Capa de Red

La capa de red es la capa más baja que maneja la transmisión de extremo a extremo. Para lograr sus objetivos, la capa de red debe conocer la topología de la red (es decir, el conjunto de todos los enrutadores y enlaces) y elegir las rutas apropiadas incluso para redes más grandes. También debe tener cuidado al escoger las rutas para no sobrecargar algunas de las líneas de comunicación y los enrutadores, y dejar inactivos a otros. Por último, cuando el origen y el destino están en redes diferentes, ocurren nuevos problemas. La capa de red es la encargada de solucionarlos.

ASPECTOS DE DISEÑO DE LA CAPA DE RED

el funcionamiento de la **conmutación de almacenamiento y reenvío** en una red de comunicaciones. En este contexto, se presentan los componentes principales de la red, como los enrutadores del Proveedor de Servicio de Internet (ISP) y los equipos de los clientes. Un ejemplo de estos equipos es el host H1, que está conectado directamente a un enrutador del ISP, mientras que H2 está en una LAN conectada a su propio enrutador que se comunica con el ISP.

El proceso comienza cuando un host desea enviar un paquete. Este se transmite al enrutador más cercano, ya sea dentro de la LAN o a través de un enlace con el ISP. El paquete se almacena en el enrutador hasta que se procesa completamente, verificando la suma de verificación. Luego, se reenvía al siguiente enrutador en la ruta hasta llegar al destino final, donde se entrega.

Este proceso de transmitir, almacenar y reenviar paquetes en la red se denomina **conmutación de almacenamiento y reenvío** , que es un enfoque fundamental para el envío de datos en redes de comunicaciones.

Servicios proporcionados a la capa de transporte

La capa de red proporciona servicios a la capa de transporte en la interfaz entre la capa de red y de transporte. Hay que diseñar los servicios de manera cuidadosa, con los siguientes objetivos en mente:

1. Los servicios deben ser independientes de la tecnología del enrutador.
2. La capa de transporte debe estar aislada de la cantidad, tipo y topología de los enrutadores presentes.
3. Las direcciones de red disponibles para la capa de transporte deben usar un plan de numeración uniforme, incluso a través de redes LAN y WAN.

Para ellos se ofrece dos tipos de servicios que la capa de red deberia ofrecer:

1. **Servicios sin conexion**: este enfoque argumenta que los enrutadores deben encargarse solo de mover paquetes sin preocuparse por la confiabilidad. Los paquetes se envían sin conexión y no se realiza un seguimiento del estado o del orden de los paquetes. La responsabilidad de controlar errores y flujo recae en los **hosts** .
2. **Servicio orientado a conexion**: Este enfoque argumenta que la red debería garantizar un servicio confiable, similar al sistema telefónico tradicional, para asegurar calidad en el servicio, especialmente en aplicaciones de tiempo real como voz y video. Este modelo es más cercano a la idea de un servicio orientado a conexión, donde la red establece una conexión antes de que los paquetes se envíen y se maneja el control de errores y flujo de manera centralizada.

Implementacion del servicio sin conexion

cómo funciona una **red de datagramas** cuando la capa de red ofrece un **servicio sin conexión** . En este tipo de servicio, los paquetes de datos se envían de manera independiente, sin necesidad de una conexión previa entre los enrutadores. Cada paquete se maneja por separado y se enruta según la información contenida en su dirección, sin coordinación con los otros paquetes del mismo mensaje.

Funcionamiento de una red de datagramas:

1. **División de los mensajes** :

- Supongamos que un proceso (P1) en un host (H1) tiene un mensaje largo que desea enviar a otro proceso (P2) en un host (H2). La capa de transporte se encarga de dividir este mensaje en varios paquetes, ya que el mensaje es más grande que el tamaño máximo permitido por el protocolo de la capa de red.

1. **Envío de los paquetes** :

- Los paquetes se entregan a la capa de red, que los divide y los envía uno por uno a través de la red. Cada paquete se transmite por el enrutador más cercano según las tablas de enrutamiento de los enrutadores.

1. **Tabla de enrutamiento** :

- Los enrutadores utilizan tablas internas que les indican a qué salida (línea de transmisión) enviar los paquetes, basándose en las direcciones de destino. Cada entrada en la tabla es un par de destino y línea de salida. El enrutador toma la decisión de dónde enviar cada paquete de acuerdo con estas tablas.

1. **Almacenamiento y reenvío** :

- Los enrutadores almacenan temporalmente los paquetes después de recibirlos, verifican su integridad (mediante sumas de verificación) y luego los reenvían al siguiente enrutador o destino. Los paquetes pueden tomar diferentes rutas en la red, como se ve en el ejemplo de la figura 5-2, donde el paquete 4 es enviado por una ruta diferente a los otros paquetes debido a un cambio en la tabla de enrutamiento.

1. **Algoritmos de enrutamiento** :

- Los enrutadores utilizan **algoritmos de enrutamiento** para decidir la mejor ruta para los paquetes. Estos algoritmos pueden ajustar las rutas en tiempo real según el tráfico o la congestión de la red.

Ejemplo de servicio sin conexión:

- El **Protocolo IP** (Internet Protocol) es el ejemplo principal de un servicio sin conexión. En IP, cada paquete lleva una dirección IP de destino que los enrutadores utilizan para reenviar el paquete hacia su destino. IP no establece una conexión antes de enviar los paquetes, y cada paquete se enruta de forma independiente. En **IPv4** , las direcciones son de 32 bits, y en **IPv6** , son de 128 bits.

Implementacion del servicio orientado a conexion

El servicio **orientado a conexión** en redes de comunicación, se basa en la creación de **circuitos virtuales** . La idea detrás de los circuitos virtuales es evitar la necesidad de elegir una nueva ruta para cada paquete enviado. En cambio, cuando se establece una conexión, se elige una ruta de la máquina de origen a la máquina de destino como parte de la configuración de conexión y se almacena en tablas dentro de los enrutadores. Esa ruta se utiliza para todo el tráfico que fluye a través de la conexión, de la misma forma en que funciona el sistema telefónico. Cuando se libera la conexión, también se termina el circuito virtual. Con el servicio orientado a conexión, cada paquete lleva un identificador que indica a cuál circuito virtual pertenece. Si un tercer host quisiera establecer conexion con alguno de los involucrados, tambien puede usar el identificador de conexion 1, inicialmente, pero es tarea de los enrutadores de asignar un unevo identificador a la conexion nueva para eevitar confusion.

Algoritmos de enrutamiento

El algoritmo de enrutamiento es aquella parte del software de la capa de red responsable de decidir por cuál línea de salida se transmitirá un paquete entrante. Si la red usa datagramas de manera interna, esta decisión debe tomarse cada vez que llega un paquete de datos, dado que la mejor ruta podría haber cambiado desde la última vez. Si la red usa circuitos virtuales internamente, las decisiones de enrutamiento se toman sólo al establecer un circuito virtual nuevo. En lo sucesivo, los paquetes de datos simplemente siguen la ruta ya establecida. Este último caso a veces se llama enrutamiento de sesión, dado que una ruta permanece vigente durante toda una sesión (por ejemplo, durante una sesión a través de una VPN).

- Enrutamiento estatico (algortimos no adaptativos): no basan sus decisiones de enrutamiento en mediciones o estimaciones del tráfico y la topología actuales. En cambio, la decisión de qué ruta se usará para llegar de I a J (para todas las I y J) se calcula por adelantado, fuera de línea, y se descarga en los enrutadores al arrancar la red
- Enrutamiento dinamico (algoritmo adaptativos) : cambian sus decisiones de enrutamiento para reflejar los cambios de topología y algunas veces también los

cambios en el tráfico

Todos los algortimos usan arboles sumideros que no es mas que la red de rutas optimas de todos los orignes a un destino dado que forma un arbol con raiz en el destino

Algoritmo de la ruta mas corta

Aplica el algoritmo conocido de Dijkstra

Inundacion

La **inundación** es una técnica de enrutamiento en la que cada paquete recibido se envía a todos los enrutadores vecinos, excepto al que lo recibió. Esto asegura que el paquete llegue a todos los nodos de la red, pero puede generar muchos paquetes duplicados, lo que no es eficiente.

Para evitar la propagación infinita de paquetes, se utilizan dos mecanismos:

1. **Contador de saltos** : Cada paquete tiene un contador que disminuye con cada salto. Cuando llega a cero, el paquete se descarta.
2. **Registro de paquetes enviados** : Cada enrutador lleva un registro de los paquetes que ya ha enviado para no retransmitirlos. Utiliza un número de secuencia para identificar los paquetes y evitar duplicados.

Aunque la inundación no es eficiente para la mayoría de los casos, es muy robusta y garantiza la entrega de paquetes, incluso en redes con fallos. Es útil en redes donde se necesita difundir información a todos los nodos, como en redes inalámbricas.

Enrutamiento por vector de distancia

opera haciendo que cada enrutador man□tenga una tabla (es decir, un vector) que proporcione la mejor distancia conocida a cada destino y el enlace que se puede usar para llegar ahí. Para actualizar estas tablas se intercambia información con los vecinos. Eventualmente, todos los ruteadores conocen el mejor enlace para alcanzar cada destino. Éste fue el algoritmo original de enrutamiento de ARPANET y también se usó en Internet con el nombre de RIP (uso de Bellman Ford). Se puede utilizar metricas como el salto o el retardo. La principal desventaja es destinos inalcanzable

por salida del enlace. Los enrutadores vecinos no lo sabran y seguiran intentando usar rutas fallidas durante varios intercambios de informacion.

Enrutamiento por estado de enlace

Las variantes de este enrutamiento llamadas IS-IS y OSPF son los algoritmos de enrutamiento más utilizados dentro de las redes extensas y de Internet en la actualidad. La idea detrás del enrutamiento por estado del enlace es bastante simple y se puede enunciar en cinco partes. Cada enrutador debe realizar lo siguiente para hacerlo funcionar:

1. **Descubrir a sus vecinos y conocer sus direcciones de red** : Cada enrutador debe identificar sus vecinos, generalmente enviando paquetes de tipo HELLO a través de enlaces punto a punto y esperando respuestas.
2. **Establecer la métrica de distancia o de costo para cada uno de sus vecinos**: Cada enrutador asigna una métrica a cada uno de sus enlaces (por ejemplo, costo basado en el ancho de banda o el retardo de los enlaces).
3. **Construir un paquete que indique todo lo que acaba de aprender**: Una vez que se recaba la información necesaria, el enrutador crea paquetes con los detalles sobre los enlaces descubiertos y sus costos. Estos paquetes se envían a otros enrutadores.
4. **Enviar este paquete a todos los demás enrutadores y recibir paquetes de ellos** : Los paquetes de estado del enlace se distribuyen a todos los enrutadores mediante un proceso de inundación. Los enrutadores mantienen un registro de los paquetes para evitar duplicados y asegurar que la topología esté actualizada.
5. **Calcular la ruta más corta a todos los demás enrutadores.** : Cada enrutador usa el algoritmo de Dijkstra para calcular la ruta más corta hacia otros enrutadores y actualiza sus tablas de enrutamiento.

De hecho, se distribuye la topología completa a todos los enrutadores. Después se puede ejecutar el algoritmo de Dijkstra en cada enrutador para encontrar la ruta más corta a los demás enrutadores.

Enrutamiento jerarquico

El **enrutamiento jerárquico** se utiliza para gestionar redes de gran tamaño, donde las tablas de enrutamiento tradicionales pueden volverse demasiado grandes y difíciles de

manejar. A medida que las redes crecen, los enrutadores deben gestionar un mayor número de rutas, lo que puede consumir mucha memoria, tiempo de procesamiento y ancho de banda. El enrutamiento jerárquico organiza la red en **regiones** o **niveles** para reducir el tamaño de las tablas de enrutamiento, simplificando la administración de rutas.

Concepto básico:

- **Regiones** : La red se divide en áreas o regiones, cada una con su propia topología interna. Los enrutadores dentro de una región conocen bien la estructura local pero no la de otras regiones.
- **Rutas jerárquicas** : En lugar de que cada enrutador conozca la topología de toda la red, solo conoce la topología dentro de su propia región y cómo llegar a otras regiones. El tráfico entre regiones se enruta a través de un enrutador designado para esa región, reduciendo la complejidad de las tablas de enrutamiento.

Ventajas:

- **Reducción del tamaño de las tablas de enrutamiento** : El ejemplo en el texto muestra que, al dividir la red en varias regiones, la tabla de enrutamiento de un enrutador se reduce significativamente.
- **Escalabilidad** : Al organizar la red en jerarquías, se pueden manejar redes de mucho mayor tamaño sin sobrecargar a los enrutadores.

Desventajas:

- **Aumento de la longitud de la ruta** : Aunque se reducen las tablas de enrutamiento, las rutas pueden volverse más largas. Por ejemplo, un paquete puede necesitar pasar por más enrutadores de diferentes regiones para llegar a su destino, lo que incrementa el tiempo de transmisión.

Ejemplo de jerarquía multinivel:

- En una red grande como la de un país o un continente, se pueden utilizar varios niveles de jerarquía. Por ejemplo, un enrutador local en Berkeley podría enviar paquetes a Los Ángeles, este a Nueva York, y luego a Nairobi antes de llegar a su destino en Kenia.

Número de niveles óptimos:

- En una red con **N enrutadores** , el número óptimo de niveles en la jerarquía es **ln N** (logaritmo natural de N). Esto significa que la jerarquía no debe ser demasiado profunda para evitar largas rutas, pero debe ser suficiente para reducir el tamaño de las tablas de enrutamiento. En el ejemplo, con 720 enrutadores, se podría dividir la red en 24 regiones, cada una con 30 enrutadores, o usar una jerarquía de tres niveles con 8 clústeres de 9 regiones, lo que reduce significativamente el número de entradas en las tablas.

Enrutamiento por difusion (broadcasting)

Es un proceso en el que un mensaje se envia simultaneamente a varios o todos los hosts en uan red.

Los principales métodos son:

1. **Envío individual:** El origen envía un paquete a cada destino.
    - **Problema:** Ineficiente, lento y requiere conocer todos los destinos.
2. **Enrutamiento multidestino:** Un paquete incluye una lista de destinos.
    - **Ventaja:** Usa mejor el ancho de banda.
    - **Problema:** Complejo y el origen aún debe conocer todos los destinos.
3. **Inundación:** Reenvía paquetes por todas las líneas excepto la de entrada.
    - **Ventaja:** Simple.
    - **Problema:** Genera duplicados excesivos.
4. **Reenvío por ruta invertida (RPF):** Solo reenvía paquetes que llegan por la mejor ruta al origen; descarta duplicados.
    - **Ventaja:** Más eficiente que la inundación.
    - **Problema:** No minimiza completamente los paquetes.
5. **Árbol de expansión:** Reenvía paquetes solo por un árbol que conecta todos los enrutadores sin ciclos.
    - **Ventaja:** Usa el mínimo número de paquetes.
    - **Problema:** Los enrutadores deben conocer la estructura del árbol, lo que no siempre es posible.

- **RPF:** Genera 24 paquetes con algunos duplicados.
- **Árbol de expansión:** Usa solo 14 paquetes, minimizando el tráfico.

Enrutamiento multidifusion(multicasting)

es un método para enviar mensajes a grupos específicos en una red, optimizando el uso del ancho de banda.

Principales conceptos:

1. **Multidifusión frente a difusión:**
    - La difusión envía un mensaje a toda la red, lo que puede ser ineficiente si muchos nodos no necesitan el mensaje.
    - La multidifusión optimiza esto al enviar mensajes solo a nodos que pertenecen a un grupo definido.
2. **Árboles de expansión recortados:**
    - Los paquetes se envían a través de árboles que conectan únicamente a los miembros del grupo.
    - Ejemplo: Para dos grupos en una red, se crean árboles distintos que eliminan enlaces innecesarios, reduciendo el tráfico y aumentando la eficiencia.
3. **Protocolos para multidifusión:**
    - **MOSPF (Multicast OSPF):** Cada enrutador construye su propio árbol de expansión recortado, eliminando enlaces no relevantes.
    - **DVMRP (Distance Vector Multicast Routing Protocol):** Utiliza mensajes **PRUNE** para recortar el árbol de expansión, eliminando ramas no necesarias.
4. **Árboles basados en núcleo:**
    - En lugar de múltiples árboles por emisor, se crea un solo árbol por grupo con un punto central (núcleo).
    - Los emisores envían paquetes al núcleo, que luego los distribuye por el árbol.
    - Ventaja: Reduce el almacenamiento y el cálculo en los enrutadores.
    - Desventaja: Puede ser menos eficiente para ciertos emisores, dependiendo de la ubicación del núcleo.
5. **Uso práctico:**
    - Los árboles compartidos, como los basados en núcleo, son comunes para grupos dispersos en Internet. Ejemplo: PIM (Protocol Independent Multicast).

Enrutamiento anycast

En anycasy , un paquete se entrega al miembro mas cercano de un grupo. En ocasiones los nodos proporcionan un servicio, como la hora del dia o la distribucion

del contenido , en donde todo lo que importa es obtner la informacion correcta sin importar el nodo con el que haya hecho contacto.

Enrutamiento para hosts moviles

aborda el desafío de localizar y enrutar paquetes a dispositivos que cambian de ubicación mientras permanecen conectados a la red. Este problema surge en escenarios como usuarios con laptops o dispositivos móviles que desean mantener conectividad constante, sin importar su ubicación. Para ello utiliza su direccion base permanente para determinar su ubicacion base. Si el host se mueve, obtiene una nueva direccion local llamada direccion de custodia.

El host movil informa a un agente base sobre su nueva direccion. Los paqutes enviados al host movil van primero a su agente de base, este lo encapsula los paquetes (tunelizacion) y los reenvia a la direccion de custodia del host . Una vez que el emosir conoce la nueva direccion del host movel, puede enviar paquetes directamente evitando al agente de base .

Enrutamiento en redes ad hoc

En el caso anterior, los host son moviles pero los enrutadores son fijos. Un caso extremo es cuando los enrutadores mismo son moviles. (bomberos en areas de peligros, vehiculos militares, flota de barco , reunion de personas con computadoras portatiles en un area que no cuenta con WiFi)

Características principales :

1. **Dinámica de la red** :

- Los nodos se mueven constantemente, lo que cambia la topología sin previo aviso.
- Esto hace que el enrutamiento sea mucho más complicado que en redes cableadas.

1. **AODV (Ad hoc On-demand Distance Vector)** :

- Algoritmo popular que encuentra rutas "bajo demanda", es decir, solo cuando se necesitan.

- Esto ahorra recursos, ya que evita mantener rutas que podrían volverse obsoletas rápidamente.

Cómo funciona AODV :

1. **Descubrimiento de rutas** :

- Si un nodo quiere enviar un paquete y no tiene una ruta al destino, genera una **solicitud de ruta (ROUTE REQUEST)** .
- La solicitud se difunde a través de la red mediante inundación controlada, alcanzando los nodos vecinos.
- Cuando la solicitud llega al destino, este responde con un **paquete de respuesta (ROUTE REPLY)** , que vuelve al origen usando la ruta inversa.
- Cada nodo en el camino almacena información para reenviar paquetes futuros.

1. **Mantenimiento de rutas** :

- Los nodos periódicamente envían mensajes de "hello" para verificar que los enlaces a sus vecinos siguen activos.
- Si un enlace falla, los nodos eliminan las rutas afectadas y notifican a otros nodos para que también actualicen sus tablas.
- Nuevas rutas se descubren cuando sea necesario.

1. **Optimización** :

- El algoritmo usa números de secuencia para evitar confusiones entre rutas antiguas y nuevas.
- Se limita la difusión para reducir la sobrecarga, comenzando con un área pequeña y ampliándola si es necesario.

Otros enfoques :

- **DSR (Dynamic Source Routing)** : Similar a AODV pero almacena la ruta completa en los paquetes.
- **GPSR (Greedy Perimeter Stateless Routing)** : Usa la ubicación geográfica de los nodos para calcular rutas sin necesidad de tablas de enrutamiento.

Algoritmos de Control de Congestion

Congestion: Situacion en la cual hay demasiado trafico de paquetes en una red, lo que provoca que el bufer de los enrutadores dentro de los enrutadores se llenen y se pierden algunos paquetes.

Cuando en una red hay **congestión** , significa que hay más datos (tráfico) tratando de pasar por los recursos disponibles (como cables o enrutadores). Esto provoca retrasos, pérdida de paquetes y una red lenta. Hay dos maneras principales de manejar este problema:

1. **Aumentar los recursos** : Esto significa dar más espacio para el tráfico. Algunos ejemplos son:

- **Mejorar la red** : Usar cables más rápidos o enrutadores más potentes.
- **Abrir rutas alternativas** : Como activar caminos de respaldo cuando hay demasiados datos en uno principal, igual que abrir más carriles en una autopista llena.
- **Ajustar rutas según el tráfico** : Si una ruta está muy ocupada, desviar los datos por caminos menos congestionados.

1. **Reducir el tráfico** : Esto significa limitar la cantidad de datos que se envían. Algunas formas de hacerlo son:

- **Negar nuevas conexiones** : Si una red está llena, simplemente no aceptar más conexiones para evitar que todo se detenga.
- **Enviar advertencias a los usuarios** : Si una parte de la red está congestionada, se puede pedir a las computadoras o dispositivos que envían muchos datos que bajen su velocidad.
- **Eliminar paquetes no importantes** : Si no hay otra opción, la red puede descartar los datos menos prioritarios para liberar espacio.

Enrutamiento consciente del trafico

es un enfoque que considera la carga de los enlaces de la red para calcular las mejores rutas para enviar datos. A diferencia de los métodos de enrutamiento tradicionales que usan **ponderaciones fijas** basadas en características como el ancho de banda o el retardo de propagación, este enfoque incluye variables dinámicas como la **carga actual** o el **retardo de encolamiento** .

Cuando las ponderaciones cambian en función de la carga, pueden generarse oscilaciones:

1. Si un enlace está congestionado, el sistema redirige el tráfico a otra ruta menos ocupada.
2. Esto puede congestionar la nueva ruta, y el sistema vuelve a cambiar las rutas al enlace original.
3. Este ciclo puede repetirse, causando **enrutamiento errático** e inestabilidad en la red.

Control de admision

Es una tecnica utilizada en las redes de circuitos virtuales para evitar la congestion. La idea es no establecer un nuevo circuito virtual a menos que la red pueda transportar el trafico adicional sin congestionarse. Si la red está cerca de la saturación, se rechaza la solicitud para evitar que la situación empeore, similar a cuando un sistema telefónico no permite nuevas llamadas si está sobrecargado.

Regulacion de trafico

Es el proceso de controlar el flujo de datos en una red para evitar que se congestione. La idea es asegurarse de que los emisores no envíen más datos de los que la red puede manejar sin que se atasque.

1. Detección de congestión

- Los enrutadores de la red están vigilando constantemente el uso de los recursos de la red, como los **bufers** (donde se guardan los paquetes) y los **tiempos de retraso** . Si los enrutadores detectan que los paquetes se están acumulando o que los tiempos de espera aumentan, significa que la red está a punto de congestionarse.

2. Notificación de congestión

- Cuando un enrutador detecta que hay congestión, tiene que avisar a los **emisores** (los dispositivos que envían datos). Esto se hace de varias maneras:
- **Paquetes reguladores** : El enrutador envía un paquete especial de vuelta al emisor, diciéndole que **reduzca la velocidad** con la que está enviando datos (por

ejemplo, un 50% menos).

- **Notificación Explícita de Congestión (ECN)** : En lugar de enviar paquetes adicionales, el enrutador simplemente marca los paquetes con una señal que indica que hubo congestión en el camino. El emisor verá esta señal y reducirá su velocidad cuando reciba la respuesta.
- **Contrapresión de salto por salto** : En lugar de esperar que el emisor reduzca su velocidad, la red puede "empujar" hacia atrás la señal de congestión desde el lugar donde se detectó. Así, los enrutadores intermedios también comienzan a reducir el flujo de datos hacia la zona congestionada.

3. ¿Por qué es importante?

- La regulación de tráfico es crucial para **evitar que la red se colapse** . Si no se controla, los paquetes pueden empezar a **perderse** y los tiempos de espera se incrementan, lo que afecta el rendimiento de la red.

Desprendimiento de carga

es una estrategia utilizada por los enrutadores para manejar la congestión cuando ya no pueden procesar más paquetes. Es como una medida extrema para evitar que la red se colapse, similar a cómo se apagan algunas áreas de una red eléctrica cuando hay demasiada demanda. Se usa principlametn cuando no se resuelve la congestion con los metodos anteriores.

¿Cómo decidir qué paquetes eliminar?

- **Para transferencias de archivos** : es mejor eliminar los paquetes antiguos, ya que los nuevos aún no se pueden usar.
- **Para streaming de video o audio (tiempo real)** : es mejor eliminar los paquetes nuevos, ya que los antiguos ya no sirven.

Existen dos enfoques para esto:

1. **Vino** : se elimina primero el paquete más antiguo.
2. **Leche** : se elimina primero el paquete más nuevo.

Desprendimiento inteligente de carga

Los enrutadores pueden **marcar la importancia de los paquetes** , de modo que si deben eliminar algunos, descarten primero los de menor importancia.

Detección temprana aleatoria (RED)

En lugar de esperar a que la red se sature, los enrutadores pueden empezar a **descartar paquetes al azar** cuando la cola de paquetes empieza a ser muy larga. Esto da tiempo a los emisores para reducir su velocidad antes de que el problema empeore.

ECN vs RED

ECN (Notificación Explícita de Congestión) es mejor que RED, ya que le avisa al emisor directamente sobre la congestión, lo que permite reaccionar más rápido.

En resumen, el desprendimiento de carga ayuda a evitar que la red se colapse, y puede hacerse de manera más inteligente al eliminar primero los paquetes menos importantes.

Capa de red de Internet

- **Internet como una red de redes** : Internet está compuesta por múltiples redes interconectadas, llamadas **Sistemas Autónomos (AS)** .
- **Redes troncales** : Son las redes de alto ancho de banda que conectan diversas partes de Internet. Las más grandes son llamadas **redes de Nivel 1** .
- **Proveedores de Servicios de Internet (ISP)** : Conectan redes regionales y proporcionan acceso a Internet para hogares, negocios y centros de datos.
- **Capa de red de Internet** : El protocolo **IP (Protocolo de Internet)** es el principal responsable de interconectar estas redes y transportar paquetes entre la fuente y el destino sin garantizar la entrega.

Funcionamiento de la comunicación en Internet:

1. **Capa de transporte** : Divide los datos en paquetes.
2. **Enrutadores IP** : Reenvían los paquetes a través de las redes hasta el destino.
3. **Capa de red en el destino** : Ensambla los paquetes para devolver los datos completos al proceso receptor.

Redundancia y rutas en Internet:

- Existen **múltiples rutas** entre dos puntos debido a la conectividad redundante en las redes troncales e ISP.
- Los **protocolos de enrutamiento** se encargan de decidir qué rutas tomar para enviar los paquetes de forma eficiente.

En resumen, la capa de red de Internet fue diseñada siguiendo principios claros para garantizar su funcionamiento eficiente, sencillo y escalable. La interconexión de redes, utilizando IP, permite la transferencia de paquetes a través de diversas rutas, garantizando la comunicación entre miles de millones de dispositivos.

EL protocolo IP version 4 (IPv4)

El encabezado de un datagrama IPv4 consta de dos partes : una fija de 20 bytes y una opcional cuya longitud varia. Los bits se transmiten de izquierda a derecha y de arriba hacia abajo, en un orden de red big-endian , lo que requiere una conversion en maquinas little-endian.

Campos:

- Version(4 bits): INdica la version del protocolo, con la version 4 como la mas utilizada. IPv6 es la siguiente
- IHL (Internet Header Lenght - 4 bits) : Especfica la longitud del encabezado en palabras de 32 bits. El valor minimo es 5 y el maximo es 15 lo que limita el encabezado a 60 bytes.
- Servicios diferenciados (6bits): Originalmente destinado a priorizar diferentes clases de servicio (por ejemplo, voz o transferencia de archivos), este campo ha cambiado de nombre y ahora se utiliza para marcar el paquete con una clase de servicio y gestionar la congestión.
- Longitud Total (16 bits): Indica el tamaño total del datagrama (encabezado + datos), con un límite de 65,535 bytes
- Identificacion (16 bits): Usado para identificar fragmentos de un datagrama y asegurar que pertenezcan al mismo paquete.

- **Suma de verificación del encabezado (16 bits):** Se utiliza para detectar errores en el encabezado mientras el paquete viaja por la red. El algoritmo suma todas las medias palabras de 16 bits del encabezado a medida que vayan llegando, mediante el uso de la aritmética de complemento a uno, y después obtiene el

complemento a uno del resultado. Para los fines de este algoritmo, se supone que la Suma de verificación del encabezado es cero al momento de la llegada. Dicha suma de verificación es útil para detectar errores mientras el paquete viaja por la red. Tenga en cuenta que se debe recalcular en cada salto, ya que por lo menos hay un campo que siempre cambia (el campo Tiempo de vida), aunque se pueden usar trucos para agilizar ese cálculo

- **Desplazamiento del Fragmento (13 bits):** Indica la posición del fragmento dentro del paquete original, permitiendo la reconstrucción del datagrama.
- **Tiempo de Vida (TTL - 8 bits):** Contador que limita la vida útil del paquete en la red. En cada salto, este valor disminuye hasta que llega a cero, momento en el cual el paquete se descarta.
- **Protocolo (8 bits):** Especifica el protocolo de capa superior, como TCP o UDP.
- **Dirección de Origen (32 bits):** Dirección IP del origen del paquete.
- **Dirección de Destino (32 bits):** Dirección IP del destino del paquete.
- **Opciones (Variable):** Permite incluir características adicionales, como seguridad, enrutamiento estricto o libre, y registro de ruta.

Direcciones IP

Una característica que define a IPv4 consiste en sus direcciones de 32 bits. Cada host y enrutador de Internet tiene una dirección IP que se puede usar en los campos Dirección de origen y Dirección de destino de los paquetes IP. Es importante tener en cuenta que una dirección IP en realidad no se refiere a un host, sino a una interfaz de red, por lo que si un host está en dos redes, debe tener dos direcciones IP. Sin embargo, en la práctica la mayoría de los hosts están en una red y, por ende, tienen una dirección IP. En contraste, los enrutadores tienen varias interfaces y, por lo tanto, múltiples direcciones IP.

Prefijos

Las direcciones `IPv4` tienen una mascara que consiste de un numero de 32 bits con un prefijo de d bits todos puestos en 1 y los demas en 0. El prefijo representa la direccion de la red mientras que el resto representa la direccion de un host en particular.

Las direcciones IP son jerarquicas , esto significa que la direccion IP se divide en una parcion de red (que es comun para todos los hosts en una misma red) y una porcion

de host (que varia entre los diferentes dispositivos en al red). Un prefijo de red es un bloque de direcciones IP continuas . Se representa como `DireccionIP/longitudDeRed` por ejemplo `128.208.0.0/24` donde el `/24` indica que los primeros 24 bits corresponden a la red y el resto son para los hosts.

Los **enrutadores** utilizan este prefijo para hacer el reenvío de los paquetes, basándose únicamente en la porción de red de la dirección.

La `mascara de subred` se utiliz para separar la porcion de red y la porcion de host de una direccion IP. Se representa en forma binaria con una serie de 1s (para la red) y 0s (para los hosts)

OK, funcionamiento:

- Cada enrutador (y host)) tiene una tabla de ruta, con campos como : Direccion de Red, Mascara de Red, Interfaz , Metrica ,....
- Cuando llega un datagrama, se toma la direccion de destino y por cada mascara de red se hace la operacion AND , si coincide el resultado con el correspondiente direccion en red, entonces se decide enviar el datagrama por esa red.
- Analogamente para un host, se ve si la direccion de desino pertence a la misma red, etonces no necesita ser enviada ela route, mediante las direcciones MAC

Subredes

El proceso de creacion de una subred es un tecnica fundamental para adminsitrar direccion IP dentro de una red, permitiendo dividir una red mas grande en varias mas pequennas. Por ejemplo una organizacion quiere dividir en partes mas pequennas para diferentes departamnetos o segmentos de su infrastructura.

Por ejemplo a un red que se le asigna `128.208.0.0/16` se divide en subredes para tres departamntos:

- Ciencias COmputacionales : `128.208.0.0/17` que cubre direcciones de `128.208.128.0` a `128.208.191.255`

Lo que se ahce para saber el rango es fijar todos los bits del host a `1` y a partir de ahi se calcual la longtidu del rango y se le suma a la direccion de la red.

Cuando un paquete llega a un enrutador, el enrutador debe determinar a qué subred pertenece la dirección IP de destino. Para hacerlo, el enrutador realiza una operación

de **AND** entre la dirección de destino del paquete y las máscaras de subred de las subredes configuradas. La idea es verificar qué subred tiene el prefijo más largo que coincide con la dirección de destino.

Al dividir la red en subredes, los enrutadores no tienen que mantener una tabla con todas las direcciones de cada host (lo que sería ineficiente). En cambio, pueden usar las **máscaras de subred** y realizar la operación AND para verificar qué subred corresponde a la dirección de destino. Esto permite que el proceso de enrutamiento sea más eficiente.

CDIR : Enrutamiento Interdominio sin Clases

El problema central que aborda CIDR es la **explosión de las tablas de enrutamiento** en Internet.

Problemas en el Enrutamiento

1. **Tamaño de las tablas de enrutamiento** :

- Los enrutadores en ISP grandes deben manejar tablas de enrutamiento con millones de entradas.
- Buscar en estas tablas a altas tasas de transmisión requiere hardware especializado y memoria rápida.

1. **Intercambio de información de enrutamiento** :

- Los enrutadores deben compartir información sobre las rutas.
- A mayor tamaño de las tablas, mayor complejidad en el intercambio y mayor probabilidad de errores.

1. **Ineficiencia en la asignación de direcciones IP** :

- Una jerarquía rígida (como en la red telefónica) haría ineficiente el uso de los 32 bits de las direcciones IP.

Solución: CIDR

1. **Agregación de rutas** :

- Combina múltiples prefijos pequeños en uno más grande.

- Ejemplo: Tres redes (Cambridge, Oxford y Edimburgo) se resumen en un único prefijo `/19`, reduciendo el número de entradas en las tablas.

1. **Flexibilidad en la asignación de direcciones** :

- Las direcciones no están restringidas por clases rígidas (A, B, C).
- Permite que diferentes prefijos representen bloques de distintos tamaños.

1. **Prefijo más largo coincidente** :

- Cuando un paquete coincide con múltiples prefijos, se elige el que tiene **la mayor cantidad de bits significativos**
- Ejemplo: Un paquete destinado a una dirección en San Francisco coincidirá primero con un prefijo más específico `/22` antes que con un prefijo más general `/19`.

Ejemplo de Asignación de Direcciones

Universidad	Dirección inicial	Dirección final	Tamaño del bloque	Prefijo
Cambridge	`194.24.0.0`	`194.24.7.255`	2048 direcciones	`/21`
Edimburgo	`194.24.8.0`	`194.24.11.255`	1024 direcciones	`/22`
Disponible	`194.24.12.0`	`194.24.15.255`	1024 direcciones	`/22`
Oxford	`194.24.16.0`	`194.24.31.255`	4096 direcciones	`/20`

- Los enrutadores locales mantienen entradas para cada prefijo.
- En un enrutador distante (ej., Nueva York), los prefijos se combinan en un solo bloque `/19` para reducir las tablas de enrutamiento.

Ventajas de CIDR

1. **Reducción de tablas de enrutamiento** :

- La agregación disminuye las entradas necesarias, lo que reduce costos y complejidad.

1. **Mayor eficiencia en el uso de direcciones IP** :

- Se asignan bloques con tamaños adecuados, evitando desperdicio.

1. **Escalabilidad** :

- Permite que Internet maneje un mayor número de redes sin problemas significativos en el enrutamiento.

Direccionamiento con clases y especiales

las clases de direcciones son :

- A : bit inicial 0 y prefijo /8
- B : bits iniciales 10 y prefijo /16
- C: bits inciales 110 y prefijo /24
- D : bits iniciales 1110 y no tien prefijo (se usa para enviar paquetes a mutiples host)
- E : bits inciiales 1111 y refijo reservado : para uso futuro

Los problemas son: las redes de clase A son demasidado grande para la mayoria de organiczaiones

Las redes de clase B incluso tambien ,pero son las que mas de adecuan . Las de tipo C son ya muy pequennas , con apenas 256 direcciones. Por tanto se desperdicia de direcciones.

Entonces se introdujo los conceptos de subrdes y CDIR , permitiendo usar prefijos arbitrarios y haciendo mas eficientes las tablas de enrutamiento globales.

`0.0.0.0` representa 'esta red' utilizado spo un host durante el arranque

`255,255.255.255` direccion de difusion para todos los hosts en la reed local

`127.x.y.z` reservada para pruebas de loopback, paquetes no salen del hsot

Direcciones con un solo bit 1 en el campo del host: Usada apra difusiones dirigidas a redes especficas.

NAT (network address translation)

tecnica descrita en el RFC 3022 que traduce direcciones IP internas privadas en una direccion IP publica para comunicarse con Internet. Es comun en redes domesticas y pequennas empresas, donde un enrutador o caja NAT actua como intermedio.

Funcionamiento de NAT

1. **Dentro de la red interna:**
    - Cada dispositivo tiene una dirección IP privada (por ejemplo, **10.0.0.1**).
    - Los paquetes generados llevan esta dirección como origen.
2. **En el dispositivo NAT:**
    - La dirección IP de origen se reemplaza por la dirección IP pública del cliente (por ejemplo, **198.60.42.12**).
    - Se utiliza el campo **puerto de origen** del encabezado TCP/UDP para identificar la conexión original, evitando conflictos si varios dispositivos internos usan el mismo puerto.
3. **Cuando llega una respuesta de Internet:**
    - La caja NAT consulta su tabla de traducción para identificar el dispositivo interno correspondiente y reescribe la dirección IP de destino al reenviar el paquete.

Ventajas de NAT

- **Ahorro de direcciones IP públicas:** Un ISP puede asignar una sola dirección IP pública a una red con múltiples dispositivos internos.
- **Privacidad y seguridad básica:** Bloquea paquetes entrantes no solicitados por defecto.
- **Compatibilidad con redes existentes:** Funciona sin modificar significativamente la infraestructura de red.

Problemas y Limitaciones de NAT

1. **Viola el modelo IP original:**
    - Cada IP debería ser única globalmente, pero NAT permite que múltiples dispositivos compartan una dirección pública.
2. **Rompe la conectividad de extremo a extremo:**
    - Un host externo no puede iniciar una conexión con un dispositivo interno sin configuraciones adicionales (como **port forwarding** o técnicas de NAT transversal), por ejemplo dos hosts interno en redes privadas que usan NAT (donde se tiene configurado por el enrutador el reenvio de puertos)
3. **Dependencia del estado de conexión:**
    - Si la caja NAT falla, se pierden las conexiones activas.

4. **Problemas de compatibilidad con capas superiores:**
    - NAT depende de los encabezados TCP/UDP. Si estos cambian (por ejemplo, puertos más grandes), NAT deja de funcionar.
5. **Limitación de puertos:**
    - Cada dirección pública puede gestionar un máximo de 65,536 conexiones simultáneas (menos en la práctica debido a los puertos reservados).
6. **Problemas con aplicaciones específicas:**
    - Protocolos como FTP o multimedia (H.323) pueden fallar, ya que incluyen direcciones IP en sus datos y NAT no los entiende de forma nativa.

IPv6

IPv6 tiene como objetivos:1. Soportar miles de millones de dispositivos.

1. Reducir el tamaño de las tablas de enrutamiento.
2. Hacer el procesamiento de paquetes más rápido.
3. Mejorar la seguridad (autenticación y privacidad).
4. Mejorar la calidad del servicio, especialmente para multimedia.
5. Soportar la **multidifusión** .
6. Permitir la **movilidad de los dispositivos** sin necesidad de cambiar su dirección.
7. Evolucionar con el tiempo y coexistir con IPv4 durante una transición gradual.

- IPv6 tiene direcciones de 128 bits en vez de 32 bits .
- Tiene solo 7 campos en el encabezado comparado con los 13 campos en IPv6
- Mejora en seguridad , autenticacion y privacidad son integradas en IPv6.
- Mejora en calidad de servicios

Encabezados de IPv6

- Version : Campo siempre tiene el valor de 6 para IPv6.
- Servicios diferenciados: Similar al cmapo de IPv4 se usa para al calidad de servicios y manejar paquetes con requisitos especiales como los de baja laencia . Los dos bits inferiores se usan para indicar congestionamiento
- Etiqueta de flujo: Permite agrupas paquetes con los mismos requisitos de tratamiento.
- **Longitud de carga útil** : Este campo indica la longitud del contenido del paquete, excluyendo los 40 bytes del encabezado

-

- **Siguiente encabezado** : Señala el próximo encabezado de extensión (si lo hay), o el protocolo de transporte como TCP o UDP. Esto ayuda a simplificar el encabezado al permitir encabezados de extensión opcionales.
- **Límite de saltos** : Similar al campo TTL (Time to Live) en IPv4, se utiliza para evitar que los paquetes queden atrapados en un bucle infinito. El valor se decrementa en cada salto de red.
- **Dirección de origen y destino** : Las direcciones son de 128 bits (16 bytes), lo que proporciona una cantidad prácticamente ilimitada de direcciones. Las direcciones IPv6 se representan en formato hexadecimal y pueden ser abreviadas utilizando ciertas reglas.

IPv6 no incluye una suma de verificacion en el encabezado ya que se considera inneceario debidao a las sumas de verificacion ya implementados en las capas de enlace de datos y transporte.

Se ha desarrollado una nueva notacion para escribir direcciones de 16 bytes las cuales se escriben como 8 grupos de 4 digitos digitos hexadecimal, separando los grupos por dos puntos como sigue :

`8000:0000:0000:0000:0123:4567:89AB:CDEF`

Protocolos de control de Internet

Son protocolos que complementan el protocolo IP en la capa de red del modelo OSI o TCP/IP. Su objetivo es gestionar, supervisar y controlar la comunicacion en la red.

ICMP: protoclo de mensaje de control de internet

> Es un protocolo disennado para gestionar y diagnosticar problemas en la red. opera encapsulado en paquetes IP y permite la comunicacion entre dispositivos para informar eventos , errores y pruebas.
>
> Tipo de mensajes ICMP mas importantes

> - Destination unreachable (Destino inaccesible): Se genera cuando un enrutador no puede entregar un paquete a su destino.
> - Time Excedded (timepo excedido): Se envia cuando el TTL (Time to Live) de un paquete llega a cero
> - Parameter Problem (problema de parametros): Indica errores en campos del encabezado IP, sugiriendo problemas en el software del host o enrutador.
> - **Source quench (Fuente disminuida)** :Antiguamente usado para regular el flujo de paquetes en situaciones de congestión.
> - **Redirect (Redirección)** :Un enrutador indica al host emisor que actualice su tabla de rutas con una mejor ruta disponible.
> - **Echo (Eco) y Echo reply (Respuesta de eco)** : Usado para verificar si un dispositivo está activo y alcanzable.
> - Timestamp request/reply (Estampa de tiempo, Petición/Respuesta): Similar a Echo/Echo reply, pero incluye marcas de tiempo para evaluar el rendimiento de la red.
> - **Router advertisement/solicitation (Anuncio/Solicitud de enrutador)** :Facilita la detección de enrutadores cercanos.
>
> `ping` y `traceroute` son herramientas basadas en ICMP

ARP : Protocolo de Resolucion de Direcciones

El **ARP (Address Resolution Protocol)** es un protocolo de red que traduce las direcciones **IP** (de capa de red) en direcciones **MAC** (de capa de enlace de datos). Este protocolo es esencial en redes como Ethernet, donde las tarjetas de interfaz de red (NIC) utilizan direcciones físicas únicas de 48 bits, y no entienden las direcciones IP directamente.

Cómo funciona ARP?

1. **Escenario inicial:**
    - Supongamos que tienes dos computadoras en la misma red local:
        - **PC1** :
        - Dirección IP: `192.168.1.2`
        - Dirección MAC: `AA:BB:CC:DD:EE:01`
        - **PC2** :
        - Dirección IP: `192.168.1.3`

        - Dirección MAC: `AA:BB:CC:DD:EE:02`
    - Ahora, **PC1** quiere enviar un mensaje a **PC2** .
2. **Problema:**
    - PC1 sabe la dirección IP de PC2 (`192.168.1.3`), pero no conoce su dirección MAC (`AA:BB:CC:DD:EE:02`), que es necesaria para enviar un paquete en Ethernet.
3. **Proceso ARP:**
    - **PC1 envía un mensaje de difusión ARP** a toda la red local:
        - "¿Quién tiene la IP `192.168.1.3`? Dime tu dirección MAC".
        - Este mensaje se envía a todas las computadoras de la red (dirección MAC de difusión: `FF:FF:FF:FF:FF:FF`).
4. **Respuesta:**
    - Solo **PC2** responde al mensaje ARP porque es el dueño de la IP `192.168.1.3`.
    - PC2 envía un mensaje directo a PC1 diciendo:
        - "Yo tengo la IP `192.168.1.3` y mi dirección MAC es `AA:BB:CC:DD:EE:02`".
5. **Guardado en caché:**
    - PC1 guarda esta información (IP ↔ MAC) en una tabla llamada **caché ARP** para no tener que preguntar otra vez en un futuro cercano.
6. **Envío del paquete:**
    - Ahora que PC1 conoce la dirección MAC de PC2, puede enviar el paquete directamente a su dirección MAC.

Resumen en pasos

1. Un dispositivo necesita la dirección MAC asociada a una IP.
2. Envía una difusión ARP preguntando: *"¿Quién tiene la IP X?"*
3. El dispositivo con esa IP responde con su dirección MAC.
4. La información se guarda en la caché ARP para futuros usos.
5. El paquete se envía a la dirección MAC encontrada.

¿Por qué es importante?

ARP permite que las computadoras en una red local Ethernet se comuniquen sin necesidad de configurar manualmente las direcciones MAC. Esto simplifica mucho la gestión de redes.

No son importantes las direcciones MAC en un entorno fuera de una red local, ya que las direcciones MAC son especidficas de la capa de enlace de datos.

Funcionamiento cuando H1 (host) envia datos en Internet a H2 (host)

1. H1 revisa si H2 esta en la misma red local (LAN), para ellos revisa la mascar de subred y la direccion IP de H2. Si H2 esta en al misma red, entonces H1 usara ARP para encontrar la direccion MAC de H2 directamnte y enviar el datagrama. SI no tiene exito , sabe que debe enviar el paquete a su router (gateway predeterminada)
2. H1 lanzara un ARP para encontrar la MAC del router. Si la direccion MAC de router no esta en la cache ARP de H1, H1 envia una solicitud ARP para la direccion IP del router. El router responde y H1 usa esta direccion como destino en la trama.
3. El paquete viaja hacia el router , por ahi envia lainformacion hasta que llega a la red correspondiente usando las tabalas de rutas.
4. Cuando llega al destino que alcanza al router que tiene acceso a la red H2, se hace ARP para encontrar la direccion MAC de H2, si la direccion esta en la cache ARP de H2 se usa sino se hace una solictud. H2 responde y envia su MAC para que le puedan enviar la trama, extrae el paquete IP y lo procesa.

DHCP : Protocolo de Configuracion Dinamica de Host

Es un mecanismo que permite asignar automaticamente configuraciones esenciales de red como direcciones IP a los dispositivos en una red, eliminando la necesidad de configurar manualmente a cada equipo.

Resumen del funcionamiento de DHCP

1. **Solicitud de configuración (DHCP DISCOVER):**
   - Cuando un dispositivo arranca, solo tiene su dirección MAC (de la NIC) y no cuenta con una dirección IP.
   - El dispositivo envía un mensaje de difusión **DHCP DISCOVER** para buscar un servidor DHCP.

2. **Oferta de configuración (DHCP OFFER):**
    - El servidor DHCP responde con un mensaje **DHCP OFFER** , asignando una dirección IP disponible.
    - Si el servidor DHCP está en otra red, el router puede retransmitir las solicitudes y respuestas.
3. **Asignación y confirmación:**
    - El host acepta la dirección IP ofrecida enviando un **DHCP REQUEST** .
    - El servidor confirma la asignación mediante un mensaje **DHCP ACK** .
4. **Duración de la asignación (arrendamiento):**
    - Las direcciones IP asignadas tienen un período de arrendamiento.
    - Antes de que expire, el host debe enviar una solicitud de renovación al servidor DHCP.
    - Si el host no renueva, la dirección IP queda libre para ser reasignada.

Protocolo de enrutamiento en Internet

OSPF (Open Shortest Path First)

un estándar ampliamente utilizado para el enrutamiento intradominio. Aquí tienes un resumen estructurado de los puntos más importantes:

1. Contexto del Protocolo OSPF

- **Clasificación:** Protocolo de enrutamiento intradominio (puerta de enlace interior).
- **Origen:** Diseñado por IETF en 1988, se convirtió en estándar en 1990.
- **Finalidad:** Superar limitaciones de protocolos anteriores como RIP (e.g., convergencia lenta, problema de conteo al infinito).
- **Base:** Protocolo de estado del enlace basado en grafo.

2. Requerimientos y Características Clave

1. **Publicación Abierta:** Garantiza la disponibilidad del protocolo a cualquier organización.
2. **Soporte para Métricas de Distancia:** Incluye distancia física, retardo, entre otras.
3. **Adaptabilidad Dinámica:** Responde automáticamente a cambios en la topología.

4. **Enrutamiento por Tipo de Servicio:** Originalmente diseñado para IP con diferentes tipos de tráfico.
5. **Balanceo de Carga (ECMP):** Divide tráfico entre rutas de igual costo para optimizar el rendimiento.
6. **Jerarquías de Enrutamiento:** Soporte para áreas jerárquicas que facilitan la escalabilidad.
7. **Seguridad Básica:** Evita la inyección de información falsa.
8. **Soporte para Enlaces y Túneles:** Compatible con múltiples tipos de redes y configuraciones.

3. Estructura y Operación

- **Modelado con Grafos:**
  - Representa redes como grafos con nodos y arcos ponderados.
  - A cada enlace se le asigna un costo.
- **Tipos de Enrutadores:**
  - **Internos:** Operan dentro de un área específica.
  - **De Frontera de Área:** Conectan áreas a la red troncal.
  - **Troncales:** Gestionan la red troncal (Área 0).
  - **De Límite de AS:** Conectan sistemas autónomos distintos.
- **Áreas:**
  - Dividen grandes sistemas autónomos en unidades manejables.
  - Cada AS incluye un área troncal que conecta todas las demás áreas.

4. Mensajes OSPF

Cinco tipos principales:

1. **Hello:** Descubre vecinos.
2. **Link State Update:** Actualiza información de estado del enlace.
3. **Link State Ack:** Confirma recepción de actualizaciones.
4. **Database Description:** Anuncia entradas de estado del enlace.
5. **Link State Request:** Solicita información específica.

5. Proceso Operativo

- Los enrutadores intercambian mensajes para construir una base de datos topológica.

- Cada enrutador calcula las rutas más cortas dentro de su área mediante algoritmos como Dijkstra.
- Los enrutadores de frontera y troncales manejan rutas inter-áreas y hacia otros sistemas autónomos.

6. Ventajas

- Escalabilidad mediante áreas jerárquicas.
- Balanceo de carga para mejor rendimiento.
- Respuesta rápida a cambios de topología.
- Protocolo abierto y ampliamente adoptado.

Caracteristicas

- Algoritmo de Dijkstra , estado de enlace , velocidad, latencia y congestion
- Envia actualizaciones modulares

BGP : protocolo de Puerta Enlace Exterior

BGP (Border Gateway Protocol): protocolo de enrutamiento externo que se utiliza para intercambiar información de enrutamiento entre sistemas autónomos (AS). A diferencia de los protocolos de enrutamiento internos como OSPF, que operan dentro de una única red, BGP es crucial para la conectividad entre diferentes redes en Internet. BGP utiliza un enfoque de política de enrutamiento basado en políticas de acceso y preferencias de ruta para determinar las rutas más eficientes para el tráfico entre AS. Esto permite a BGP manejar la complejidad y la escalabilidad de Internet, donde hay millones de routers y redes.

- **BGP se utiliza para interconectar redes (AS)** en lugar de simplemente administrar las rutas dentro de una red.
- **Las decisiones de enrutamiento no se basan solo en la distancia** , sino en políticas que pueden estar relacionadas con costos, seguridad o acuerdos comerciales.
- **Escalabilidad** : BGP maneja redes masivas como las de Internet, donde existen millones de rutas y redes interconectadas

5.6.8 Multidifusión de Internet

La **multidifusión** permite que un proceso envíe datos a múltiples receptores de forma simultánea, en lugar de tener que enviar una copia a cada receptor individualmente (como en la comunicación uno a uno). Esto es útil para aplicaciones como transmisión de eventos en vivo, actualizaciones de programas, o conferencias telefónicas entre varios participantes.

- **Direcciones IP de clase D** : Estas direcciones son usadas para identificar grupos de hosts que forman parte de una red de multidifusión. Existen más de 250 millones de grupos posibles gracias a los 28 bits disponibles para estas direcciones.
- **Direcciones de multidifusión locales** : Estas direcciones están reservadas para la comunicación dentro de una red local (LAN), como 224.0.0.1 (para todos los sistemas en una LAN) o 224.0.0.251 (para servidores DNS en una LAN).
- **Protocolo IGMP** : Utilizado por los hosts para unirse o abandonar grupos de multidifusión. Los enrutadores multicast envían consultas periódicas a los hosts para que informen a qué grupos pertenecen.
- **Protocolos de enrutamiento para multidifusión** : Como **PIM (Protocol Independent Multicast)** , que se utiliza dentro de un AS para crear árboles de expansión de multidifusión. PIM tiene dos modos: **denso** (cuando los miembros están distribuidos por toda la red) y **disperso** (cuando los miembros están distribuidos en lugares específicos como suscriptores de televisión por IP).

5.6.9 IP Móvil

La **IP móvil** permite que los usuarios de Internet mantengan su conexión mientras se desplazan entre diferentes redes, como un dispositivo que cambia de ubicación de una red a otra (por ejemplo, al moverse entre diferentes puntos de acceso Wi-Fi).

- **Desafíos de la movilidad** : En un sistema de direccionamiento IP tradicional, los paquetes enviados a una dirección IP fija no llegarán al dispositivo si este se traslada a otro lugar. Esto plantea problemas para mantener la conectividad sin interrumpir las aplicaciones en uso.
- **Agentes de base** : Cada red que permite la movilidad debe tener un "agente de base" que facilite la comunicación con los dispositivos móviles. Cuando un host móvil se conecta a una red foránea, obtiene una nueva dirección IP (llamada dirección de custodia) y notifica al agente de base sobre su nueva ubicación. Los paquetes destinados al host móvil se envían a través de un túnel al agente de base, que los redirige al dispositivo móvil.

- **ARP y Proxy ARP** : Para asegurarse de que los paquetes lleguen correctamente a los hosts móviles, se utiliza ARP. Cuando el móvil está en su ubicación original, responde con su dirección Ethernet. Cuando está en una red foránea, el agente de base responde a las consultas ARP en lugar del móvil.
- **Problemas de NAT y filtrado de ingreso** : Los paquetes enviados por el host móvil desde una red foránea pueden ser descartados debido al filtrado de ingreso en los enrutadores, ya que su dirección IP de origen no coincide con su ubicación real. La solución es utilizar túneles para enviar los paquetes correctamente.
- **IPv6 para movilidad** : En IPv6, la solución de IP móvil es más eficiente, ya que los paquetes iniciales siguen una ruta larga, pero luego se optimizan para tomar una ruta directa entre el móvil y el destino.

RESUMEN

La capa de red se encarga de enrutar los paquetes desde la fuente hasta el destino, proporcionando servicios a la capa de transporte. Dependiendo de la red, este proceso puede basarse en datagramas, donde se decide la ruta para cada paquete, o en circuitos virtuales, donde la ruta se define al establecer el circuito.

Algoritmos de enrutamiento:

1. **Inundación:** Es un algoritmo simple que envía un paquete a todas las rutas disponibles.
2. **Enrutamiento por vector de distancia:** Cada nodo mantiene una tabla con las distancias a los demás nodos.
3. **Enrutamiento de estado del enlace:** Cada nodo mantiene información sobre el estado de sus enlaces.

Estos algoritmos se adaptan a los cambios en la topología de la red, buscando rutas eficientes y cortas.

Enrutamiento avanzado:

- **Jerarquía:** Utilizada en redes grandes para organizar las rutas en varios niveles.
- **Enrutamiento para hosts móviles:** Permite a los dispositivos moverse sin perder la conectividad.
- **Difusión, multidifusión y anycast:** Permiten enviar paquetes a varios destinatarios (multidifusión) o a uno entre varios posibles (anycast).

Congestión y calidad de servicio (QoS): Las redes pueden congestionarse fácilmente, lo que lleva a aumentos en los retrasos y pérdida de paquetes. Los diseñadores de redes deben crear infraestructuras con suficiente capacidad, elegir rutas descongestionadas y aplicar métodos de control como la reducción de la velocidad de las fuentes. La calidad del servicio (QoS) se refiere a asegurar que los paquetes de ciertas aplicaciones se entreguen dentro de ciertos parámetros (por ejemplo, con un retardo mínimo o máxima velocidad de transmisión). Los métodos comunes para garantizar QoS incluyen:

- **Modelado de tráfico**
- **Reservación de recursos en enrutadores**
- **Control de admisión**

Interconexión de redes: Cuando diferentes redes se interconectan, pueden surgir problemas como:

- Diferentes tamaños de paquetes, lo que requiere fragmentación.
- Diferentes protocolos de enrutamiento, lo que necesita protocolos comunes entre redes.
- La **tunelización** puede ayudar en redes hostiles o cuando las redes de origen y destino son diferentes.

Protocolos en la capa de red:

- **IP (Internet Protocol):** El protocolo base para el enrutamiento.
- **ICMP (Internet Control Message Protocol):** Usado para mensajes de control.
- **ARP (Address Resolution Protocol):** Asocia direcciones IP con direcciones MAC.
- **DHCP (Dynamic Host Configuration Protocol):** Asigna direcciones IP dinámicamente.
- **MPLS (Multiprotocol Label Switching):** Permite enrutar paquetes IP a través de diversas redes.
- **OSPF (Open Shortest Path First):** Un protocolo de enrutamiento utilizado dentro de una red.
- **BGP (Border Gateway Protocol):** Utilizado para enrutar entre diferentes redes.