**RESUMEN REDES DE INFORMACION 2do Parcial**

**CONTINUACION UNIDAD N°2: CAPA DE INTERRED**

Clase 28 de Mayo

**ICMP (Internet Control Message Protocol – Protocolo de Control de Mensajes de Internet)**

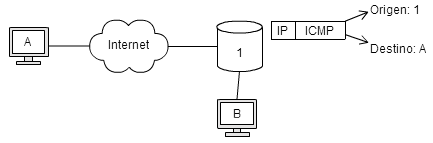
Este protocolo surgió como ayuda al protocolo IP para informar al origen cuando un paquete no puede ser entregado por diversos motivos, ya que el protocolo IP por sí solo no informa nada. Se ejecuta en la capa Interred y en todos los dispositivos.

¿Cómo funciona? Cuando un paquete no llega al destino ya sea porque el Host está apagado, o porque la red esta caída, o por cualquier otra causa, el router que detecte que el TTL de ese paquete llego a 0 va a ser el encargado de construir un paquete ICMP el cual es encapsulado en un datagrama IP. Dentro, el paquete ICMP contiene distintos códigos que representan a la variedad de posibilidades o causas por las que no puede ser entregado un paquete, por lo que existen diversos tipos de paquetes ICMP.

Algunos tipos de paquetes o códigos ICMP son:

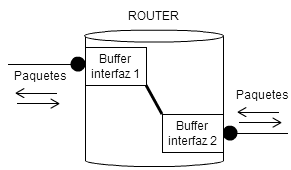
* Tiempo de vida excedido: Este paquete se construye cuando por un problema de encaminamiento un paquete queda dando vueltas por la red y su TTL (tiempo de vida) llega a cero. El router que detecte esta situación será el encargado de descartar ese paquete y crear un paquete ICMP que es encapsulado en un datagrama IP. En el encabezado del datagrama se colocara como dirección origen la dirección IP del router que creó el paquete y como dirección destino la dirección de la Pc que envió el paquete.

Por ejemplo si A envía un paquete a B y el mismo no puede llegar al destino por diversas causas. El Router 1 que detecta dicha situación crea un paquete ICMP que es encapsulado en un datagrama IP al cual le coloca como dirección destino la Pc que envió el paquete (o sea A) y como dirección origen la propia IP del router.



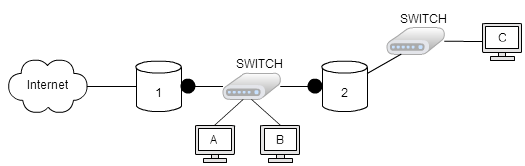
* Problema de parámetro: Este paquete es muy raro que se produzca ya que se crea cuando el software TCP/IP esta corrompido (no funciona bien) y se construyen encabezados incorrectos, lo que genera un problema de parámetros. En este caso se descarta el paquete porque no tiene bien estructurada la cabecera del paquete IP y se crea un paquete ICMP del tipo problema de parámetro. Decimos que es raro que esta situación se presente ya que es muy poco probable que la pila de protocolos TCP/IP falle.
* Source Quench (apaciguamiento en origen): Está relacionado con la calidad de servicio o también llamado control de congestión. Decimos que una red se congestiona cuando en la misma ingresan muchos paquetes, más de los que la red es capaz de procesar y encaminar.

¿Puede un router saber cuándo una red esta congestionada? Si puede, ya que la función del router es recibir paquetes por una interfaz (que tiene muchas) y a partir de una tabla de encaminamiento saca esos paquetes por otra interfaz. Cuando al dispositivo le llegan muchos paquetes por una interfaz los mismos se van colocando en una cola o buffer de entrada, y de la misma manera, una vez que son encaminados esos paquetes, se van colocando en una cola o buffer de salida de una interfaz. Es decir que cada interfaz tiene una cola de entrada y una cola de salida.



Si el router detecta que las colas de las interfaces comienzan a crecer mucho se da cuenta que la red se está congestionando ya que llegan más paquetes de los que es capaz de procesar. Para prevenir que esta situación ocurra se controla que las colas no se llenen, ya que cuando se llenan no hay más espacio físico en el buffer para almacenar los paquetes y los mismos deben ser descartados porque no pueden procesarse. Por lo tanto, cuando el buffer de un router se está llenando, el dispositivo crea un paquete ICMP de tipo Source Quench y le manda a decir al origen que no deje de transmitir pero que disminuya la tasa de transmisión para que pueda almacenar y procesar todos los paquetes que recibe. Esta es una forma de controlar la congestión.

* Redirect (Redirección): Supongamos tenemos la topología que se muestra en la imagen siguiente:



Dicha topología representa a una empresa en donde vemos que hay una cierta cantidad de máquinas conectadas a un switch, y a su vez este último dispositivo está conectado a dos routers. Si tuviéramos que configurarle una puerta de enlace o Gateway a la maquina A, ¿Seria la del router 1 o la del router 2? Antes que nada tengamos en cuenta que solo se le puede configurar una sola puerta de enlace a la Pc. Entonces, la respuesta va a depender de si la maquina A se comunica mayormente con las PC que están dentro de su red privada, en cuyo caso se configuraría como Gateway la IP del router 2; o de si la maquina A tiene mayormente una salida hacia internet, en cuyo caso se configuraría como Gateway la IP del router 1.

Supongamos que vamos a configurar como puerta de enlace para la maquina en cuestión la IP del router 2 porque en su mayoría las comunicaciones que efectúa son con las PC de su LAN. Si se presentara el caso en el cual la maquina A desea mandar una paquete hacia internet, el mismo se dirigirá hacia el router 2 como lo hemos configurado. En este caso, el router 2 se va a dar cuenta que no tiene sentido mandar el paquete hacia la empresa por lo tanto lo que hará es crear un paquete ICMP del tipo redirect y se lo va a enviar a la Pc A. Con esto se logrará que la Pc modifique momentáneamente su puerta de enlace de manera que re-direccione los paquetes hacia el router que le permite la salida a internet.

* Echo request (Solicitud de eco) y Echo reply (Respuesta de eco): Estos tipos de paquetes ICMP siempre van de a pares.

El comando **ping <dominio>** o **ping <dirección IP>** (ping6 o ping-6 para IPv6 en Windows XP y Linux respectivamente) es un programa que envía paquetes ICMP de Solicitud de Eco (echo request) hacia la dirección que hemos indicado. Como todas las maquinas tienen el protocolo ICMP funcionando, cuando reciben una echo request tienen la obligación de enviar una Respuesta de eco (echo replay). El comando ping me permite saber si un dispositivo está activo, si es operativo, si tengo conectividad hacia ese dispositivo; y además me brinda información sobre la transferencia de los paquetes. Valida la conectividad extremo a extremo.

Si el paquete de solicitud de eco nunca llego al dispositivo al cual se envió, no se obtendrá un paquete echo replay como respuesta, por lo que podría pensar que el dispositivo está inactivo pero en realidad no es así. Para que esto no suceda, al momento de ejecutar un echo request, se envían varios paquetes al dispositivo para evitar la hipotética situación en la que uno o más de ellos se pierda. Desde los sistemas operativos de los routers se envían 5 paquetes, desde Windows 4 paquetes, y desde Linux se mandan muchos paquetes hasta que el usuario decida cortar ese envío.

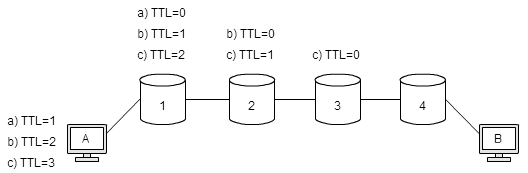
Cada vez que se envían estos paquetes recibo cierta información de la red que puedo analizar, como por ejemplo:

* El tiempo que tardó en ir y volver cada paquete.
* El tiempo mínimo, es decir cual paquete fue más rápido.
* El tiempo máximo, es decir cual paquete fue más lento.
* El tiempo promedio que tardaron los paquetes en llegar.
* Cantidad de paquetes enviados, recibidos, perdidos, etc.
* También dice: “respondiendo desde”, es decir que me da la dirección desde donde está saliendo la respuesta.
* Time Stamp Request (Solicitud de Marca de Tiempo) y Time Stamp Reply (Respuesta de Marca de Tiempo): Es muy similar el echo request y echo replay, con la diferencia de que además se incluye información del momento en el que llego el paquete. Es decir que no solo analiza el tiempo de ida y vuelta, sino que analiza cuanto tiempo tarda en llegar y cuánto tiempo tarda en regresar. Se utiliza para hacer pruebas de red, analizar la performance de la red (que tan rápido o lento viajan los paquetes por la red).

Comando **traceroute <dominio>** o t**raceroute <dirección IP>** (también **tracert**)**:** Muestra la traza o el camino por el cual viaja un paquete, es decir los routers que va atravesando un paquete. Se basa en paquetes echo request y echo replay, es decir en ICMP. El software del traceroute controla 30 saltos como máximo, ya que se supone que un paquete como máximo en 30 saltos tiene que llegar al destino, sino es significa que se quedó en un loop.

Si hacemos un ping y no obtenemos respuesta, en ese caso ejecutamos un traceroute para detectar en donde está el problema

La ejecución de este comando demora mucho tiempo debido a que su funcionamiento es el siguiente:



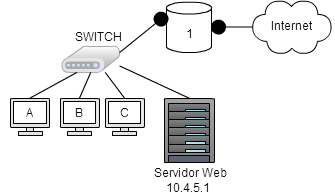
Supongamos que queremos saber por cuales routers pasa un paquete que viaja desde la Pc A hacia la Pc B. Para ello, se construye un paquete ICMP con el TTL=1 en el primer ciclo. Cuando este paquete sale de la Pc A y llega al router 1 el dispositivo disminuye en uno el valor del TTL dejándolo en cero. Como el TTL del paquete vale ahora cero, el mismo deberá ser descartado y por lo tanto el router construirá un paquete ICMP del tipo Time exceded (Tiempo de vida excedido) donde coloca su IP como dirección origen y se lo envía a la maquina A. De esta manera el traceroute ya sabe cuál es el primer salto.

A continuación el proceso se repite, solamente que esta vez el traceroute crea un paquete ICMP con el TTL=2. De esta manera, el paquete pasara por el router 1 donde el TTL valdrá 1, y finalizara en el router 2 donde el TTL valdrá cero. Esta vez entonces, el encargado de armar el paquete ICMP de tipo Time exceded será el router 2, quien colocara como IP origen su propia IP y se lo enviara a la maquina A. De esta manera el traceroute conoce el dispositivo que está en el segundo salto. El proceso de crear paquetes ICMP en la maquina A, aumentar su TTL en uno, enviarlo, recibir una respuesta, enviar un nuevo paquete, etc. Se repetirá hasta llegar a la dirección de destino de la maquina B. Así es como es posible conocer todos los saltos que da un paquete para llegar a su destino, y como previamente adelantamos, la ejecución del comando demora mucho tiempo debido a que los paquetes van y vuelven una gran cantidad de veces.

* Red Unreachable (Red Inalcanzable): Cuando un paquete llega a un router y ese router no tiene en su tabla de encaminamiento la dirección de red de destino, el dispositivo construye un paquete ICMP del tipo Red Inalcanzable para notificarle al origen que el paquete no puedo ser entregado y fue descartado.
* Host Unreachable (Host inalcanzable): Esto ocurre cuando se le envía un paquete a un dispositivo que está apagado.

**ARP (Address Resolution Protocol – Protocolo de Resolución de Dirección):**

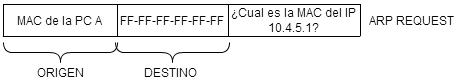
Este protocolo sirve para, dada una dirección IP de un dispositivo, averiguar cuál es la dirección MAC de ese dispositivo.



Supongamos que la maquina A quiere navegar por internet y por lo tanto comunicarse con el servidor Web. ¿Tiene forma la máquina de saber la dirección IP del servidor Web? Si, lo puede saber gracias al DNS que es una aplicación que a partir de un dominio me dice cuál es la dirección IP correspondiente a ese dominio. Entonces, supongamos que la maquina A desea navegar hacia una página web cualquiera. El navegador no puede construir un paquete ya que solo conoce el domino y no conoce la dirección IP de esa página. Por consiguiente lo que hace es ejecutar el DNS para que el mismo le devuelva la dirección IP de la página a la que desea acceder. Ahora sí, el navegador está en condiciones de construir el paquete con dirección IP de origen de la máquina que envía el paquete, mientras que la dirección IP destino será la de la página a la cual se va a acceder.

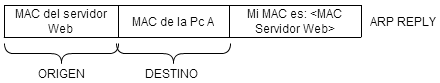
Ahora bien, esos datos que estaban en el protocolo HTTP del navegador van descendiendo por la pila de protocolos y se transforma en un segmento en la capa de transporte (protocolo TCP), luego se encapsula en un paquete IP en la capa Interred, y por último se encapsula en una trama Ethernet en la capa de Host a Red. Decimos en una trama Ethernet porque se trata de una red de difusión. Pero en este último paso es preciso especificar la dirección MAC origen y la del destino. La dirección MAC del origen la conocemos ya que la Pc A al bootearse la toma de la ROM de la placa, pero ¿Cuál es la dirección MAC del destino? Para saberlo utilizaremos el protocolo ARP que, como antes mencionamos, nos permite dada una IP averiguar cuál es la correspondiente MAC de esa IP. Nótese que la función del protocolo ARP es similar a la del protocolo DNS pero no es la misma, ya que en primer lugar se ejecutan en capas distintas y por otro lado averiguan cosas distintas.

Ahora explicaremos el funcionamiento del protocolo ARP: Como estamos hablando de una red de difusión los dispositivos necesitan algún esquema de direccionamiento para saber que tramas van dirigidos hacia ellos y cuáles no. Para ello se utiliza una consulta que se llama **ARP request – Solicitud ARP** que es una trama que es construida por la maquina origen, que en nuestro caso es la maquina A. Dicha trama tiene el siguiente formato:



Vemos que como dirección origen la trama lleva la MAC de la Pc A que es la que quiere comunicarse con el servidor Web. La dirección destino es la dirección de broadcast o difusión porque como la maquina A no sabe que MAC tiene el servidor, entonces lo obliga al switch que mande la solicitud por todos los puertos. En la parte de datos se coloca la IP del dispositivo sobre el cual queremos saber su dirección MAC, en nuestro caso colocamos la IP del Servidor Web. Esta trama será procesada por todos los dispositivos debido a que utilizamos como destino la dirección de difusión, pero solo emitirá la respuesta aquel dispositivo cuya dirección IP coincida con la indicada en el campo de datos.

La respuesta a la ARP request se conoce como **ARP reply – ARP respuesta**. La trama de dicha respuesta tiene el siguiente formato:



Donde en la dirección origen colocamos la dirección del dispositivo que emite la respuesta, en este caso el Servidor Web. En el campo de dirección destino colocamos la MAC de la Pc hacia la cual va destinada la respuesta, en este caso la Pc A. Por último, en el campo de datos viajara la dirección MAC del servidor que es la que precisaba la Pc A. Ahora si será posible que la Pc A se comunique con el Servidor Web.

¿Qué sucede si, un momento después, decido volver a comunicarme con el Servidor Web? No será necesario repetir el proceso antes explicado ya que la dirección del Servidor obtenida se guarda en una tabla ARP. Todas las Pc ejecutan el módulo ARP, es decir que tienen una tabla ARP, y tendrán una por cada placa de red. Un router que tiene muchas placas o interfaces tendrá muchas tablas, más específicamente tendrá una tabla por cada interfaz LAN. La tabla tiene la siguiente estructura:

|  |  |  |
| --- | --- | --- |
| IP | MAC | TIPO |
| 10.4.5.1 | 00-12-34-56-78-90 | Dinámico |

Cada entrada de la tabla tiene un tiempo de vida luego del cual expira y es borrada de la misma. Es decir que si nosotros ejecutamos un ARP y luego de un tiempo no utilizamos esa dirección conseguida, la misma será borrada de la tabla y cuando la necesitemos nuevamente deberemos ejecutar otra vez un ARP. La columna “tipo” dentro de la tabla hace referencia a si la dirección fue conseguida a través de la ejecución de un ARP o no, en el caso de que efectivamente haya sido así, el campo de esa columna será completado con la palabra “dinámico” debido a lo antes explicado. Cuando un campo de la columna “tipo” tiene el valor “estático” significa que esa dirección va a permanecer en la tabla y no se va a borrar por haber expirado su tiempo.

¿Por qué hacer que la tabla sea dinámica? Si a una Pc le cambian su dirección IP, o si le cambian la placa de red y cambia su dirección MAC, la asociación IP/MAC hecha en la tabla ya no es válida. Por ese motivo es que se hizo que la tabla sea dinámica.

Cada dispositivo, antes de enviar un broadcast para conseguir una dirección, se fija si no contiene dicha dirección en su tabla, de manera de evitarse la ejecución del ARP.

Los comandos ARP son:

**arp –a** 🡪 muestra toda la tabla ARP

**arp –s <IP> <MAC>** 🡪 para configurar una dirección de manera estática y que no se borre una entrada de la tabla.

**arp –d <IP> <MAC>** 🡪 para borrar una entrada de la tabla

¿Qué sucede si deseo comunicarme con una Pc que se encuentra en una red distinta a la mía? En este caso, si se ejecutara un ARP Request, el broadcast solamente llegara a las Pc que se encuentran en la misma red que la máquina que ejecuto el ARP, debido a que tiene alcance local (la MAC es una dirección Plana, de alcance local). Por lo tanto en estos casos lo que se hace es mandar un ARP request colocando como dirección IP destino la del Router (el gateway), de manera que consigamos la MAC de dicho dispositivo y podamos completar la comunicación. Obviamente, antes de mandar el ARP Request, lo que se hace es consultar la tabla para ver si ya poseemos el dato de la MAC del router.

Clase 4 de Junio

**METODOS DE CONFIGURACION DE DIRECCIONES IPv4:**

Existen dos formas de configurar una dirección IP a un dispositivo: estática y dinámica.

**Estática:** Al configurar de esta manera una dirección, si nosotros apagamos la Pc y la volvemos a encender, la computadora cuando este prendida se va a fijar que dirección IP tiene configurada y automáticamente la va a tomar y comenzara a tener conectividad.

Cuando nos referimos a una dirección IP decimos que es una dirección lógica, en cambio cuando nos referimos a una dirección MAC hablamos de direcciones físicas. Esto es así porque las direcciones MAC están grabadas en la ROM de la placa, mientras que las direcciones IP se graban en un archivo del sistema operativo.

Ventajas:

* Seguridad: El administrador sabe que una máquina que tiene configurada una dirección IP estática siempre tendrá esa dirección IP.
* Es simple

Desventaja:

* Si la máquina que posee configurada una dirección IP estática se saca de su red y es conectada en otra red o en otro switch, no tendrá conectividad hasta que se le cambie la dirección IP y se le configure una que pertenezca al rango de direcciones IP de la nueva red. Esto es así ya que al configurarle a una Pc un Gateway, esa dirección de puerta de enlace deberá estar si o si dentro del mismo rango de direcciones de las Pc de la Red. Por lo tanto, al sacar a la Pc de su red y conectarla en otra, es posible que la dirección estática que tenga configurada no se encuentre dentro del rango de las direcciones IP que utilizan las computadoras de la nueva red.

Generalmente la configuración estática de direcciones IP se utiliza en empresas que cuentan con pocos dispositivos, o aquellas empresas que una vez que arman su red no cambian de lugar los dispositivos. Además, un requisito para configurar una dirección IP estática es que las computadoras tengan disco rígido.

**Dinámica:**

Existen diversos métodos para la configuración de direcciones IP dinámicas.

* **RARP (Reverse Address Resolution Protocol – Inverso del Protocolo de Resolución de Dirección):** Como su nombre lo indica, este protocolo es el inverso del protocolo ARP. Es decir, si en el protocolo ARP podíamos averiguar una dirección MAC a partir de una dirección IP, en el protocolo RARP podremos averiguar una dirección IP a partir de su correspondiente MAC. Esta técnica no requiere que la Pc tenga disco rígido.

Cuando la maquina bootea no tiene dirección IP porque no tiene disco rígido, pero si tiene una

dirección MAC que la toma de la ROM de la placa. Esta dirección MAC es mandada a un servidor y este busca en una tabla la asociación MAC/IP que coincida con la MAC que recibió. Cuando encuentra la MAC se fija que IP tiene asociada y se la entrega a la máquina que lo solicito. Esta máquina tendrá conectividad mientras está encendida, pero cuando se apaga pierde su dirección IP porque no tiene disco duro donde almacenarla. Por lo tanto, cuando vuelva a ser encendida el proceso debe repetirse.

Para que este protocolo funcione hay que configurar un servidor, al cual se le deben configurar tablas que establezcan la relación entre MAC/IP. Es necesario que haya un servidor RARP por cada LAN, y cuando una maquina es agregada a una LAN su relación MAC/IP debe ser introducida en la tabla. El RARP suele utilizarse, por ejemplo, para los cajeros de los supermercados o cualquier dispositivo que no tenga disco rígido y necesite una dirección IP.

Desventaja: Se le pueden configurar pocos parámetros, solamente la IP, mascara y Puerta de Enlace. Además, cuando se le configura una dirección IP a una maquina esa dirección IP podrá ser usada solamente por esa máquina ya que en la tabla se encuentra almacenada dicha asociación. Es decir que por más que una computadora se apague y se vuelva a prender, siempre se le asignara la dirección IP que fue reservada para ella que aparece en la tabla. Cuando las maquinas son movidas de una red a otra deben actualizarse las tablas.

* **BOOTP (Bootstrap Protocol – Protocolo de Arranque):** Cuando una maquina arranca se ejecuta el Bootstrap que es un proceso a través del cual se hace un reconocimiento del Hardware de la Pc. Su Funcionamiento es similar al protocolo RARP, ya que también es necesario tener un servidor Bootp que cuando recibe una dirección MAC busca la dirección IP asociada a la misma.

Ventaja: Permite configurar más parámetros a una máquina. Por ejemplo, dominio al que pertenece una máquina, DNS primario, DNS secundario, etc.

Desventaja: Una vez que se le asigna una dirección IP a una máquina, esa dirección IP podrá ser usada solamente por esa máquina, queda reservada para ella (al igual que sucedía con RARP). Es decir que por más que una computadora se apague y se vuelva a prender, siempre se le asignara la dirección IP que fue reservada para ella que aparece en la tabla. Cuando las maquinas son movidas de una red a otra deben actualizarse las tablas.

**DHCP (Dinamic Host Configuration Protocol – Protocolo de Configuración Dinámico de Host):**

Este protocolo es una extensión o mejora del protocolo Bootp, y es el más usado en la actualidad. Consiste en un servidor que define un pool o conjunto de direcciones IP, las cuales pueden ser asignadas a los distintos dispositivos. Cuando una computadora se enciende, el servidor busca en el conjunto de direcciones alguna de ellas para “prestársela” a la Pc. La computadora utilizara esta dirección todo el tiempo que desee hasta que la Pc se apague. Cuando esto ocurre, la dirección IP asignada a la maquina vuelve al pool de direcciones quedando disponible para ser asignada a cualquier otra computadora. Recibe el nombre de dinámico justamente porque porque permite utilizar una IP del conjunto y cuando ya no se utiliza, esta vuelve a estar disponible. Este protocolo es de alto nivel ya que se ejecuta o está montado sobre el protocolo UDP de la capa de transporte, y por lo tanto trabaja con segmentos. Su funcionamiento es del tipo cliente-servidor, es decir que tiene un componente cliente y otro servidor. Además, como sabemos, en las distintas capas del modelo TCP/IP hay distintas formas de direccionamiento, por ejemplo en la capa de enlace se trabaja con MAC origen/MAC destino, en la capa de IP se trabaja con IP origen/IP destino, y en el caso de la capa de transporte se trabaja con puertos: puerto origen/puerto destino. Es por ello que el protocolo DHCP trabaja con el puerto cliente N° 68 y con el puerto servidor N° 67.

Es necesario configurar el cliente DHCP en la computadora, para ello ingresamos al entorno de red y seguimos el mismo procedimiento descripto para la asignación de una dirección IP estática, con la diferencia de que hacemos click en donde dice obtener una IP automáticamente. Notemos que cuando esto sucede se desactiva la opción de ingresar una dirección IP para la Pc porque la dirección se asignara dinámicamente. También es necesario configurar el DHCP en el servidor. En este caso es un poco más complejo porque se requiere configurar cual será el rango para el pool de direcciones que pondrán ser asignadas.

¿Conviene hacer las asignaciones de manera dinámica o estática? Depende del dispositivo, ya que a los servidores, impresoras e las interfaces de los routers les conviene tener configuradas direcciones IP estáticas porque si cambian de dirección cada cierto intervalo de tiempo no podrán ser encontrados por las computadoras. Por lo tanto, el DHCP permite configurar direcciones IP tanto dinámica como estáticamente, y para este último caso lo que hace es asociar una dirección MAC con una dirección IP. De esta manera, cuando le soliciten asignar una dirección IP, el protocolo se fija si para la MAC de ese dispositivo hay reservada una IP, y de ser así se la asigna.

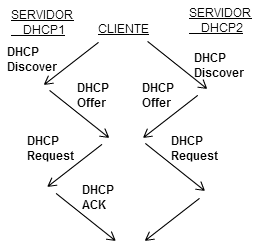
En DHCP también es posible hacer exclusiones de direcciones IP. Supongamos que yo asigno un determinado rango al pool de direcciones, pero no todas esas direcciones están disponibles para ser asignadas porque la 1 la usa el servidor de correo, la 2 la usa un router, la 3 la usa una impresora, etc. Por lo tanto, esas direcciones deberán ser excluidas del pool de direcciones porque no podrán ser asignadas de manera dinámica a los dispositivos por estar ocupadas. Por lo tanto este protocolo necesita que se le configure el rango de direcciones y las exclusiones.

Por otra parte, también es necesario configurar en el servidor lo que se conoce como lease o tiempo de alquiler. Previamente mencionamos que las asignaciones de las direcciones IP se hacen de manera dinámica, por lo tanto se considera que se le “alquila” o “presta” una IP a una computadora. Estas direcciones son prestadas por un periodo de tiempo (o lease) que esta predeterminado, aunque si se desea el administrador puede configurarlo. Ahora, ¿Qué sucede si una maquina está encendida, no se apaga nunca y vence su periodo de tiempo de préstamo? El protocolo se encarga de reasignarle la dirección IP a la computadora automáticamente. Para ello va controlando el tiempo de préstamo, y antes de que el mismo llegue a cero el protocolo le envía una solicitud de renovación al servidor. Si la Pc es apagada, el lease de la dirección vuelve a cero y la misma pasa al pool de direcciones.

Ventajas:

* Permite trasladar máquinas de una red a otra sin tener que cambiar todas las configuraciones. Simplemente con arrancar la maquina se le asignara dinámicamente una dirección IP del rango de direcciones de la red a la que está conectada.

Funcionamiento del DHCP:



Una vez que el servidor está en alto, en un estado conocido como listen (escuchando) los clientes comienzan la comunicación enviando peticiones hacia el servidor. Entonces, el Cliente al bootear envía un paquete llamado **DHCP Discover** para tratar de descubrir o buscar servidores DHCP. Cuando el paquete llegue a un servidor, el mismo se dará cuenta de que una Pc necesita una IP y, por lo tanto, buscara y tomara del pool de direcciones una dirección IP que no se esté usando para asignársela al dispositivo. Pero antes de la asignación, el servidor le ofrecerá la dirección y todos los parámetros de configuración (mascara, servidor DNS primario, servidor DNS secundario, dominio, puerta de enlace etc.) al dispositivo a través de un paquete llamado **DHCP offer**. La computadora en respuesta envía un paquete que se conoce **DHCP Request** o **Solicitud DHCP** en donde solicita y acepta los parámetros y dirección ofrecidos por el servidor. Ahora sí, el servidor le asigna la dirección IP al dispositivo y envía una paquete llamado **DHCP ACK** o **DHCP de Consentimiento** hacia la Pc donde le informa que ya tiene conectividad y le envía todos los parámetros e IP. A partir de este último paquete la computadora o dispositivo que solicito la dirección IP ya tiene conectividad.

¿Qué pasa si bootean 30 máquinas al mismo tiempo y todas quieren conectarse? Para que sea posible atender a todas cada mensaje DHCP tiene un ID de transacción que permite diferenciar las transacciones de las distintas máquinas. Todos los paquetes DHCP que se intercambian una Pc en particular y un servidor tendrán el mismo ID de Transacción, solo cambia para diferenciar las maquinas.

DHCP Discover:

Cada vez que arranca una máquina y crea un mensaje del tipo DHCP Discover se le genera automáticamente un ID de Transacción que le permite al servidor diferenciar las transacciones entre las distintas máquinas. Este número va a ir colocado en el campo de datos, junto con todos los parámetros de la conexión (mascara, servidor DNS primario, servidor DNS secundario, dominio, puerta de enlace etc.), solamente que como todavía no se asignó ninguna dirección esos campos valen cero.

Los datos son encapsulados en un segmento, ya que el protocolo corre sobre UDP. En el encapsulado del segmento únicamente debemos indicar cuál es el puerto origen y cuál es el puerto destino. Dijimos previamente que el cliente escucha por el puerto 68, que en este caso será el puerto origen ya que es el cliente quien manda el mensaje, y el servidor utiliza el puerto número 67, que en este caso será el destino ya que hacia él va dirigido el mensaje.



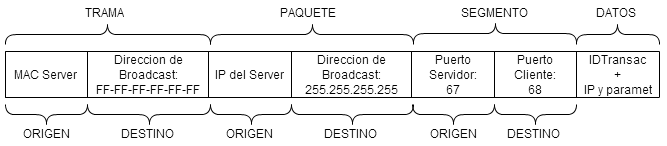
A su vez el segmento es encapsulado en un paquete, que llevara una dirección IP de origen y una dirección IP destino. En este caso, como la Pc recién esta booteando y no tiene una IP asignada (de hecho está solicitando una) en el campo origen ira la dirección 0.0.0.0 que es la de This Host; mientras que el campo destino llevara la dirección de Broadcast ya que la maquina está buscando un servidor y no conoce todavía la dirección de aquel con el que trabajará.

Por último, el paquete es encapsulado en una trama. En el origen colocaremos la MAC de A que es la Pc que inicia la comunicación; mientras que en el origen colocamos la dirección de Broadcast porque todavía no conocemos con que servidor vamos a trabajar y por ende su MAC. Algo interesante que podemos notar es que siempre existe una relación entre el direccionamiento de capa 2 y el direccionamiento de capa 3: si el direccionamiento de capa 2 es un Broadcast, también lo será el de capa 3, como ocurre en este caso con el campo destino. A su vez si la IP es Unicast también lo será la dirección MAC, como ocurre en este caso con el campo origen. Esta relación se da entre el origen de una trama de capa 2 y el origen de un paquete, o entre destino y destino.

DHCP Offer:

Cuando el servidor recibe el paquete DHCP Discover construye como respuesta un paquete DHCP Offer, donde le ofrece a la Pc que lo solicito una dirección IP. El campo de datos llevará el mismo ID de transacción que el mensaje DHCP anterior, y a su vez se comenzaran a completar los datos con los parámetros que el administrador haya configurado en el servidor y que se le asignaran a la Pc que solicito la dirección IP.

Los datos son encapsulados en un segmento, pero esta vez los puertos se van a invertir, es decir que en el puerto origen irá el 67 que corresponde al servidor y es quien envía el mensaje, mientras que en el puerto destino irá el 68 que corresponde al cliente y es quien recibe el mensaje.



El segmento es encapsulado en un paquete, en donde la IP de origen será la del servidor, mientras que la IP de destino es la dirección de Broadcast debido a que no podemos usar la dirección IP de la maquina porque todavía no se le ha asignado ninguna.

Por último, el paquete es encapsulado en una trama. Como MAC origen colocaremos la del servidor, mientras que para el destino utilizaremos la difusión, manteniendo la lógica con el direccionamiento de capa 3.

¿Qué sucede si se cae el servidor DHCP? Aquellas maquinas que ya tienen una dirección IP configurada no tendrán problema, pero aquellas que estén booteando no podrán conseguir ninguna IP porque nadie recibirá sus paquetes DHCP Discover. Por ello, para solucionar este inconveniente, lo que se hace es duplicar los servidores. Como los mensajes son mandados por Broadcast, serán recibidos por todos los servidores, cada uno de los cuales enviará una respuesta, por lo tanto el cliente deberá seleccionar aquella que llegue primero. La respuesta que emita el cliente llegara, nuevamente, por Broadcast a todos los servidores pero solo enviara el paquete de respuesta DHCP ACK aquel servidor que haya sido seleccionado (lo sabrá por el ID de transacción que tenga el paquete). El servidor que no haya sido seleccionado libera la dirección IP que iba a asignarle al dispositivo para que pueda ser utilizado por otro que la solicite.

* Mensaje DHCP Nack: Este mensaje es enviado del servidor al cliente. Esto ocurre cuando el servidor le ofreció a la computadora que lo solicito una dirección IP que previamente fue asignada a otro equipo, por lo tanto le avisa con este mensaje que le niega la asignación de dicha dirección. Por este motivo, siempre antes de enviar el mensaje DHCP ACK el servidor chequea que la dirección a configurar no pertenezca a otro equipo (no esté duplicada), y en caso de ser así envía el DHCP Nack. Esta situación puede ocurrir ya que los servidores DHCP pueden tener el mismo rango de direcciones IP a ser asignadas. Aquella Pc a la que se le niega la asignación de la dirección IP comienza el proceso nuevamente hasta que le es asignada una IP válida.
* Mensaje DHCP Inform: Este mensaje va desde el cliente hacia el Servidor, donde la maquina solicita que se le pasen nuevamente los parámetros que le han sido asignados. Generalmente se utiliza para chequear que la configuración sea la correcta.
* Mensaje DHCP Release: Este mensaje va desde el cliente hacia el Servidor y se utiliza para abandonar o liberar una dirección IP.

Comandos para utilizar DHCP:

C:\>ipconfig/release 🡪 libera la dirección asignada al equipo.

C:\>ipconfig/renew 🡪 solicita una dirección al/los servidor/es DHCP.

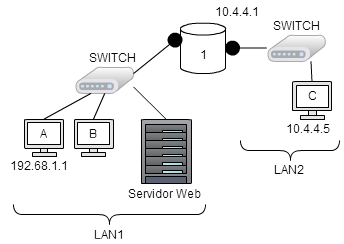
DHCP Relay (Retransmisión DHCP):

Supongamos que tenemos una empresa que tiene 20 LAN o redes privadas. Si quisiéramos asignar direcciones IP dinámicamente a las 20 redes utilizando el método RARP hubiera sido necesario tener 20 servidores, uno por cada LAN. Veremos a continuación que si utilizamos el protocolo DHCP será posible asignarle direcciones a todas las Pc de las distintas redes locales utilizando solamente 1 servidor.

Un router tiene la función especial de dividir dominios de broadcast, es decir que cuando una Pc que pertenece a una LAN manda un mensaje por difusión, ese mensaje únicamente tendrá alcance para todas las computadoras que pertenezcan a esa red privada, hasta la interfaz del router inclusive. Por ende, el mensaje de difusión no llegará hasta las computadoras pertenecientes a otras redes. Entonces, si la maquina A envía un Broadcast para pedir que se le asigne una dirección IP no habrá problema porque el servidor podrá escucharla debido a que están en el mismo dominio de Broadcast. Pero supongamos ahora que la maquina C es la que solicita que se le asigne una dirección IP y para ello envía un mensaje de difusión buscando al servidor DHCP. En este caso, dicho mensaje no llegara hasta el servidor por encontrarse en dominios de Broadcast diferentes. Para solucionar este inconveniente se configura una función especial en el router que se llama **DHCP Relay.** Esta función debe ser configurada en todas las interfaces donde no haya un servidor DHCP.

A partir de la función DHCP Relay lo que se hace es darle al router la dirección IP del servidor DHCP, de manera que cuando llegue hasta el router un paquete DHCP Discover solicitando una dirección IP, el mismo podrá transformar la comunicación de Broadcast en una de tipo Unicast desde su interfaz hasta el servidor DHCP. Esta petición llegara entonces hasta el servidor, quien se encarga de buscar una IP para ofrecer y se la enviara al router quien a su vez se la difundirá a la Pc. Como vemos, con la función DHCP Relay, el router hace de intermediario en la comunicación.

¿Cuántos rangos de direcciones IP tendrá el servidor? ¿Cómo sabe el servidor cuando le llega una petición si tomar una dirección para asignar que pertenezca al rango de direcciones de la LAN1 o de la LAN2?

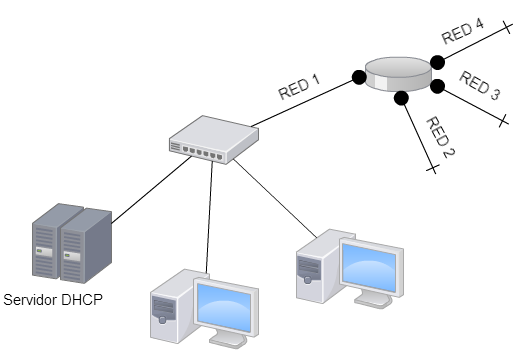


Supongamos que sale una petición de la maquina C que se encuentra en la LAN2 hacia el servidor que se encuentra en la LAN1. Al llegar dicha petición al router, el dispositivo completara el campo “Relay Agent IP Address – Dirección IP del agente de Relay” del paquete de la petición con la dirección de interfaz 10.4.4.1. De esta manera, el servidor sabrá que deberá ofrecerle a la máquina que lo solicito una dirección IP perteneciente al rango de direcciones que incluye a la dirección 10.4.4.1, que es el definido para la LAN2. Por lo tanto, el servidor podrá saber de qué rango de direcciones tomar una dirección IP libre para asignar en función del campo Relay Agent IP address que es completado por el router.

Clase 11 de junio

**Agente Relay DHCP:**

Si el cliente DHCP está en la misma red que el servidor, no hay problema ya que el servidor recibe el “discover” y le contesta, asignándole una dirección IP. Un ejemplo de esto en la imagen, sería que uno de los hosts de la red 1 solicitara una dirección de IP.

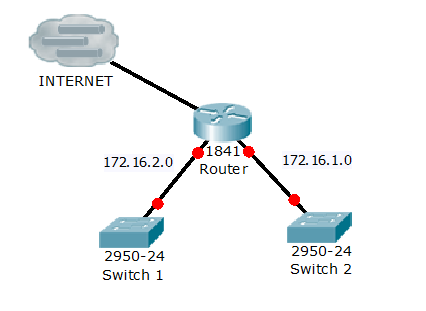
****Cuando existen multiples redes LAN conectadas, para no tener un servidor DHCP en cada LAN, conviene configurar un solo servidor DHCP que puede ser el router, un Windows, o un Linux. Y además, para que funcione correctamente, a todos los routers hay que configurarle un “agente relay”, el cual consiste en indicar la dirección IP de la red, donde se encuentra el servidor DHCP. Cada interfaz del router que se configura, lo que hace es retransmitir dicha petición “DHCP Discover”, transformándolo en un paquete unicast (que contiene una dirección IP origen y una de destino) agregando su IP en el campo "Relay agent IP adress”(IP propia del router que redirecciona el paquete) y lo direcciona hacia el servidor DHCP, cuando este lo recibe primero controla la dirección del router que hizo la petición y de acuerdo a esto busca que rangos de IP tiene configurado para dicho router, y le asigna una IP válida. En la imagen, el router que “une” las redes 1,2,3 y 4 debe ser configurado como agente relay para las redes 2, 3 y 4 ya que el servidor DHCP se encuentra en la red 1.

Agente Relay

**Traducción de direcciones de Red:**

**NAT(network adress translation):**

Supongamos el siguiente escenario



200.0.0.1

Mientras haya tráfico entre la subred 172.16.1.0 (desde ahora subred 1) y la subred 172.16.2.0 (desde ahora subred 2) no hay traducción de direcciones ya que el tráfico es dentro de la red. La traducción de direcciones IP se produce cuando hay tráfico entre algún dispositivo de la subred 1 o la subred 2, hacia internet. En este caso el router está pre-configurado para que no salgan al exterior paquetes con direcciones de IP privadas, es decir que se produce una traducción en dos situación: tanto cuando sale un paquete hacia internet como cuando viene un paquete de internet y entra a la red local.

Ventajas: La ventaja principal es que desde el exterior a la empresa no pueden ver la topología de la red, y no se puede acceder a una dirección de ip privada.

Otra ventaja es la seguridad, es decir que al tener acceso a internet en un solo punto, puedo configurar un firewall que indica que paquetes pueden salir y entrar, y que paquetes no.

Desventaja: La principal desventaja es que hay que hacer traducciones tanto cuando sale un paquete como cuando entra, entonces en el router se produce un “cuello de botella” porque todo el tráfico entrante y saliente pasa por él. Lo que provoca que se pierda mucho tiempo.

**Tipos de traducciones NAT:** Todas las traducciones NAT se realizan a través de tablas, estáticas o dinámicas.

* **NAT Estático:** Se produce una traducción 1 a 1, es decir que por cada dirección de IP privada voy a necesitar una dirección de IP pública. Este tipo de traducción se utiliza generalmente para servidores, y esto se debe a que de otra forma estos servidores no serían visibles desde el exterior. El NAT del router se configura de tal forma que en la tabla de direcciones, la dirección de IP privada que pertenece al servidor siempre va a verse desde el exterior con una determinada IP pública. Una desventaja de este tipo de traducciones es que no se ahorran direcciones de IP.
* **NAT Dinámico:** Es similar al DHCP. En este caso se definen un conjunto (pool) de direcciones públicas para realizar las traducciones. Por ejemplo, suponiendo que tengo 100 direcciones privadas en la empresa y se asigna un pool de 40 direcciones públicas, entonces cada vez que un dispositivo con IP privada quiere salir a internet, el NAT del router va a asignar alguna dirección del pool de direcciones públicas al dispositivo, registrando esta asociación en la tabla NAT. El proceso se repetirá con todos los dispositivos hasta agotar las 40 direcciones públicas que tenía disponibles.

La principal desventaja es que puede haber como máximo 40 dispositivos conectados a internet al mismo tiempo. Cuando un dispositivo ya no necesita conectarse a internet, la dirección de IP pública que utilizaba se libera, es decir que ser elimina la asociación de la tabla NAT, quedando disponible para otro dispositivo que requiera conexión. Otra desventaja es que si en un momento determinado desde internet se requiere acceder a un dispositivo que no tiene una asociación hecha en la tabla NAT no se podrá acceder a él, siendo posible solo si previamente el dispositivo solicitó conexión a internet.

* **PAT (Port Address Translation – Traducción de direcciones de puerto):** Hoy en día cuando se quiere acceder a internet ya no da “ocupado”, esto se debe a que existe un agotamiento de las direcciones IP se propuso la idea de hacer traducciones a nivel de puerto, es decir a nivel de capa de transporte.

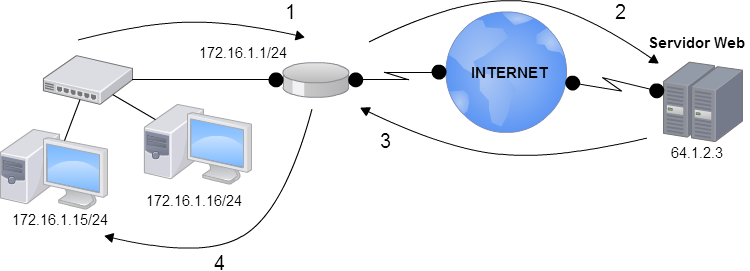
Los números de puertos sirven en el caso en que si en una misma máquina hay varias aplicaciones que envían y reciben paquetes a internet, como estos paquetes entran con la misma dirección IP, se utiliza el número de puerto para distinguir a que aplicación corresponde el paquete que acaba de ingresar. Este número de puerto está formado por 16 bits, con lo que puedo obtener 65.535 puertos posibles que se pueden clasificar en 3 tipos (definidos en RFC):

* + - De 0 a 1023 🡪 Puertos públicos o “bien conocidos” (well known):
    - De 1024 a 49151 🡪 Puertos Registrados
    - 49152 a 65535 🡪 Puertos Dinámicos o Privados

Por otro lado, la forma de describir una comunicación única y distinguible en internet es mediante un número que se conoce como “socket”. El socket está formado por una dupla IP + Puerto. Nunca existirán dos sockets que sean iguales.

En conclusión, PAT modifica la dirección IP (privada/publica) y el puerto (privado/público). Esto es útil ya que si una empresa tiene solamente una dirección de IP pública para “salir” a internet, todas las comunicaciones desde dentro de la empresa utilizarán solo esa dirección, sin tener problemas con la cantidad de host que pueda haber. Una desventaja es que el router que es un dispositivo de capa 3, al utilizar PAT modifica el puerto, metiéndose en la capa 4 de OSI, lo cual para algunos esto no debería ser así.

Veamos el siguiente gráfico cual es el comportamiento de 4 paquetes que entran y salen de la red LAN o de un Servidor Web.



Analizaremos los paquetes a continuación, donde se muestran los encabezados de paquete y segmento. Además los datos se encapsulan en una trama, con una dirección MAC que es obtenida del destino mediante ARP.

Paquete 1: De host a router

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | Paquete (IP) | | | Segmento (puerto) | |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 172.16.1.15 | 64.1.2.3 | 6000 | 80 | DATOS |
| *Origen* | *Destino* | *Origen* | *Destino* |  |

Paquete 2: del router de la empresa a internet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Paquete (IP) | | Segmento (puerto) | |  |
| |  | | --- | |  | |  | |  | | --- | |  | |  |  |
|  |  |  |  |  |
| 201.4.6.8 | 64.1.2.3 | 6000 | 80 | DATOS |
| *Origen* | *Destino* | *Origen* | *Destino* |  |

Paquete 3: del servidor al router

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Paquete (IP) | | Segmento (puerto) | |  |
| |  | | --- | |  | |  | |  | | --- | |  | |  |  |
|  |  |  |  |  |
| 64.1.2.3 | 201.4.6.8 | 80 | 6000 | DATOS |
| *Origen* | *Destino* | *Origen* | *Destino* |  |

Paquete 4: del router al host

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Paquete (IP) | | Segmento (puerto) | |  |
| |  | | --- | |  | |  | |  | | --- | |  | |  |  |
|  |  |  |  |  |
| 64.1.2.3 | 172.16.1.15 | 80 | 6000 | DATOS |
| *Origen* | *Destino* | *Origen* | *Destino* |  |

Supongamos ahora que una maquina tiene dos procesos contra el mismo servidor, es decir se accede a una página web, desde dos ventanas distintas para realizar tareas distintas. Si bien son de la misma PC para el SO son dos procesos distintos, entonces en ese caso en la tabla PAT del router vamos a tener la misma IP privada para ambos procesos, pero cada uno tendrá un puerto distinto para distinguirlos.

Otro caso que podría plantearse sería el de dos PC distintas que se encuentran dentro de una misma red, para las cuales se genera el mismo número de puerto para un proceso. En este caso, lo que hace el router es controlar que no exista un socket idéntico al que se quiere generar y si fuera así le cambia el número de puerto, es decir lo “traslada”.

Una desventaja de PAT es que si yo tengo una sola IP pública para salir a internet, y quiero tener 70000 servidores no puedo, a pesar de que tengo 65536 puertos, es demasiado procesamiento y no llegaría, se saturaría antes. La solución a esto es solicitar al ISP otra dirección pública y se le hace un overload o sobrecarga a 2 direcciones públicas.

Clase 18 de Junio

# UNIDAD N°3: ENCAMINAMIENTO Y CONGESTION

¿Qué dispositivo implementa el encaminamiento? El encaminamiento, que es una función de capa 3, es implementado por el router ya que es el dispositivo que está físicamente preparado para hacerlo, aunque una Pc también puede poseer una tabla de encaminamiento.

¿Qué es encaminar? Es ejecutar un **algoritmo de encaminamiento**, que es una parte de software de capa 3 (de red) que se encarga de que cada vez que llega una paquete por una interfaz decidir por cual de todas las líneas de salidas (o interfaces) va a salir.

¿En base a que se encamina? Se encamina en base a la dirección de destino. Cuando llega un paquete a un router en realidad llega una secuencia de bits. El dispositivo lo que hace es pasarle los bits a la capa de enlace la cual se va a encargar de determinar hasta donde llega la trama en dicha secuencia de bits, y se desencapsula la parte de la capa de enlace quedando solamente el paquete IP. El router toma la dirección IP de destino del paquete y la consulta en su tabla de encaminamiento, esto es compara la dirección con cada una de las entradas de la tabla para buscar una coincidencia (cabe aclarar que una tabla de encaminamiento únicamente posee direcciones de red o subred, no direcciones de Host porque si no sería muy extensa). Una vez encontrada la coincidencia, el router tendrá la información necesaria para encaminar el paquete, que consiste en la máscara e interfaz por la que saldrá el mismo, por lo tanto coloca al paquete en la cola de salida de la interfaz correspondiente. Debemos mencionar que un router posee una cola de entrada y una cola de salida por cada interfaz, que físicamente consiste en dos buffer, uno de entrada y otro de salida, en donde se van almacenando los paquetes. En la cola de entrada se almacenan los paquetes que llegan al dispositivo, mientras que en la cola de salida se colocan los paquetes que van a egresar del mismo. La interfaz de salida para poder sacar el paquete debe encapsularlo en una trama, y dicho encapsulamiento va a depender de la tecnología que haya a partir de esa interfaz (puede ser PPP, ATM, Ethernet etc.), por lo tanto antes de sacar un paquete por una interfaz el router debe consultar que protocolo de capa 2 o tecnología hay en esa interfaz. Los routers se denominan dispositivos de almacenamiento reenvío, esto significa que receptan el paquete, controlan que el mismo haya llegado correctamente y de ser así lo procesan o caso contrario lo descartan.

Como vemos, un paquete para llegar de un origen a un destino es posible que realice varios saltos, es decir que atraviese varios routers, lo que implica muchos encapsulamientos y desencapsulamientos. Por ello las direcciones de capa 2 se denominan planas o de alcance local, es decir que van cambiando salto a salto dependiendo de la tecnología que haya, mientras que las direcciones de capa 3 (IP origen e IP destino) no deberían cambiar extremo a extremo. La única forma de una dirección de capa 3 cambie es si se implementa NAT, es decir la traducción de direcciones privadas a públicas al salir a internet; pero si la comunicación tiene lugar entre dispositivos de una misma red la dirección IP no va a cambiar nunca de extremo a extremo en dicha comunicación.

**Protocolos de encaminamiento:**

Los protocolos de encaminamiento implementan los algoritmos de encaminamiento.

**Requisitos de los algoritmos de encaminamiento:**

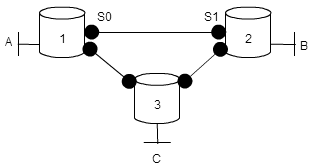
Los algoritmos de encaminamiento deben cumplir con los siguientes requisitos:

* **Simplicidad:** La simplicidad hace referencia a que un algoritmo de encaminamiento no sea complejo en su implementación, es decir que no gaste muchos recursos para decidir por donde encaminar un paquete. Para implementar algún algoritmo de encaminamiento siempre se debe hacer algún gasto de recursos, como por ejemplo tiempo de procesamiento del microprocesador, memoria RAM del procesador, etc. Otro recurso que se consume es el ancho de banda ya que los routers deben mantener actualizadas sus tablas de encaminamiento y para ello es necesario enviar actualizaciones a través de la red que consumen ancho de banda. La idea es que se demore y se consuma lo menos posible a la hora de aplicar los algoritmos para que el router no pierda mucho tiempo y pueda encaminar más paquetes.
* **Corrección:** hace referencia a que el encaminamiento de un algoritmo debe hacerse de manera eficiente, logrando que el paquete llegue al destino por medio de la ruta más corta o por la ruta en donde se tarda el menor tiempo posible.
* **Robustez:** hace referencia a que los routers pueden adaptarse a los cambios de topología. Es decir que cuando un router detecta que la interfaz por la cual va a enviar un paquete esta caída deberá avisar rápidamente a sus vecinos que la red es inaccesible o no está disponible en el tramo que se comunica con dicha interfaz. Estas notificaciones se hacen por medio de las actualizaciones que se envían los routers entre sí.
* **Estabilidad:** Esta característica está relacionada con la **convergencia**. Sabemos que por lo general cada router está conectado a muchas redes LAN. Por lo tanto, para que todos los dispositivos encaminen correctamente, es decir para que todos los routers conozcan las distintas redes que existen y que cuando les llegue un paquete con una dirección IP determinada no tengan que descartarlo por no conocer la red a la que debe llegar, es necesario que entre ellos se envíen notificaciones o actualizaciones en donde se indiquen las distintas redes LAN a las que están conectados. Como consecuencia, lo antes mencionado implica que un router debe conocer (debe tener en sus tablas de encaminamiento) no solamente las redes privadas que tiene directamente conectadas, sino también las redes privadas que están conectadas a otros routers. Por ello, cuando arrancan y se encienden los routers pasa un tiempo hasta que entre todos se envían información y entre todos aprenden todas las redes que hay en la red y cuáles de ellas están activas. Una vez que todos los dispositivos saben cómo llegar a todas las redes de la topología se dice que se logró la **convergencia**. A veces por error de encaminamiento tarda mucho tiempo en lograrse la convergencia antes mencionada, pero una vez lograda se dice que la red es **estable** y por un tiempo va a funcionar siempre igual. La red dejara de ser estable cuando se caiga una red privada o se dé de alta alguna de ellas, y en consecuencia el router que detecte cualquiera de estas situaciones deberá avisarle al resto de sus compañeros para que nuevamente se logre la convergencia. Una red que no logre nunca la convergencia se dice que es inestable.
* **Optimalidad:** significa cumplir todas las funciones de manera óptima, de la mejor manera posible. Implica ser eficaz y eficiente a la vez, es decir no solo cumplir el objetivo que es encaminar los paquetes sino lograrlo de la mejor forma posible consumiendo la menor cantidad de recursos que se pueda, encaminando rápido, etc.

**Tipos de encaminamiento (Forma de construcción de las tablas de encaminamiento):**

Un router aplica un algoritmo de encaminamiento y encamina consultando una tabla de encaminamiento. Esa tabla se puede construir de dos formas:

* **Encaminamiento estático (No adaptativo):** Un encaminamiento es estático cuando el administrador le indica al router que para llegar hasta una red tiene que salir por una interfaz determinada. Es decir que es el administrador quien le dice al router como tiene que armar su tabla de encaminamiento. Dependiendo del sistema operativo esto se logra, en el caso de Linux con el comando **route add**, y si es un router Cisco se ejecuta el comando **ip route**, pero de todas maneras en ambos casos se debe indicar la dirección de red a la que quiero llegar, la máscara y una interfaz de salida o ip del próximo salto.



Por ejemplo si consideramos la gráfica, supongamos que desde el router 1 queremos llegar a la LAN B, por lo tanto deberemos indicarle al dispositivo que cada vez que se quiera llegar a la red B se debe salir por la interfaz Serial 0 (S0). Para ello se anotara en la tabla de encaminamiento del router 1 la dirección de red, mascara y la interfaz serial 0. Entonces, cada vez que venga un paquete con dirección de destino igual a la de la red B, el router va a encaminar ese paquete por la interfaz que le fue indicada y anotada en su tabla de encaminamiento.

Las ventajas del encaminamiento estático son:

* no se consumen recursos de red (ancho de banda de la red, ciclo de CPU ni memoria RAM) porque no se requiere que se envíen actualizaciones de encaminamiento entre los routers. Estas actualizaciones solamente son enviadas entre los dispositivos para construir tablas de manera dinámica, es decir para realizar un encaminamiento dinámico. Aquí es el administrador quien le dice al router como quiere que encamine.
* Es más seguro porque el administrador sabe exactamente por donde van a viajar los paquetes porque es él quien lo indica.

Las desventajas del encaminamiento estático son:

* No se adapta a los cambios de topología, por ello recibe el nombre de no adaptativo. El Router en sus tablas solo pone las redes activas, por lo tanto si una interfaz no está activa, es decir está en un estado que se conoce como down, el dispositivo borrara la entrada referente a dicha interfaz. De esta manera, cuando un enlace se cae también se cae la interfaz relacionada con dicho enlace, y por más que el administrador le haya indicado al router que cuando llega un paquete con una determinada dirección debe salir por una determinada interfaz, si ese enlace esta caído también lo estará su interfaz y el dispositivo directamente borra esa entrada de su tabla de encaminamiento dando como resultado que el paquete no podrá ser enviado.
* Necesita una doble configuración de encaminamiento. Por ejemplo si consideramos la gráfica antes presentada y suponemos la situación en la que una Pc de la LAN A se quiere comunicar con una Pc de la LAN B deberemos tener que indicarle al router 1 cuál es la interfaz por la que va a salir su paquete cuando se dirija hacia la red privada B, y de la misma manera deberá configurarse en el router 2 cuál es la interfaz por la cual debe salir un paquete si su destino es el de la red privada A. Es decir que el encaminamiento estático necesita que se le configuren las rutas en ambos sentidos de la comunicación.

Solo conviene configurar estáticamente el encaminamiento cuando se trabaja sobre redes chicas, no así cuando trabajamos sobre una red que tiene muchas redes privadas. Principalmente porque es muy laboriosa la configuración de cada uno de los routers, y además porque si llegara a agregarse una nueva red o si se cayera algún enlace el trabajo de actualizar las tablas de encaminamiento demandara mucho tiempo del administrador.

* **Encaminamiento dinámico (Adaptativo):** Permite adaptarse a los cambios de topología. Solamente hay que configurarle al router una sola vez el encaminamiento dinámico y luego el dispositivo por su cuenta ira actualizando su tabla de encaminamiento tanto si se levanta una nueva red como si se cae un enlace o interfaz. La forma de actualizar su tabla de encaminamiento es a partir de actualizaciones que se van pasando entre todos los routers cada cierto tiempo cuando detectan que se produjo algún cambio en la red de los antes mencionados. Las actualizaciones, que son enviadas entre routers adyacentes, contienen información acerca de cómo hace un router para llegar a las distintas redes, y de esta manera entre todos los dispositivos aprenden como llegar a todas las redes.

Las ventajas del encaminamiento dinámico son:

* Flexibilidad para adaptarse a los cambios de topología. Si se da de alta o se cae una red, rápidamente se envían actualizaciones entre los routers y todos se enteran del suceso que ocurrió, sin necesidad de intervención de un administrador.

Las desventajas del encaminamiento dinámico son:

* No es seguro porque no hay certeza de por dónde van a viajar los paquetes ya que eso va a depender de que enlaces estén en alto y cuáles no, y además es una decisión que toma el dispositivo por su cuenta.
* Consume ancho de banda de los enlaces, ciclos de vida del micro, memoria RAM, pérdida de tiempo para la construcción de las actualizaciones así como para enviarlas a los routers adyacentes, pérdida de tiempo para procesar las actualizaciones que se reciben. Además, los protocolos más viejos están configurados para que cada cierto periodo de tiempo se envíen actualizaciones y no solo cuando se produce algún cambio en la topología de la red.

El protocolo de encaminamiento dinámico tiene que estar configurado en todos los routers de la red debido a la necesidad de enviarse actualizaciones para formar las tablas de encaminamiento, no es posible lo ejecuten algunos y otros no. Este tipo de encaminamiento se utiliza para redes grandes.

Un router puede tener configurado tanto un encaminamiento dinámico como uno estático al mismo tiempo, pero en este caso se aplicara primero el tipo de encaminamiento estático porque es más confiable y dejara en segundo plano al encaminamiento dinámico. Lo antes mencionado se conoce como distancia administrativa. Un router puede tener configurado muchos protocolos de encaminamiento, es decir que puede llegar a un destino de distintas maneras pero debe elegir solo una, y será aquella que tenga una menor distancia administrativa.

Diferencia entre un switch y un router: El switch no necesita ser configurado, cuando arranca tiene las tablas vacías y a medida que van llegando paquetes al dispositivo él mismo se encarga de ir armando su tabla de encaminamiento. En cambio un router debe ser configurado por un administrador: la ip de cada una de sus interfaces y el encaminamiento deben serle indicados, ya que por defecto ningún router encamina (solo traen su sistema operativo).

**Algoritmos de encaminamiento:**

La teoría de encaminamiento se representa por la Teoría de Grafos, en donde los routers son presentados como nodos y los enlaces entre los routers son graficados como arcos.

* **Algoritmo de la ruta más corta:** También conocido como el algoritmo de Dijkstra. Este algoritmo consiste en primer lugar en hacer un relevamiento de la topología de la red para luego realizar el grafo de la misma, y una vez hecho esto analiza todas las posibles rutas que haya para comunicarse desde un router determinado hasta otro en busca del camino más corto.

Para poder seleccionar cual es la ruta más corta el algoritmo tiene en cuenta un criterio que se conoce como **métrica**. Este algoritmo le da un determinado valor a cada arco del grafo en función de la métrica que se defina, y siempre busca como resultado el valor matemático más chico. En realidad, si bien el algoritmo se conoce bajo el nombre de “la ruta más corta” un nombre más representativo podría ser “la ruta más conveniente” ya que existen varias métricas o diferentes criterios para decidir cuál es el mejor camino, como por ejemplo:

* Cantidad de saltos: Esta métrica era la utilizada por los primeros protocolos en donde se contaba la cantidad de routers por los que pasaba un paquete para llegar a su destino. Era efectiva ya que todos los enlaces entre routers tenían la misma velocidad y en este caso cada arco recibe el valor 1. El proceso consistía en calcular, para cada camino posible, la cantidad de saltos que daba un paquete para llegar a su destino y aquel camino que tuviera menor valor, es decir aquel que diera menos saltos, sería el más corto.
* Ancho de banda (1/Bw): En este caso cada arco tiene un valor que representa al ancho de banda del enlace. Lo conveniente es elegir aquel camino que posea un mayor ancho de banda de transmisión, pero como el algoritmo selecciona la ruta más corta al final del proceso seleccionara aquel camino que tenga un menor valor. Por lo tanto, para evitar este inconveniente en lugar de colocar el ancho de banda como valor para cada uno de los arcos utilizaremos su inversa, así cuanto más grande sea el denominador menor será el valor del enlace, y al realizar la suma de todos los enlaces para llegar al destino aquel que presente un menor valor será el que tenga un mayor ancho de banda.

Hoy en día este algoritmo es muy utilizado ya que los enlaces pueden presentar distintos anchos de banda.

* Costo: Hay enlaces que son muy costosos para atravesarlos, y si uno quiere pasar por él tiene que pagarle al proveedor de internet. Por lo tanto podría utilizarse el algoritmo para calcular el camino que sea más barato o menos costoso.
* Retardo: Consiste en elegir el camino que tarde menos tiempo en llegar al origen. El camino más rápido no necesariamente es el que mayor ancho de banda tenga ya que pueden estar muy congestionados los routers.
* Longitud promedio de las colas de los routers

Hay protocolos que utilizan una sola métrica, pero también hay otros protocolos que utilizan varias métricas que es lo que se conoce como **función**. La función es la suma de muchos términos, en donde en cada término va a haber un coeficiente K que ya trae un valor por defecto pero que puede ser modificado por el administrador si así lo desea, y ese coeficiente estará multiplicando a una métrica. El motivo de que la métrica este siendo multiplicada por un coeficiente es el de darle un peso o importancia a una métrica por sobre otra. Es decir que si una métrica es más importante para el administrador en comparación con otra métrica entonces la primera estará multiplicada por un valor de coeficiente más bajo para denotar dicha situación.

F=

* **Inundación: COMPLETAR**

Clase 25 de Junio

**Encaminamiento**

Existen dos tipos de redes según el tipo de encaminamiento:

* Redes de Datagramas o de Conmutación de Paquetes:

En una transmisión cada paquete es tratado de forma independiente y encaminado por un camino diferente. Es más lento, pero es robusto, ya que si un router se cae, no se pierde la comunicación, solo se pierden algunos paquetes que son los que estaban en ese momento en la memoria del router caído.

* Redes de Circuitos Virtuales

Es un hibrido entre la conmutación de paquetes y la conmutación de circuitos, ya que para enviar datos desde un punto A a un punto B, no se mandan directamente sino que se debe establecer el camino por el cual se dará la comunicación. Esto se llama “abrir el circuito”. El encaminamiento se produce solo al comienzo, donde el primer paquete va abriendo el camino y los demás lo van siguiendo. Por este motivo estas redes son más rápidas.

Consta de 3 fases al igual que en la red telefónica: establecimiento de conexión, transferencia de datos y liberación del circuito. La diferencia con esta es que el ancho de banda no es dedicado, sino compartido. Todos los paquetes viajan por un mismo camino físico, pero sin ancho de banda dedicado.

Los paquetes de datos contienen sólo el número del circuito virtual para identificar al destino. En cada salto no se deberá encaminar como sucede en las redes de Datagramas, sino que solo se conmutan los paquetes, lo cual es mucho más rápido.   
Si se produce algún inconveniente en la red, todos los circuitos virtuales en la red se pierden y hay que volver a establecer la conexión.

**Algoritmo Jerárquico**

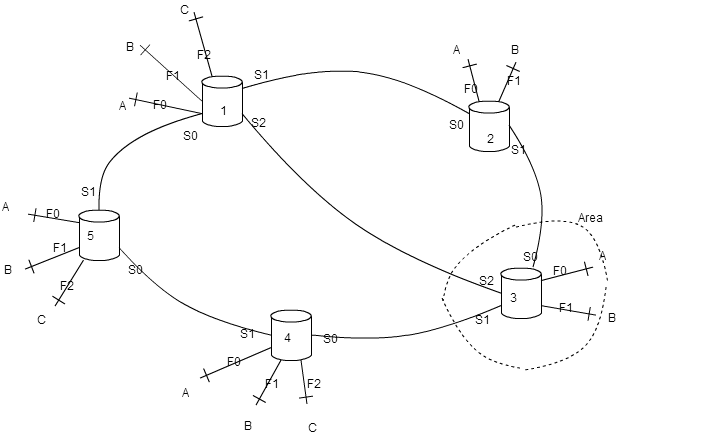
Cuando las redes son muy grandes, es difícil que un sólo router contenga toda la información de encaminamiento de la red. Además, los cambios que se producen en la red son constantes, con lo que nunca se tendría una información fiel del estado de ésta en cada momento. Para ello, se realiza un encaminamiento jerárquico, en el que cada router pertenece a un nivel de jerarquía, (área) reenviando los paquetes a los encaminadores de una jerarquía inmediatamente superior o inferior. El proceso se repite hasta llegar al nivel de los hosts. Este algoritmo busca reducir las entradas en las tablas de encaminamientos de los routers, ya que mientras más grandes las tablas, mayores recursos son necesarios para recorrerlas y consume más tiempo.

Veamos un ejemplo:

Dado el siguiente esquema, 1, 2, 3, 4 y 5 son routers, F0 F1 y F2 son interfaces LAN y S0 S1 y S2 son interfaces WAN (serial punto a punto). F0 = Fastethernet 0. S0 = serial 0

Las interfaces LAN se conectan a un switch, las WAN son interfaces punto a punto.

En cada router hay direcciones de red y subred.

  
La tabla de encaminamiento del router 1 por ejemplo, debe contener en cada entrada todas las redes, una métrica (en este caso va a ser la cantidad de saltos) y la interfaz por donde enviar los paquetes hacia esa red. La tabla sería:

3

Redes WAN

Redes LAN

|  |  |  |
| --- | --- | --- |
| Red | Métrica | Interfaz |
| 1A | 0 | F0 |
| 1B | 0 | F1 |
| 1C | 0 | F2 |
| 2A | 1 | S1 |
| 2B | 1 | S1 |
| 3A | 1 | S2 |
| 3B | 1 | S2 |
| 4A | 2 | S0 |
| 4B | 2 | S0 |
| 4C | 2 | S0 |
| 5A | 1 | S0 |
| 5B | 1 | S0 |
| 5C | 1 | S0 |

Cabe aclarar que en la tabla de encaminamiento también se incluyen las redes Wan, en este caso no se pusieron para simplificar el ejemplo.

En la tabla el router debe conocer todas las redes y las interfaces para llegar a ellas. El algoritmo jerárquico lo que hace es agrupar las redes por AREAS que serían grupos jerárquicos y evita conocer a todas las redes, sino solamente encaminar hacia las áreas y luego dentro de cada área se vuelve a encaminar.  
Aplicando el algoritmo jerárquico, la tabla de encaminamiento quedaría:

|  |  |  |
| --- | --- | --- |
| Red | Métrica | Interfaz |
| 1A | 0 | F0 |
| 1B | 0 | F1 |
| 1C | 0 | F2 |
| área 2 | 1 | S1 |
| área 3 | 1 | S2 |
| área 4 | 2 | S0 |
| área 5 | 1 | S0 |

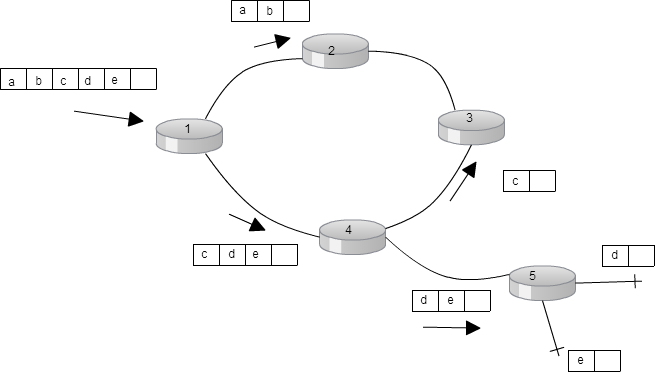
De esta manera, el router 1 ya no necesita saber cómo llegar a todas las redes, sino solamente a sus propias redes y a las demás áreas.

**Algoritmo de Difusión**

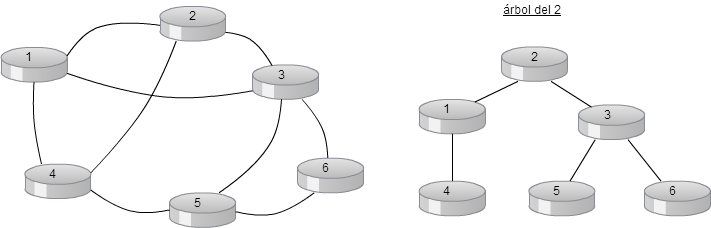
A la transmisión de un paquete, en forma simultánea a todos los destinos, se le conoce como difusión. Para esto existen 3 métodos:

* Encaminamiento multidestino
* Árbol de expansión
* Reenvío por ruta invertida

El encaminamiento multidestino consiste en construir un paquete con muchos destinatarios. Cuando ese paquete llega a un router, este construye nuevos paquetes con menos destinatarios, enviándolos por diferentes interfaces. Esto se dá en cada salto, hasta llegar a un paquete Unicast (un solo destinatario).



En el árbol de expansión, se eliminan caminos redundantes, asegurándose que el paquete se envíe una sola vez. Cada router construye un árbol de expansión consigo mismo como raíz para poder llegar a todos los destinatarios una única vez para no consumir tanto ancho de banda como lo hace la inundación. La desventaja que tiene es que no siempre se puede implementar, porque para que cada router pueda armar su árbol de expansión, tiene que conocer la topología de la red.



En el reenvío por ruta invertida, cuando un paquete llega a un router, éste analiza si el camino por donde llegó es la ruta más corta, es decir, si es el camino por donde el propio router al que le llega el paquete hubiera enviado el paquete si la comunicación fuera en sentido invertido.  
Si A transmite a B, B verifica si lo que llega de A viene por el camino más corto, es decir, si viene por donde B hubiera enviado los paquetes a A.

Si es el camino más corto, significa que es el primer paquete que llega y debe retransmitirlo. Si no es el camino más corto significa que ese paquete ya llego antes y lo descarta ahorrando ancho de banda.

**Encaminamiento por multitransmisión o multicast:**

En este protocolo, a diferencia del encaminamiento por difusión donde un mensaje se envía a todos los host de una red, los mensajes se envían a un grupo donde no necesariamente todos tienen que estar en la misma red o subred. Una computadora puede tener configurada además de una dirección IP una dirección IP de grupo, entonces cada vez que esa computadora se agrega a un grupo se dice que va a escuchar además de su IP, paquetes que tengan la IP de grupo que tiene configurada. Las IP de multicast son las de clase D y van de la 224.0.0.0 a la 239.255.255.255, aunque casi todas comienzan con la 224.

Los Routers tienen que conocer cuáles son las máquinas de las redes que ellos conectan, a que grupo pertenece para que cuando llegue un paquete de multidifusión sepa donde enviarlo, para esto existe un protocolo que se llama IGMP (*Internet Group Management Protocol)*.

Para manejar esto, primero que nada cuando una máquina quiere inscribirse a un grupo el protocolo va a registrar que pertenece a ese grupo y se lo va a informar a todos los otros Routers para que estén informados. A su vez los Routers pueden manejar muchos grupos, para distinguirlos crea lo que se llama árbol de expansión, con él mismo como raíz del grupo. De esta forma cada router va a tener un árbol de expansión por cada grupo de maquinas que esté manejando en un determinado momento, entonces cuando llega un paquete con destino a un grupo el router busca el árbol de expansión de ese grupo y solo lo envía a las redes que corresponden.

Clase 30 de julio

**Algoritmo de vector de distancia:**

Es uno de los primeros algoritmos de encaminamiento que se utilizó en internet.

Las principales características son:

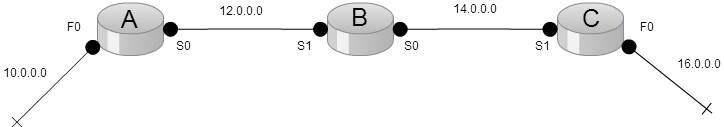
* Es dinámico: no es estático, es decir que las tablas de enrutamiento se actualizan con la red encendida y automáticamente. Se adapta a los cambios de topología.
* Es Distribuido: Significa que entre todos los Routers de la red comparten información y se ayudan entre ellos a armar las tablas de encaminamiento.

Se llama “de vector” porque al querer enviar un paquete se conoce por cual interfaz debo enviarlo es decir, que apunta hacia algún lado. Además se llama “de distancia” ya que se puede saber cuántos saltos, o cual es la métrica para llegar a una determinada red. Este algoritmo no es capaz de armar la topología de la red, ni un grafo que pueda representarla.

Mediante este algoritmo cada router regularmente les pasa toda su tabla de encaminamiento a los Routers vecinos, esto es malo ya que se consume ancho de banda inútilmente porque cada un período de tiempo determinado cada router realiza actualizaciones sin importar si hubo o no hubo cambios en la red y las envía por broadcast.

Por último hay que destacar que el protocolo RIP de encaminamiento, implementa este algoritmo.

Funcionamiento: Supongamos la siguiente topología de red



En cada router tenemos interfaces del tipo FastEthernet (F0, F1, etc.) y del tipo Serial (S0, S1, etc.). Inicialmente cuando se activa una interfaz el router comienza a armar su tabla de encaminamiento de forma autmática, por ejemplo si al router A en la interfaz F0 se le configura la dirección 10.0.0.8 va a suponer que en dicha interfaz está la red 10.0.0.0 y automáticamente graba estos datos en la tabla de encaminamiento.

De esta forma cada router al bootear, si están activas sus interfaces y configuradas con direcciones IP comienza a armar las tablas de encaminamiento automáticamente quedando de la siguiente forma:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Dirección de red** | **Interfaz** | **Distancia** |  | **Dirección de red** | **Interfaz** | **Distancia** |
| 10.0.0.0 | F0 | 0 |  | 12.0.0.0 | S1 | 0 |
| 12.0.0.0 | S0 | 0 |  | 14.0.0.0 | S0 | 0 |
| *Tabla de encaminamiento del router A* | | |  | *Tabla de encaminamiento del router B* | | |

|  |  |  |
| --- | --- | --- |
| **Dirección de red** | **Interfaz** | **Distancia** |
| 14.0.0.0 | S1 | 0 |
| 16.0.0.0 | F0 | 0 |
| *Tabla de encaminamiento del router C* | | |

Luego de esto, debo indicarle que tipo de encaminamiento quiero que utilice. Si es encaminamiento estático debería configurar todas las tablas de encaminamiento en todos los Routers. Pero en el encaminamiento dinámico, se le indica inicialmente y luego el router realiza las actualizaciones de forma automática con la ayuda de los Routers vecinos.

Lógicamente cada tabla de encaminamiento debería tener la misma cantidad de renglones como cantidad de redes existan, en este caso son 4 las redes. Por esto se dice que en el caso de que las tablas de encaminamiento de los Routers sean las mencionadas anteriormente, la red no converge.

Se dice que la red converge cuando todos los Routers saben cómo llegar a todas las redes de la topología, y esto se completa mediante el algoritmo en cuestión. Luego de configurarse en todos los Routers, comienzan las actualizaciones, supongamos que en primera instancia el router A toma su tabla de encaminamiento actual, le suma 1 a la columna de distancia y se la envía al router B el cual de dicha actualización toma el renglón de la red 10.0.0.0 la cual no alcanzaba y descarta la de la red 12.0.0.0 debido a que ya la tenía registrada con una distancia menor. Luego de esto el router B realiza la misma acción de sumar 1 al campo de distancia y envía la actualización al router C. Cuando C ya actualiza su tabla, vuelve a realizar dicha acción y se la envía a B, el cual actualiza sus datos y sumando 1 al campo distancia, se la envía al router A.

Luego de todas estas actualizaciones, las tablas de encaminamiento quedarían de la siguiente forma:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Dirección de red** | **Interfaz** | **Distancia** |  | **Dirección de red** | **Interfaz** | **Distancia** |
| 10.0.0.0 | F0 | 0 |  | 12.0.0.0 | S1 | 0 |
| 12.0.0.0 | S0 | 0 |  | 14.0.0.0 | S0 | 0 |
| 14.0.0.0 | S0 | 1 |  | 10.0.0.0 | S1 | 1 |
| 16.0.0.0 | S0 | 2 |  | 16.0.0.0 | S0 | 1 |
| *Tabla de encaminamiento del router A* | | |  | *Tabla de encaminamiento del router B* | | |

|  |  |  |
| --- | --- | --- |
| **Dirección de red** | **Interfaz** | **Distancia** |
| 14.0.0.0 | S1 | 0 |
| 16.0.0.0 | F0 | 0 |
| 12.0.0.0 | S1 | 1 |
| 10.0.0.0 | S1 | 2 |
| *Tabla de encaminamiento del router C* | | |

En conclusión es un algoritmo de convergencia lenta, y consume ancho de banda para enviar las actualizaciones, sin embargo es lo que se utilizaba en los inicios y funcionaba muy bien.

Tiene algunos problemas al caerse un enlace, estos suelen ser:

* + - Bucles, o loops: Como convergen lentamente pueden haber inconsistencias en las tablas y se crean estos bucles. Estos bucles se evitan mediante lo que se conoce como “Horizonte dividido”. Esto consiste en que si por ejemplo el router A le “enseña” al router B sobre la red 10.0.0.0, entonces cuando B le envíe actualizaciones al router A no le agregará ninguna información sobre dicha red, ya que fue A el que se lo enseño a B. Esta lógica se utiliza para toda la red y de esta forma se evitan los bucles.
    - Conteo al infinito: Si se comienza a enviar paquetes de actualizaciones cuando en la red hay un bucle, la métrica tiende a crecer haciéndose un conteo al infinito. Por esto mismo se crea un máximo para la métrica solucionando este problema.

Lógicamente la memoria RAM de los Routers es un recurso limitado, y si hay muchos Routers en la red (como en internet) se puede llenar dicha memoria con lo cual la red no sería convergente. Para evitar esto se recurre al encaminamiento jerárquico, cuando la red es muy grande se divide en trozos que se llaman Sistemas Autónomos (SA) y que son administrados por un solo ISP o proveedor de servicios, y dentro de estos sistemas autónomos se utilizan este tipo de algoritmos. Si todavía siguiera siendo grande la red, se vuelve a dividir, obteniendo una jerarquía de 3 niveles.

Además estos algoritmos utilizan temporizadores, los cuales son:

* + - De actualización: controla cada cuanto tiempo tengo que enviar la actualización de encaminamiento. Se puede configurar (por defecto es 30 seg.)
    - De espera: cuando un router avisa a otro que se cayó una red, el segundo en lugar de eliminar rápidamente dicha red de su tabla de encaminamiento, la marca y espera un tiempo determinado para corroborar que no se active nuevamente, si no es así la elimina.

**Algoritmo de encaminamiento de estado de enlace (o también SPF – Short path first):**

Este algoritmo surgió para mejorar el encaminamiento, al igual que el anterior es dinámico y distribuido, la diferencia principal es que envía actualizaciones solo ante cambios en la topología (ya se de alta, de baja o de cambios en la red), lo que contribuye a que la red converja mucho mas rápidamente.

Las actualizaciones no las envía por broadcast si no que las envía por multicast, esto es bueno ya que no consume procesamiento en las PC y también debido a que la dirección de grupo se auto configura al activar este protocolo de encaminamiento.

Además este algoritmo a diferencia del anterior, conoce la topología de la red, de forma que cada router puede armar un grafo que representa a la topología.

Funcionamiento:

1. **Descubrir al vecino:** cada router debe conocer cuáles son sus vecinos, es decir conocer a los Routers que están directamente conectados a él, con la finalidad de averiguar cuales están trabajando con el mismo protocolo. Esto lo hace enviando paquetes Hello por todas sus interfaces cada 10 segundos tratando de descubrir nuevas redes, cuando un router que lo recibe tiene el mismo protocolo, le manda otro paquete hello, con la IP que representa la id del router.
2. **Medir el retardo:** Envía paquetes echo, para conocer el retardo (distancia) que hay a cada vecino.
3. **Construir el Paquete de Estado de Enlace (LSP – link state packet):** lo construye con la información obtenida en el paso anterior. El paquete LSP contiene los vecinos directamente conectados, la IP que tienen y a qué distancia están.
4. **Inunda los LSP en la red:** Distribuye los paquetes LSP creados en el paso anterior, y los saca por todas sus interfaces, pero en el caso que reenvía un paquete LSP que le llega lo saca por todas las interfaces menos por la que le llegó. Se tiene que garantizar que todos los paquetes lleguen a todos lados. Esta inundación debe ser controlada, esto se hace colocándole al paquete LSP un número de secuencia o versión y un TTL.

El campo versión se utiliza debido a que si un router crea un paquete LSP y lo inunda en la red, el resto de los Routers lo van a “retransmitir” solo una vez, pero si a un router determinado le vuelve a llegar un paquete LSP que ya retransmitió, lo descarta a lo sumo que el campo versión sea superior al que re transmitió anteriormente, es decir que el paquete LSP es mas actualizado y el campo TTL se establece para que el paquete tenga un límite de saltos y no quede en la red indefinidamente dando vueltas.

1. **Arma un grafo de la red:** Una vez que se termina la inundación, todos los Routers tienen paquetes LSP de todos los demás Routers de la red, y con estos paquetes construye un grafo que representa la topología de la red.

Una vez que cada router tiene armado el grafo correspondiente a la red, con los “costos” de cada enlace calcula el camino más corto o la ruta más corta (dijkstra), y recién ahí construye la tabla de encaminamiento. El protocolo que utiliza este algoritmo, es OSPF.

Clase 06 de agosto

**SISTEMAS AUTÓNOMOS e INTERNET**

Internet es una red de redes interconectadas, los Routers de internet tienen tablas de encaminamiento donde cada renglón de dicha tabla tiene direcciones de red o de subred pero no de hosts. Como en internet no es viable que cada router tenga un renglón por cada red que hay en internet, debido a que la memoria RAM no sería suficiente, se implementa el encaminamiento jerárquico mediante la utilización de Sistemas Autónomos (S.A.). Un S.A. es un conjunto de Routers que son administrados y gestionados por la misma empresa u organización, normalmente una empresa de telecomunicaciones (por ejemplo Fibertel). Dentro de cada S.A. a su vez se utiliza el encaminamiento jerárquico, pero como no puedo configurar un algoritmo de encaminamiento jerárquico en un router, lo que se hace es configurarle un protocolo que utilice un algoritmo de este tipo.

Estos protocolos, dentro de Internet, se dividen en dos grandes grupos:

* + **Protocolos IGP:** Protocolo de Gateway interior, se implementan dentro de los sistemas autónomos, lo que hace que cada sistema autónomo pueda implementar protocolos distintos. Son ejemplos de este tipo los protocolos RIP, OSPF, ISIS y otros propietarios como el EIGRP.
  + **Protocolos EGP:** Protocolo de Gateway Exterior, se implementan para interconectar los distintos S.A. independientemente del protocolo IGP que implemente cada uno. Algunos ejemplos son BGPv4 (Border Gateway Protocol).

Cada router soporta más de un protocolo, pero puede suceder el caso en que para llegar a una red el router detecte que tiene 2 o más alternativas posibles. Es el administrador el que le indica que ruta almacenar en la tabla de encaminamiento, esto es conocido como *Distancia administrativa* y se define como la confiabilidad de una ruta. Las redes que están directamente conectadas tienen una distancia administrativa 0, o sea que si o si opto por esa opción. Las rutas estáticas definidas por el administrador, tienen una distancia admin. Igual a 1.

Por otro lado, los protocolos tienen distintas distancias administrativas, veamos las distancias de algunos:

* RIP = 120
* OSPF = 110

Entonces si algún router “aprende” a llegar a una red por RIP y por OSPF, lógicamente utilizará la de OSPF ya que tiene una menor distancia administrativa.

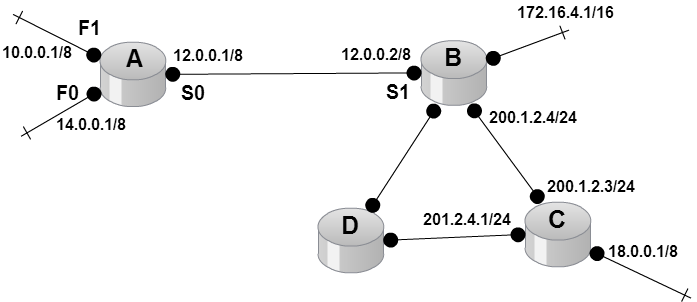
**Protocolo RIP (Routing Information Protocol):**

Algunas características:

* Fue el primer protocolo de encaminamiento que se usó cuando nació internet.
* Es un protocolo de vector de distancia, es decir que implementa el algoritmo de vector de distancia desarrollado anteriormente.
* La métrica que utiliza es la cantidad de saltos, debido a que cuando se creó internet todos los enlaces tenían el mismo ancho de banda (56k) entonces lo que realmente interesaba era la cantidad de saltos (cada vez que atraviesa un router se cuenta un salto).
* La cantidad máxima de saltos que soporta es 15, o sea que puede atravesar como máximo 15 Routers para llegar a la red de destino y si la red se encuentra a 16 saltos, se considera inalcanzable.
* El temporizador de actualización es de 30 segundos (por defecto, pero es modificable por el administrador)
* Actualmente hay 2 versiones:
  + RIP v1:
    - V.D. classfull: Singifica que soporta clases de IP (clase a, b, c)
    - No soporta VLSM ni CIDR
    - Soporta subredes pero con la misma mask.
    - En las actualizaciones no envía la máscara de subred.
    - Las actualizaciones las envía por broadcast
  + RIP v2:
    - V.D classless
    - Soporta VLSM y CIDR
    - Envía la máscara de subred en las actualizaciones
    - Envía actualizaciones por multicast a la dirección 224.0.0.9

Configuración de RIP:

Supongamos la siguiente red:



Las siguientes líneas son un ejemplo de cómo se configura RIP en los routers, pero en este ejemplo solo lo configuramos en el router A, el alumno debería realizarlo en los demás Routers:

R > enable 🡪 Se ingresa al modo privilegiado

R # configure terminal 🡪 Se ingresa al modo de configuración global

R (config) # hostname A 🡪 le cambio el nombre al router por A

A (config) # router rip 🡪 se le indica al router que utilizará el protocolo RIP

A (config-router) # network 10.0.0.0 🡪 publico la red 10.0.0.0 que está directamente conectada

A (config-router) # network 12.0.0.0 🡪 publico la red 12.0.0.0 que está directamente conectada

A (config-router) # network 14.0.0.0 🡪 publico la red 14.0.0.0 que está directamente conectada

A (config-router) # version 2 🡪 le indicamos que utilice RIP v2

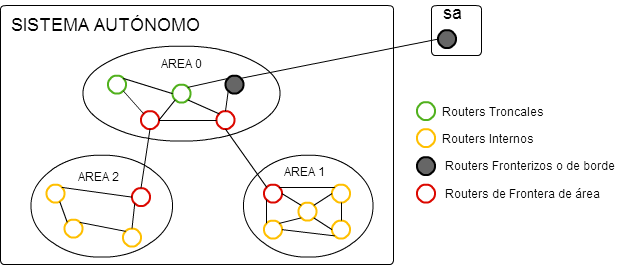
Hay que tener en cuenta que TODOS los Routers de la red tienen que utilizar la misma versión, o versión 1 o versión 2. Lo bueno de utilizar este protocolo es que configurado esto en todos los Routers, ante cambios en la red, las tablas de encaminamiento de los Routers, se autoactualizan.

Clase 11 de junio

**Protocolo OSPF (*Open Shortest Path First)*:**

* Es un protocolo público por lo que no pertenece a ninguna empresa en particular y además es del tipo IGP (de Gateway interior).
* OSPF es un software que se habilita y se configura en un dispositivo, y aplica la lógica del algoritmo de estado de enlace, es decir que lo implementa internamente. Este es mejor que el de vector de distancia ya que utiliza la métrica de “costo” de un enlace para calcular la ruta más corta.
* El costo de un enlace es el ancho de banda, pero si se utilizara el ancho de banda de cada enlace para elegir la ruta más corta mediante el algoritmo de dijkstra, el resultado sería la ruta con menor valor de ancho de banda. Como lo que se pretende es exactamente lo opuesto se define al costo como el cociente entre un ancho de banda de referencia (normalmente 100Mbps) y el ancho de banda real:
* Implementa balanceo de carga, es decir que ante caminos con igual métrica almacena los dos en su tabla de enrutamiento y envía algunos paquetes por un camino y otros por el otro.
* Como surge luego de los cambios que se hacen en internet, soporta VLSM y CIDR.
* Almacena los paquetes LSP de estado de enlace en una base de datos topológica.
* La utilización de este protocolo implica una rápida convergencia, pero si la red es muy grande las tablas de enrutamiento de los Routers crecen indefinidamente y también lo hacen las bases de datos topológicas, por lo que este protocolo permite la utilización de jerarquía e implementándola a través de áreas dentro del mismo sistema autónomo.

Veamos un ejemplo de un sistema autónomo dividido en áreas, para conocer los distintos tipos de Routers según su funcionalidad:



Este sistema dividido por áreas es útil ya que se va a reducir la base de datos topológica de cada área, ya que cada router de un área va a conocer solamente la topología de su área y como llegar a las otras áreas, pero no a los Routers de las otras áreas. La comunicación entre áreas es jerárquica, es decir toda comunicación debe atravesar el área 0 o troncal para comunicarse con otro S.A, y cada área tiene su propia base de datos topológica.

Los Routers se clasifican en:

* + - Routers Troncales: Todos los Routers que no sean ni fronterizos ni de frontera de área y que pertenecen al área troncal.
    - Routers Internos: Son todos los Routers que no sean de frontera de área y no pertenezcan al área troncal.
    - Routers fronterizos o de borde: Son los que permiten conectarse con otros sistemas autónomos.
    - Routers de frontera de área: Routers que se conectan con el área troncal.}

Para que el OSPF funcione en los enlaces punto a punto, aplica la lógica del algoritmo de estado de enlace, cada cierto tiempo envía paquetes hello por todas sus interfaces, tratando de detectar si sus vecinos están activos.

En el caso de que la red sea de multiacceso por difusión, todos los Routers deberían crear adyacencias de todos con todos, para evitar consumo de ancho de banda debido a las multiples adyacencias, pero para evitar esto todas las actualizaciones de encaminamiento en lugar de enviárselas entre ellos, se las envía a lo que se conoce como DR o Router designado (*del inglés Designed Router)* y éste es el encargado de enviárselo a todos los otros. Además en caso de que el router designado falle, existe un BDR o Router designado de respaldo *(Backup Designed Router)*, el cual lo reemplaza en caso de que el principal no funcione. El DR y el BDR son elegidos por el mismo protocolo, en base al identificador de router, que en caso de que no se le haya asignado elige como id a la dirección IP mas “alta” que tenga en sus interfaces).

Por otro lado, este protocolo implementa seguridad mediante dos métodos. En primer lugar utiliza la autenticación de Routers, lo que significa que un router sólo recibe actualizaciones de otro siempre y cuando lo tenga autenticado o registrado. Por otro lado utiliza cifrado o encriptación de actualizaciones completas (implica más lentitud).

Actualmente existen dos versiones de OSPF, OSPF v2 que se utiliza con IPv4 y OSPF v3 la cual corre en IPv6.

Las direcciones de grupo que utiliza son dos: 224.0.0.5 y 224.0.0.6.

Configuración:



RA # configure terminal 🡪 Configuramos la terminal

RA (config) # router ospf 15 🡪 El nro. es una norma general

RA (config-router) # network 192.168.1.64 0.0.0.31 area 0 🡪 dir de red + mascara wildcard + área

RA (config-router) # network 192.168.2.0 0.0.0.255 area 0 🡪 dir de red + mascara wildcard + área

RA (config-router) # network 200.1.2.64 0.0.0.3 area 0 🡪 dir de red + mascara wildcard + área

La máscara de wildcard es la inversa de la máscara de red, cambio los 1 por los 0 y los 0 por los 1.

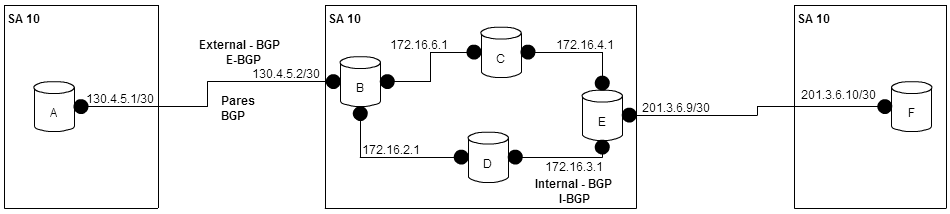
Clase 20 de Agosto

**Protocolo BGP (Border Gateway Protocol – Protocolo de Gateway de Borde):**

En la actualidad se está usando la versión BGPv4

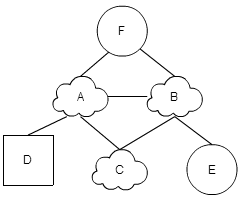
**Características:**

* **Protocolo EGP:** Este protocolo se dice “de borde” porque es del tipo de protocolo EGP. Anteriormente se estudió que tenemos dos tipos de protocolos los IGP y los EGP. Los primeros son los que se ejecutan adentro de un sistema autónomo, en cambio los segundos son los que permiten interconectar sistemas autónomos entre sí. Este protocolo se va a ejecutar en un router de borde, esto significa que es un router que hacia un lado está conectado al interior de la empresa y hacia el otro está conectado al exterior a otro sistema autónomo. Este protocolo permite interconectar sistemas autónomos diferentes.
* **Algoritmo de vector de ruta:** Este protocolo utiliza en su lógica el algoritmo de vector de ruta que recibe su nombre debido a que las actualizaciones de encaminamiento que se van a intercambiar entre los routers dan la ruta completa que hay que seguir para llegar a una determinada red. Es decir, y en comparación a los anteriores algoritmos, no expresa que para llegar al destino hay que dar cierta cantidad de saltos, sino que indica que para llegar al destino se debe pasar por el router A, luego por el router B, etc.

****

Considerando la figura antes presentada podemos ver que hay 3 sistemas autónomos, uno de los cuales tiene cuatro routers que ejecutan un protocolo IGP. Este algoritmo va a intercambiar actualizaciones de encaminamiento entre pares BGP, que son dos routers que pertenecen a sistemas autónomos distintos y se ponen de acuerdo para intercambiar actualizaciones de encaminamiento. Estos pares BGP tienen una conexión punto a punto. Como las actualizaciones que se van a intercambiar estos pares es muy importante entre ellos se establece una conexión TCP (de la capa de transporte) que es una conexión completamente fiable para la transmisión de actualizaciones de encaminamiento. La información que se van a intercambiar en las actualizaciones son los vectores de ruta antes mencionados, y dicho intercambio se dará de manera regular cada 30 segundos.

* **Tipos de Redes:** Este protocolo maneja distinto tipos de redes:



* **Stub (Conexión única):** Las redes stub se las suele llamar de conexión única porque están conectados hacia un sistema autónomo una sola vez. En otras palabras podemos decir que son todos aquellos sistemas autónomos que tienen una sola conexión a las redes de tránsito. En la gráfica el sistema autónomo D encerrado en un cuadrado es un Stub, que se va a conectar a través de una única conexión hacia otro sistema autónomo para tener conectividad con todos los otros sistemas autónomos. Si se cayera el enlace que los une, el sistema autónomo Stub queda completamente aislado (pierde conectividad hacia el exterior) si bien su red interna sigue funcionando.
* **Multihomed:** Son sistemas autónomos que se conectan por más de un enlace, es decir que están conectados con 2 o más sistemas autónomos como mínimo. Este tipo de sistema autónomo tiene más fiabilidad porque si se cae un enlace aún se encontrara conectado con el otro sistema autónomo. En la gráfica están representados por círculos: E y F.
* **De transito:** Son sistemas autónomos que presentan la característica de no generar trafico internamente sino que permiten que el trafico pase a través de ellos, es por esto que reciben el nombre de “de transito”. El fin de estos sistemas autónomos es permitir que fluyan a través de ellos paquetes que provienen de otros sistemas autónomos y por supuesto cobran por brindar ese servicio. En la grafico están representados por nubes: A, B y C.
* **Basado en políticas:** Existen políticas negativas y también positivas que son configuradas por los administradores. Como dijimos, BGP es un protocolo que permite interconectar toda la internet entre los distintos sistemas autónomos. Sin embargo puede existir un sistema autónomo de transito perteneciente a un proveedor de servicios en particular que no quiera que por el atraviesen paquetes que provengan de un sistema autónomo en particular, y para ello es necesario definir una política que establezca dicha restricción (en este caso hablamos de una política negativa). Todas estas políticas se configuran en el BGP.

Por ejemplo, supongamos que el administrador decide que no quiere que se encaminen paquetes por el sistema autónomo numero 1500 porque pertenece a la competencia. Esta política negativa se le configura al router, entonces cuando el BGP aprende como llegar a una misma red por distintos caminos lo que hace es borrar o filtrar el aprendizaje que pasa por la ruta que está restringida y conserva el resto.

Supongamos que se sabe cómo llegar a una red por tres caminos distintos, ¿Cuál elije el BGP para encaminar los paquetes? La métrica que utiliza BGP es la cantidad de saltos de sistemas autónomos, por lo tanto elegirá aquel camino que atraviese la menor cantidad de los mismos.

Un ejemplo de una política positiva, también conocidas como preferencias, puede ser que si es posible se atraviese un determinado sistema autónomo. Entonces, si existen dos maneras de encaminar y una de ellas pasa por el sistema autónomo preferido y la otra no, por más que sea más largo se elige el que respeta la preferencia. Es decir que la prioridad la tienen las preferencias. En resumen, si no hay una política definida se elige el camino más corto, mientras que si hay definida alguna política positiva si o si hay que respetarla por más que el camino que la cumpla sea el de mayor cantidad de saltos. Si se presentara el caso en el que hay dos políticas preferenciales y existen dos caminos que las satisfacen, en ese caso se elige al camino más corto que respete cualquiera de las dos políticas.

**Actualización de encaminamiento:**

Una actualización se intercambia entre dos pares BGP y la transferencia es punto a punto. Estas actualizaciones están conformadas por tres parámetros:

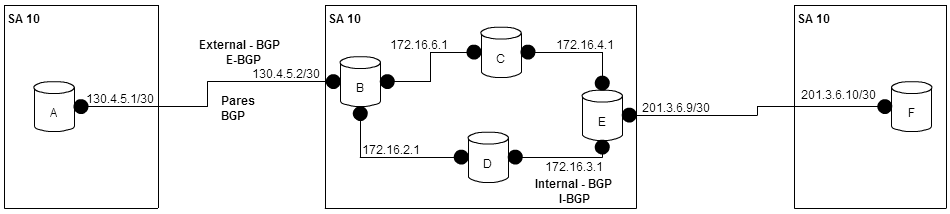
* **Dirección de red (formato CIDR):** Un ejemplo de una dirección de red en formato CIDR puede ser: 200.2.4.0/22. Una dirección es formato CIDR es una dirección IP a la cual se le agrega una designación de prefijo después del byte de la derecha para definir la parte de red de la dirección. Los routers de borde se encargan de hacer un resumen de ruta, esto significa que en lugar de mostrar las direcciones IP de todas las redes que están conectados a él lo que hacen es resumir todas esas direcciones en una sola, ya que por internet no puede haber información de tantas redes y ocupa menos espacio en las tablas de encaminamiento. Esta dirección de red resumida es la que se le publica a los otros sistemas autónomos.
* **Lista de sistemas autónomos (ruta o path):** Consiste en una lista de los sistemas autónomos que hay que atravesar para llegar a la dirección de red (por eso utiliza un algoritmo de vector de ruta). Es una lista de números, por ejemplo: 150, 2000, 20, 14 donde cada número representa un sistema autónomo.

Este protocolo garantiza que no haya bucles debido a que cuando a un sistema autónomo le pasan la lista de sistemas autónomos a atravesar o “ruta”, el mismo se fija que no haya valores duplicados de sistemas autónomos y de ser así los elimina. De esta manera nunca se pasara dos veces por el mismo sistema autónomo.

* **Dirección del próximo salto:** Es la dirección a la cual se le deben entregar los paquetes

Los números de los sistemas autónomos son todos públicos, es decir que no pueden haber dos que estén repetidos en el mundo, son globalmente únicos. Se los suele llamar NSA (Number System Autonomo – Numero de Sistema Autónomo). Por el momento tienen 16 bits y se está pensando en agrandarlos.

**Configuración: (ESTO NO SALE EN NINGUN LIBRO DE CATEDRA)**

****

La configuración debe realizarse en los router A, B y E, F que son los pares BGP. Estos routers son de borde porque hacia adentro manejan un protocolo (RIP, OSPF) y hacia afuera manejan otro protocolo distinto. A su vez el protocolo BGP se divide en dos: el I-BGP (Internal BGP – BGP Interno) y el E-BGP (External BGP – BGP Externo). El E-BGP permite a los routers definidos como pares BGP que pertenecen a sistemas autonomos distintos intercambiar información, mientras que el I-BGP les permite intercambiar información a los routers que están definidos como pares BGP y que a su vez pertenecen al mismo sistema autónomo.

A continuación se realizara la configuración de los sistemas autónomos de la siguiente representación.

Nota: en negro se colocan los comandos que van a ejecutarse siempre mientras que en color azul son los valores que pueden variar.

Configuración router A:

**RA(config)# route bgp 1000** 🡪 1000 es el número del SA del router que se está configurando.

**RA(config-router)# neighbor 130.4.5.2 remote-as 2000** 🡪se define el par BGP(vecino) con su IP y nro de SA (Router B)

**RA(config-router)# network <IP de red> mask <mascara>** 🡪 aquí se colocan las direcciones de red en formato CIDR que se quieren publicar

Configuración router B:

**RB(config)# route bgp 2000**

**RB(config-router)# neighbor 130.4.5.1 remote-as 1000** 🡪 Par BGP (vecino) Externo router (A)

**RB(config-router)# neighbor 172.16.4.1 remote-as 2000**  🡪Configuración BGP Interno router (E), nótese comparte el mismo número de SA que el Router que se está configurando. También se podría utilizar la IP de la otra interfaz del router E: 172.16.3.1

**RB(config-router)# network <IP de red> mask <mascara>**

Nota: cuando el número de SA es el mismo que el router que se está configurando estamos en presencia de I-BGP, mientras que cuando los números son distintos estamos configurando un E\_BGP

Configuración router E:

**RC(config)# route bgp 2000**

**RC(config-router)# neighbor 201.3.6.10 remote-as 3000**

**RC(config-router)# neighbor 172.16.6.1 remote-as 2000**

**RC(config-router)# network <IP de red> mask <mascara>**

FIN DEL ESTUDIO DEL ENCAMINAMIENTO

**CONTROL DE CONGESTIÓN:**

**Congestión:**

¿Cuándo se produce congestión? Cuando la cantidad de paquetes que ingresan a la red superan la capacidad de procesamiento de la misma, es decir cuando tenemos más información para encaminar que la capacidad de la red para encaminar. Cuando un router recibe más de lo que puede manejar se empiezan a aumentar las colas, se llenan los buffers y la red comienza a congestionarse. Si no se hace nada al respecto la red puede caerse (deja de funcionar). Por este motivo existen algoritmos y técnicas para tratar de controlar la congestión.

Los dispositivos que intervienen en la cogestión son los routers y los Host. Los primeros tienen como función encaminar paquetes de datos y para poder hacerlo bien debe intercambiarse actualizaciones de encaminamiento con los routers vecinos lo que introduce más tráfico de paquetes a la red. Los segundos son los encargados de incorporar paquetes a la red, para lo cual lo hacen de a ráfagas. Es decir, en las redes de datos las transmisiones no se dan de manera regular o constante sino que se dan de a ráfagas, porque puede ser que estemos leyendo algo en internet inactivos y de repente queramos mandar un mail, o abrir un video o una página, etc. El hecho de que la transmisión sea de esta manera produce congestión porque la red no está preparada para soportar que todas las personas transmitan al mismo tiempo, por ejemplo. Los encargados de detectar la congestión son los routers.

**Causas de la congestión:**

* **Poca memoria RAM en los routers:** Si hay un router que tiene poca memoria RAM la red se puede llegar a congestionar ya que los routers tienen muchas interfaces y por cada una de ellas pueden salir o entras paquetes por lo que se forma para cada interfaz una cola de entrada y una cola de salida. Estas colas físicamente se almacenan en un buffer que es un trozo de RAM. Por lo tanto, si un router tiene poca memoria RAM sus buffers van a ser chicos. Además, si las interfaces están conectadas a enlaces de alta velocidad, por los que ingresan muchos paquetes, las colas de entrada se van a llenar rápidamente. Si una vez que están llenas siguen llegando paquetes los mismos van a ser descartados porque no hay lugar físico para almacenarlos.
* **Poca capacidad de CPU:** El rendimiento de los procesadores de los routers se miden en PPS (Packets Per Second – Paquetes procesados Por Segundo).

Los routers son dispositivos de almacenamiento reenvío, esto significa que los paquetes entran por una interfaz y son almacenados en una cola, cuando el dispositivo tiene tiempo va tomando los paquetes de la cola, se fija cual es la dirección IP de destino, consulta su tabla de encaminamiento, conmuta por la interfaz que debe salir ese paquete y lo coloca en la cola de salida de esa interfaz. El paquete va a esperar en cola hasta que se desocupe la interfaz y los bits puedan ser transmitidos. Por lo tanto, si el micro de un router es lento o de poca capacidad, por más que tenga mucha memoria RAM y pueda tener colas grandes, el router va a demorar mucho tiempo en tomar un paquete de la cola, procesarlo y conmutarlo lo que va a producir que las colas se vayan llenando cada vez más hasta que se congestionen.

* **Enlaces de baja velocidad:** Por más que el router procese rápido todos los paquetes que entran, los mismos se van a ir acumulando en los buffers porque no van a poder ser transmitidos inmediatamente debido a que el enlace por el que viajan no tiene mucha capacidad, produciendo congestión en los buffers.
* **Modernización incompleta:** Supongamos que actualizamos toda la tecnología de una red (routers, enlaces, etc.) pero dejamos un enlace sin actualizar, de baja velocidad. Lo que va a suceder es que los paquetes van a ser procesados y transmitidos a altísimas velocidad por todos los enlaces, pero al llegar al enlace que no fue actualizado la velocidad de transmisión bajara mucho produciendo un cuello de botella. Es como viajar por una autopista, mientras vamos por la misma circula un gran flujo de autos a alta velocidad, pero cuando se acaba la misma se generan colas debido a la disminución de velocidad. Esto es algo que se nota los días u horarios en que hay mayor demanda.

La solución a la congestión es lograr una muy buena infraestructura, pero el problema es que esto implica gastar mucho dinero.

**Diferencia entre control de congestión y control de flujo:**

En el control de congestión intervienen todos: los dispositivos que inyectan paquetes a la red (computadoras), servidores, routers, etc. y lo que se trata de controlar es que los paquetes no se “frenen” en la red. Si tengo muchas personas que están tratando de enviar paquetes a la red, la congestión se podría controlar haciendo que los usuarios disminuyan la tasa de transmisión. La disminución de la tasa de transmisión implica que se pueden seguir lanzando paquetes a la red pero disminuyendo la cantidad de los mismos, por ejemplo en lugar de lanzar 1000 se lanzaran 500 paquetes. Es decir que se le indica a los dispositivos que incorporen menos cantidad de paquetes, una menor tasa, a la red para darle tiempo a los routers de procesarlos.

La idea es que si un Host manda un paquete a un router que esta congestionado, dicho paquete no llega a destino y por lo tanto el Host va a mandar otro paquete más que es un acuse de recibo. Como el acuse de recibo tampoco va a llegar al router, el Host después de un tiempo manda otro paquete más de acuse de recibo, y de esta manera se empeora la situación ya que encima que la red esta congestionada se sigue transmitiendo paquetes.

Por otro lado, el control de flujo se hace extremo a extremo, es decir que únicamente se implementa entre Hosts (afecta únicamente a los dispositivos que se están queriendo comunicar). El control de flujo debe implementarse cuando el receptor es más lento que el emisor, o dicho de otra forma cuando el receptor tiene menor capacidad de procesamiento que el emisor. La idea es que un Host rápido no sature a un receptor lento. El control de flujo se implementa en la capa 4 de OSI que es la capa de transporte, a través del campo ventana. A través de este campo el receptor le dice al emisor que le envíe una menor cantidad de paquetes o que no le mande nada (esto ocurre cuando el campo ventana está en cero), por lo tanto el emisor no le va a enviar paquetes al receptor hasta que este haya vaciado su buffer.

**Principios generales del control de congestión:**

Existen dos técnicas para controlar la congestión:

* **De ciclo abierto:** En una técnica de ciclo abierto no hay retroalimentación de la red. Es decir que se van a tomar decisiones y se van a aplicar normas para controlar la congestión independientemente del estado de la red (si la red está ocupada o desocupada).

Pongamos un ejemplo: supongamos que queremos controlar que nunca se congestione una calle, y para ello ponemos un policía en la intersección entre dos calles. Este policía va a controlar que solamente pase un auto cada un minuto para evitar la congestión, y lo va a hacer durante las 24 horas de todos los días, independientemente de si es una hora pico o no. Lo que intenta evitar esta técnica es el problema de que los paquetes ingresan a la red como ráfagas, entonces si se controla todo el tiempo la tasa de ingreso de paquetes se evitaría dicho inconveniente.

* **De ciclo cerrado:** En esta técnica si hay una retroalimentación de la red, es decir que se evalúa cual es el estado de la red y en base a ello se toman medidas. Estos algoritmos tratan de ver si hay congestión, y para ello se requiere monitorear continuamente la red. Básicamente consta de tres etapas:
* **Monitorear la red:** la idea de monitorear la red es intentar detectar si hay congestión o no. Para ello tenemos distintas alternativas:
* **Controlar la longitud de las colas de las interfaces de los routers:** Consiste en analizar cual es la longitud de las colas en los routers, ya que si notamos que las mismas están continuamente creciendo es posible que la red se esté congestionando en algún lado, y de seguir así la situación podría llegar a llenar el buffer y por lo tanto tener que descartar paquetes.
* **Vencimiento de los temporizadores:** Si una computadora manda un paquete con un temporizador se debe recibir un acuse de recibo dentro del tiempo establecido. Si no le llega al emisor dicho acuse es posible que sea porque la red esta lenta, y frente a esta situación el dispositivo podría mandar su paquete nuevamente pero esto haría que se ponga aún más lenta la red. En lugar de eso, el Host lo que hace es ver los temporizadores para ver si se vencieron o no, en caso de que así sea puede suponer que la red esta congestionada y por lo tanto puede colaborar para superar dicha situación mandando menos paquetes. Por lo tanto, el Host analizando sus temporizadores puede detectar que la red esta congestionada (esto se llama ventana de congestión) y disminuir la tasa de paquetes que envía (por ejemplo, en lugar de 1000 paquetes envía 600).
* **Retardo de los paquetes:** Se puede analizar el retardo de los paquetes. Si se detecta que el retardo es muy alto puede deberse a que la red este congestionada.
* **Informar acerca de la congestión:** Cuando se detecta la congestión se le debe informar a la Pc que está emitiendo, y sabe cuál es esa Pc por la IP origen del paquete. La forma de avisarle al emisor es enviándole un paquete que se llama Source Quench o Apaciguamiento en el origen, a través del cual le dice que la red esta congestionada y que baje su tasa de transmisión.
* **Tomar acciones correctivas:** Una vez que se informó al origen la situación es necesario tomar medidas correctivas para solucionar la situación.

Clase 27 de Agosto

**Métodos de control de congestión de ciclo “cerrado”:**

En el control de congestión de ciclo cerrado existe una retroalimentación de la red, por lo tanto tenemos que estar continuamente monitoreando si se produce congestión o no. Si se detecta que efectivamente existe la congestión se debe informar a los dispositivos que estaban vigilando la misma y además se deben tomar medidas correctivas para superar dicho inconveniente.

Diferencia entre redes de circuitos virtuales y datagrama: En las redes de circuitos virtuales todos los paquetes viajan por la misma ruta ya que hay un paquete que va “abriendo el camino” haciendo lo que se conoce como establecimiento del circuito, y luego se registran esas direcciones en las tablas de todos los dispositivos intermedios de manera que cuando se lanza el resto de los paquetes todos viajen por el mismo camino físico. En contrapartida, en las redes de datagrama cada paquete se encamina de manera independiente, es decir que pueden viajar por caminos físicos distintos (conmutación de paquetes pura), por más que todos los paquetes pertenezcan a una misma transferencia de archivos.

* **Control de congestión en redes de circuitos virtuales:** Si se detecta que hay congestión en una red directamente no se aceptan o establecen nuevos circuitos virtuales, lo que se conoce como **control de admisión**. Otra solución es aceptar el circuito virtual pero el camino que siga la comunicación debe ser alejado del camino en donde existe la congestión, ya que es raro que toda la red este congestionada sino que por lo general solo se congestiona una parte. Dado el caso en el que la red no estuviese congestionada se aceptan todos los circuitos virtuales.

La ventaja que presentan los circuitos virtuales es que, para que un dato se pueda transmitir extremo a extremo, se puede reservar ancho de banda de los enlaces por donde fuera a atravesar la información. También se puede reservar ciclos de procesamiento de CPU de los routers, es decir se reserva un porcentaje de uso del micro del router (recordamos la capacidad de CPU de un router se mide en PPS – paquetes procesados por segundo). Por último, también es posible reservar espacio en los buffers ya que todos los paquetes cuando llegan a un router son almacenados en colas que físicamente son una porción de memoria RAM del router. De esta manera, se puede establecer una reserva de memoria en los routers por todos el camino físico por el que van a viajar los paquetes para que la información no se descarte. Esto es posible de realizar en los circuitos virtuales porque se tiene la certeza de que todos los paquetes de la comunicación van a viajar por el camino físico.

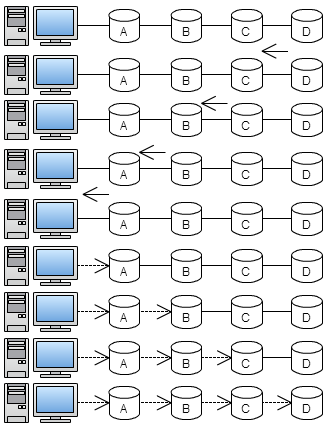
Las reservas antes mencionadas solamente se hacen en caso de que la red esta congestionada, de no ser así no conviene hacer reservas porque subutiliza la red, por ejemplo para que reservar ancho de banda si tal vez no lo use todo el tiempo ya que la transferencia de datos se da de a ráfagas y no de manera constante.

* **Control de congestión en redes de datagramas:** En este caso es más difícil controlar la congestión ya que los paquetes van a seguir caminos distintos y es muy difícil evitar pasar por la parte que esta congestionada. Por lo tanto, cada router trabaja de manera autónoma e independiente y encamina en función de sus tablas de encaminamiento, es decir que no tiene en cuenta si los routers vecinos se están congestionando o no. Los routers tienen la obligación de controlar si se están congestionando o no, y para hacerlo, cada cierto periodo de tiempo, realiza un análisis de cada una de los buffers de las interfaces que tiene. Para realizar este análisis se define un límite o umbral de paquete en cola, en donde si la cola de paquetes llega o supera dicho límite (se analiza la longitud de la cola), la interfaz del router que está asociada con ese buffer entra en un **estado de advertencia**. El estado de advertencia significa que si no se toma ninguna medida con respecto a esa interfaz, la misma puede llegar a congestionarse totalmente, es decir que se llena el buffer y empieza a descartar paquetes porque no tiene lugar físico para almacenarlos.

**Formas de informar la congestión:**

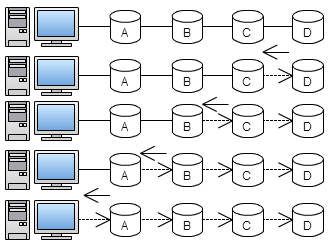
Una vez que se detectó la congestión de la manera antes explicada, se debe informar acerca de la misma tanto al emisor como al receptor, y a este último para que sepa que puede haber demoras en la recepción de paquetes. Para informar existen dos formas:

* **Bit de advertencia:** El Bit de advertencia es un campo que se encuentra en las cabeceras de muchas tecnologías (por ejemplo frame relay) e informan que hay o que se está produciendo la congestión. Consiste en dos bits uno de ellos llamado FECN y el otro BECN, en donde las últimas tres letras ECN significan Explicit Congestion Notification – Notificación de Congestión Explicita. La F de FECN significa Forward o “hacia adelante” y la B de BECN significa Back o “hacia atrás”. Entonces, cada vez que un dispositivo detecta congestión prende un bit que indica el sentido de la trama (si va hacia adelante prende un bit, si va hacia atrás prende otro). La idea es informar tanto al destino de la transmisión como al destino de la misma que hay congestión. Cuando el router detecta la congestión de la interfaz, enciende el bit de la trama de datos del paquete y el mismo sigue viajando como tenía previsto. Cuando el paquete llega a destino, éste se entera de que hay congestión y por lo tanto al enviar la respuesta de ese paquete al origen también enciende el bit para que el emisor se entere de que hay congestión (por ello hay dos bits). Cuando ambos, destino y origen, se enteraron de que hay congestión lo que hacen es disminuir la tasa de transferencia de información. Si continúan recibiendo dicho bit indicando la congestión continúan reduciendo la tasa de transferencia, y una vez que ya no lo reciben comienzan a aumentar la misma pero de manera gradual (ya que si se envía de golpe toda la información la red puede volver a congestionarse).
* **Paquetes reguladores:** cuando un router detecta que una interfaz se está congestionando, la interfaz involucrada entra en estado de advertencia. El dispositivo va a construir un paquete extra ICMP del tipo Source Quench o Apaciguamiento en origen (que es el paquete regulador), que va a tener como IP origen la IP del router y como IP destino la de la Pc emisora (el origen). El paquete regulador lo que hace es notificar de la congestión al emisor para que baje su tasa de transferencia de información. Si el emisor sigue recibiendo paquetes reguladores continúa disminuyendo su tasa de transferencia, y si pasado un tiempo ya no los recibe comienza a aumentar la misma de manera gradual. Para implementar el paquete regulador hay dos formas:
* **Paquete regulador que afecta al origen:** Consideremos la siguiente figura:



Como se ve en la figura existen 4 routers y una Pc. Además, las líneas negras representan transmisión de datos a un 100% de velocidad, mientras que las flechas punteadas representan una transmisión de datos con una tasa disminuida. Se presenta una situación en donde el router D detecta que una de sus interfaces esta congestionada y en consecuencia construye un paquete regulador que se lo va a enviar a al router C (el paquete es representado en la imagen por una flecha continua). Mientras esto ocurre, el host sigue transmitiendo datos porque todavía no le fue notificada la congestión y por lo tanto no sabe que debe disminuir la tasa de transferencia de datos. Luego, en el siguiente paso se muestra como el router C recibe el paquete regulador y se lo envía al router B. Todas las instancias continúan hasta que el paquete regulador llega a la Pc y mientras esto ocurrió la misma continua transmitiendo información normalmente. Como vemos, demora mucho tiempo en que el host se entere de la congestión (situación que no colabora con el pobre router D que esta sufriendo el problema), y ahora si una vez que ya conoce cuál es la situación comienza a disminuir su tasa de transmisión. Tuvieron que pasar 8 instancias hasta que el flujo se redujo completamente, ya que si bien la Pc bajo su tasa de transmisión algunas instancias antes, los mensajes que habían sido enviados con anterioridad siguieron fluyendo a una tasa de transferencia alta.

* **Paquete regulador salto por salto:** Consideremos la siguiente figura:

Aquí se presenta la misma situación que en el caso anterior, y se realizan las mismas salvedades con respecto a la interpretación de las líneas. El router D identifica que una de sus interfaces esta congestionada y en consecuencia construye un paquete regulador y se lo envía al router C. El router C para tratar de aliviar el problema del router D, al instante va a absorber todo el tráfico que le llega y es dirigido a D, y comienza a regular el tráfico enviándole los paquetes a una velocidad mucho más baja (de aquí proviene el nombre de salto por salto). El requisito para que C pueda ejecutar la acción antes mencionada es que tenga mucha memoria, es decir que sus buffers sean grandes porque ahora la congestión le fue pasada a él ya que se va encargar de absorber toda la transferencia, y de lo contrario (si no tuviera sus buffers grandes) también se congestionarían sus interfaces por almacenar muchos paquetes. Luego, C emite el paquete ICMP hacia el router B, y este último adopta la misma conducta absorbente que C con respecto a la regulación de la transmisión. El proceso se repite hasta que el paquete regulador ICMP llega al host, quien al recibirlo se entera que hay congestión y comienza a disminuir su tasa de transferencia. En definitiva, la idea es aliviar rápidamente la transmisión de datos salto por salto, ya que por un determinado periodo de tiempo los routers van absorbiendo la información que es transmitida a una alta tasa. En este caso tomo tan solo 5 instancias para solucionar la congestión.

* **Descarte de paquetes (desprendimiento de carga):** Supongamos que el router D presentado en la imagen anterior se congestiona, crea y envía el paquete regulador al Host, pero pasa mucho tiempo en disminuir la tasa de transferencia y en consecuencia su buffer se llena. En este caso existen varios enfoques para el descarte de paquetes:
* **Transferencia de archivos:** Supongamos que se está realizando la transferencia de un archivo grande que está dividido en muchos paquetes, en este caso lo que importa es la fiabilidad, es decir asegurarse de que los paquetes lleguen por lo que la transmisión estará montada sobre TCP. Al router que convendrá descartar aquellos paquetes que tengan un mayor número de secuencia (los últimos que van llegando), porque si descarto el que tiene menor número de secuencia, cuando el origen no reciba el acuse de recibo comenzara a retransmitir nuevamente desde el primer paquete. En cambio, si los que se descartan son los últimos paquetes, el origen únicamente tendrá que retransmitir estos últimos que fueron descartados (ya que los primeros fueron recibidos).
* **Videoconferencia:** Supongamos que se está transmitiendo una videoconferencia, en este tipo de servicio no interesa la fiabilidad, sino que lo más importante es la rapidez de la transferencia y por lo tanto la comunicación estará montada sobre UDP (aquí no hay acuse de recibo ni retransmisiones). En la transferencia de video se transmiten muchos cuadros por segundo, y generalmente cuando se graba video siempre hay un cuadro completo al cual se le van realizando desplazamientos. Por lo tanto, cada cierto periodo de tiempo se transmite el cuadro completo, pero lo que más se envían son los desplazamientos. Supongamos que llegan al router un paquete que pertenece al cuadro completo y un paquete que pertenece a la actualización o desplazamiento de la imagen, ¿Cuál le conviene descartar? En este caso es conveniente descartar la actualización y no el cuadro completo, ya que este último contiene toda la imagen y el primero solamente es una parte del cuadro completo.
* **Prioridades:** Se le pueden definir prioridades a los paquetes. Los paquetes que pertenecen a una transmisión en tiempo real tienen prioridad alta, mientras que las transferencias de archivos, correo y pagina web presentan una baja prioridad. Entonces, para el router es más importante mantener aquellos paquetes provenientes de una transmisión en tiempo real que los que provienen de una transferencia de archivos, considerando que los últimos utilizan el protocolo TCP que envía acuses de recibo y permite la retransmisión. Además, es un requisito de la transferencia en tiempo real que los paquetes lleguen rápido al destino. En resumen, los paquetes que tienen alta prioridad no se descarta y los que tienen baja prioridad si son descartados porque van a ser retransmitidos.

**QoS (Quality of service – Calidad de Servicio):**

**Flujo de datos:** Un flujo de datos son datos que se mueven de un origen a un destino. Es un conjunto de información que se mueve desde un Host hasta otro Host, atravesando una red.

Cada flujo de datos presenta distintos requerimientos o características de red (si bien todos se transmiten por el mismo medio), ya que no es lo mismo transmitir un correo electrónico que video.

La calidad de servicio hace referencia a que la infraestructura de la red brinde servicios diferenciados en función de lo que se está transmitiendo en un momento determinado. Antes no se brindaba la calidad de servicio, sea lo que sea que se transmita se colocaban los paquetes en una cola FIFO (el primero en llegar es el primero en salir) y eran lanzados a la red.

Los flujos de datos pueden caracterizarse a través de distintos requerimientos para poder brindarles una adecuada calidad de servicio:

* **Confiabilidad**
* **Ancho de banda**
* **Retardo:** El retardo significa que los paquetes llegan tarde pero siempre se atrasan en la misma cantidad de tiempo.
* **Fluctuación o variabilidad del retardo (Jitter):** la variabilidad del retardo hace referencia a aquellos paquetes que llegan tarde pero a veces demoran más y otras veces menos. Algunas aplicaciones son sensibles a la variabilidad del retardo.

A continuación se presentan algunas aplicaciones y su caracterización a través de los distintos requerimientos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **APLICACIÓN** | **CONFIABILIDAD** | **RETARDO** | **FLUCTUACION** | **ANCHO DE BANDA** |
| **Correo electrónico** | **Alta** | **Bajo** | **Baja** | **Bajo** |
| **Transferencia de archivos** | **Alta** | **Bajo** | **Baja** | **Medio** |
| **Acceso a la web** | **Alta** | **Medio** | **Baja** | **Medio** |
| **Inicio sesión remoto** | **Alta** | **Medio** | **Media** | **Bajo** |
| **Audio bajo demanda** | **Baja** | **Bajo** | **Alta** | **Medio** |
| **Video bajo demanda** | **Baja** | **Bajo** | **Alta** | **Alto** |
| **Telefonía** | **Baja** | **Alto** | **Alta** | **Bajo** |
| **Videoconferencia** | **Baja** | **Alto** | **Alta** | **Alto** |

Aquellas aplicaciones que tienen una alta confiabilidad significa que son muy exigentes en cuanto a la confiabilidad y requieren que todos los paquetes que envían lleguen al destino (de manera ordenada), y en caso de que algún paquete no lo logre se desea que sea retransmitido. Estas aplicaciones están montadas sobre TCP. Las aplicaciones que tienen baja confiabilidad no presentan mucho inconveniente si se pierde algún paquete en la transmisión, y no necesitan que sean retransmitidos porque se perderían de contexto.

Tanto el correo electrónico como la transferencia de archivos tienen un bajo retardo porque no requieren que la transferencia de datos se haga de manera inmediata. En el caso del acceso a la web y del acceso remoto el requerimiento en cuanto al retardo es medio porque se desea que no tarde mucho tiempo ya que hay un usuario esperando por los paquetes. El audio bajo demanda así como el video bajo demanda no le dan mucha importancia al requerimiento de retardo ya que no les molesta que se demore un tiempo en comenzar la reproducción de los mismos, sin embargo notemos que son muy exigentes en cuanto a la fluctuación ya que una vez que comenzó la reproducción se desea que la misma se desarrolla sin interrupciones. Tanto en la telefonía como en las videoconferencias se es muy exigente con respecto al retardo así como en la fluctuación porque no es deseable tener que estar esperando unos segundos para escuchar o ver a una persona, y mucho menos que la comunicación se interrumpa cada cierto tiempo.

Con respecto al ancho de banda, notamos que las aplicaciones de voz no son muy exigentes al respecto, pero no ocurre lo mismo con la imagen ya que digitalizar las mismas genera muchos paquetes.

Clase 3 de Septiembre

**Técnicas para alcanzar buena QoS (quality of service)**

* Sobreaprovisionamiento
* Planificación
* Almacenamiento en Buffer (en el cliente)
* Modelado de tráfico (en el servidor)
* Reserva de recursos

**Sobreaprovisionamiento:**

Brindar una muy buena infraestructura de red, que sobren recursos. Estos son: capacidad de procesamiento del enrutador, memoria (espacio en buffer) y ancho de banda. El problema de esta solución es que es muy costosa. La red de telefonía fija tiene sobreaprovisionamiento, es muy raro levantar el teléfono y no tener tono.

**Planificación:**

El dispositivo que más responsabilidad tiene de brindar QoS dentro de una red es el router. Pero por defecto los routers no brindan Calidad de Servicio, hay que configurarla. Si se quiere QoS, desde el proceso que se está ejecutando en mí maquina hasta la máquina de la otra punta a la que quiero llegar, en el medio todos los eslabones deben implementar QoS. Si uno no lo implementa se pierde la QoS.

Dentro de la planificación tenemos varios métodos: FIFO, Prioridad, WFQ y Encolamiento Justo.  
FIFO (First in First out):

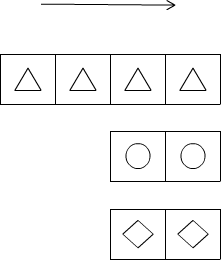
El router por defecto implementa FIFO. A medida que van llegando los paquetes los va colocando en una cola de entrada, los va procesando y los coloca en otra cola que es de salida respetando el orden en el que llegaron. El router sólo encamina con el IP de destino. Si se quiere QoS es necesario configurarla.

* Encolamiento Justo (Fair queueing) :

Supongamos que llegan al router varios paquetes seguidos del mismo flujo de datos y luego otros paquetes alternados de distintos flujos:



Si los procesa en ese orden los paquetes “círculos” y “rombos” van a demorar mucho en ser procesados, por lo tanto el router hace un análisis de flujo de datos y va a colocar los paquetes en diferentes colas según el flujo al que pertenecen. De esta manera se trata equitativamente todos los flujos que llegan. Se construyen nuevas colas por cada flujo detectado y se va atendiendo uno por uno, es decir, un triángulo, un circulo, un rombo, un triángulo, un circulo, un rombo y así…:



La desventaja es que trata a todos los paquetes por igual, y hay paquetes que necesitan mayor ancho de banda que otros. Por ejemplo la transmisión de voz necesita mucha mayor velocidad que un correo electrónico.

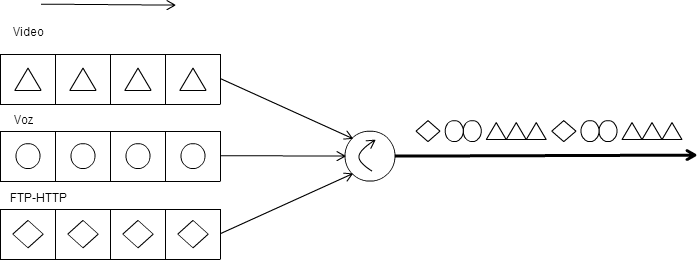
* Prioridad:

Se le da prioridad a los paquetes en el campo ToS (Type of Service) del paquete IP. Los primeros 3 bits de este campo indican prioridad, pudiendo diferenciarse 8 niveles, pero en la realidad solo se usan 6. El router analiza las prioridades y en función de esto construye distintas colas.

Siempre va a sacar por la línea de salida los paquetes de prioridad más alta hasta que se vacíe la cola, luego pasa a la siguiente cola de prioridad, cuando no quedan más paquetes pasa a la siguiente cola. La desventaja es que pueden retardarse demasiado los paquetes de prioridad más baja, si por ejemplo se está dando una video conferencia, los paquetes de transferencia de archivos, correos, páginas web van a demorar mucho en ser procesados.

* WFQ (weighted Fair Queueing = Encolamiento Justo Ponderado):

Mantiene diferentes colas de paquetes con distintos pesos en cada cola. Es un híbrido entre Fair Queueing y Prioridad. Los paquetes son clasificados por prioridad o por flujo, dependiendo el protocolo. IPv4 distingue por prioridad e IPv6 distingue por flujo. Se le asigna diferentes pesos a cada flujo y es el método que más se usa, ya que atiende a todos los flujos pero le dá más importancia a los que más lo necesitan. Por ejemplo:



**Almacenamiento en Buffer (en el cliente)**

Se utiliza para atenuar el jitter o fluctuación (variabilidad del retardo). Los flujos se almacenan en el buffer del receptor, se introduce retardo, para que el usuario no perciba las fluctuaciones. Se utiliza por ejemplo para los flujos de video o audio bajo demanda. El video se demora en arrancar porque está almacenando en buffer los paquetes que va recibiendo para luego no clavarse. Cuando tiene cierta cantidad de paquetes, los muestra.

**Modelado de Tráfico (en el Servidor)**

Controla la cantidad de paquetes que ingresan en la red para que no se congestione.

El objetivo es convertir un tráfico de ráfaga en un tráfico constante. En el servidor se controla que los paquetes se lancen de a 1 a la red para que no ingresen grandes cantidades de paquetes de una vez (ráfaga)

Existen dos formas diferentes:

* Algoritmo de cubeta con goteo:

El balde o cubeta representa un buffer en el servidor. Este se va llenando de los paquetes que se quieren introducir en la red, pero solo van saliendo gota por gota, es decir paquete por paquete. Se puede establecer por ejemplo, que se lance a la red un paquete por ciclo de reloj. También se puede controlar por cantidad de bytes, no sólo paquetes, pero siempre una cantidad constante. Esta última es más justa, ya que el tamaño de los paquetes no siempre es el mismo.

Si el balde o buffer se llena, los paquetes que llegan se descartan. No hay lugar físico para almacenarlos.

* Algoritmo de cubeta con ficha (token):

En cada pulso de reloj se genera un token o ficha, que es el derecho a transmitir un paquete (o cierta cantidad de bytes). El balde representa el buffer y éste se va cargando de fichas acumuladas. Si le llega una ráfaga de paquetes, tiene derecho a transmitirlos todos juntos según la cantidad de tokens que haya acumulado.   
Por ejemplo, mientras no hay datos para transmitir el servidor va acumulando fichas, supongamos que acumulo 6 tokens. De repente le llegan 8 paquetes para ser transmitidos. El servidor va a poder introducir a la red 6 paquetes juntos, y luego en los dos pulsos de reloj siguientes va a introducir los paquetes que le faltan. El número de fichas a acumular tiene un limite.

**Reserva de recursos**

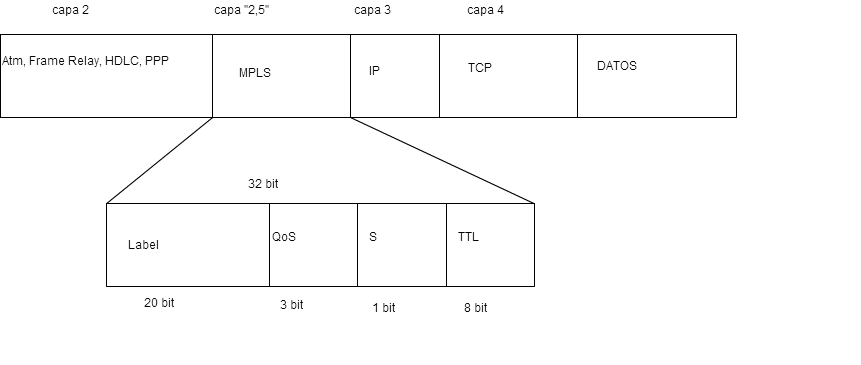
Solo se implementa si se detecta congestión. Si no hay congestión no conviene reservar recursos ya que se estaría subutilizando la red. Los recursos que se reservan son 3: Memoria, Ciclos de Micro y Ancho de Banda (de los routers). Los recursos se reservan para asegurar que se pueda establecer una conexión. El recurso más fácil de reservar es el ancho de banda.

Se usa un acuerdo de nivel de servicio LSA (level service agreement) entre el host que quiere establecer una conexión y el proveedor de servicio. Aquí el host debe declarar la cantidad de información que desea transmitir y en función de la congestión que tenga, el proveedor de servicio le dará una reserva de ancho de banda que asegure que pueda transmitir. Este LSA va a viajar por todos los router para ver que todos puedan asegurar ese ancho de banda. Si alguno está más congestionado, bajará el nivel del acuerdo.

**MPLS (Multiple protocol label switching = Conmutación de etiqueta multi-protocolo)**

Esta es otra forma de brindar QoS y es la que más se está utilizando. Es una forma de crear circuitos virtuales sobre una red de conmutación de paquetes, es decir, sobre IP.

A un paquete que está encapsulado, se le agrega una etiqueta entre la capa 2 y la capa 3 para que se encamine en función de la misma y no en función del IP. De esta manera la comunicación es más rápida ya que solo se da conmutación de capa 2 y no encaminamiento, que es mucho más lento, ya que hay que desencapsular la capa 2, leer la capa 3, ver la dirección de destino, consultar en la tabla, conmutar y además, mantener la tabla de encaminamiento, con actualizaciones que se intercambian entre los routers.



**QoS:** indica el nivel de prioridad del paquete.  
**S:** es el nivel jerarquico de etiqueta, si hay un label dentro de otro label se lo coloca en 1. Se utiliza para crear túneles.  
**TTL:** Time to Live.

Con esta técnica, los router se fijan el número de etiqueta, consultan en sus tablas que son mucho más pequeñas y conmutan los paquetes, cambiándole la etiqueta, ya que este número tiene significado local para ese router. El primer paquete del primer flujo va abriendo el camino y todos los paquetes siguientes del mismo flujo van a viajar por el mismo camino físico. Al conocer el camino se puede reservar recursos para ese flujo.