y que estas loopback se distribuyan por el IGP, es posible realizar las sesiones desde las interfaces físicas. No presenta el mismo grado de disponibilidad, ya que una caída de un enlace, corta el dialogo iBGP. Por ejemplo, se corta el enlace entre rb1 y rb2, lo que aprende por eBGP rb2, no se lo puede enseñar a rb1, y viceversa.*

Si bien no es el ejercicio propuesto, el escenario de sesiones iBGP desde las físicas podría mejorar si rb1 y rb2 también son clientes reflectores de rc1 y rc2.

Si la sesión iBGP es desde las loopback, y se distribuyen por el IGP, normalmente el camino más corto entre las loopback de rb1 y rb2 es utilizar el enlace rb1-rb2. En caso de corte del enlace entre ellos, tienen dos caminos posibles, ***rb1-rc1-rb2*** *o* ***rb1-rc2-rb2****, el IGP decidirá cual de los dos utilizar.*

Por lo expuesto, queda claro que las sesiones iBGP desde las interfaces físicas y disponibilidad, fuerza a revisar los posibles escenarios de fallas, pudiendo omitir alguno. Si utilizamos las interfaces de loopback y el IGP, delegamos el resolver el camino alternativo al IGP, este mientras exista un camino posible entre los enrutadores, permite mantener la sesión iBGP activa (podría percibirse algún corte, mientra el IGP converge, usualmente los timers del IGP son menores que los timers de iBGP)

Observación4: *Para el cluster-id, recordar los pros y contras de utilizar mismo cluster-id o diferente cluster-id. Si utilizamos las loopback para las sesiones iBGP, y estas se propagar por el IGP, la diferencia entre alternativas es muy sutil, es solo disponer de dos caminos para un mismo prefijo que difieren en el atributo cluster-id.*

Si las sesiones iBGP se levantan desde las interfaces físicas, ahí si utilizar diferente cluster-id es altamente recomendable, de lo contrario perdemos mayor grado de disponibilidad.

Observación5: *En iBGP no tenemos un control de loop de información de ruteo, en eBGP el AS-PATH nos permite hacer este control, pero dentro del AS no existe un atributo que nos permita detectarlo. Esto fuerza a que lo aprendido por una sesión iBGP no se permita re-distribuir por sesiones iBGP.*

Un loop de ruteo puede generar un loop de forwarding, en definitiva los paquetes IP quedan dando vueltas entre los diferentes routers, sin alcanzar a ningún destino, descartándose cuando llegan a TTL=0.