**Filmina 7 Direccionamiento**

**Introducción**

La capa de red implementa el servicio de comunicación host a host. Existe un componente de la capa de red en hosts y routers.

Suponiendo 2 host en una red formada con varios routers. Si H1 envía datos a H2, La capa de red en H1 toma segmentos de la capa de transporte en H1, encapsula cada segmento en un datagrama y, a continuación, envía los datagramas al router más próximo, R1. En el host de recepción, H2, la capa de red recibe los datagramas de su router más próximo R2, extrae los segmentos de la capa de transporte y los entrega a capa de transporte de H2. La función principal de los routers es reenviar los datagramas desde los enlaces de entrada a los enlaces de salida.

**Reenvío y enrutamiento**

La **función de la capa de red** es simple: transporta paquetes desde un host emisor a un host receptor. En la realización de esta tarea podemos identificar dos importantes funciones de la capa de red:

**• Reenvío:**Cuando un paquete llega al enlace de entrada de un router, este tiene que pasar el paquete al enlace de salida apropiado. El reenvío implica la transferencia de un paquete desde una interfaz de un enlace de entrada a una interfaz de un enlace de salida dentro de un *mismo* router.

• **Enrutamiento:**La capa de red tiene que determinar la ruta o camino que deben seguir los paquetes a medida que fluyen de un emisor a un receptor. Los algoritmos que calculan estas rutas se conocen como **algoritmos de enrutamiento.** El enrutamiento implica a *todos* los routers de una red. Cuyas interacciones colectivas mediante los protocolos de enrutamiento determinan las rutas que seguirán los paquetes en sus viajes desde el origen hasta el destino.

Todo router tiene **una tabla de reenvío**. Un router reenvía un paquete examinando el valor de un campo de la cabecera del paquete entrante y utilizando después ese valor para indexarlo dentro de la tabla de reenvió del router. El resultado de la tabla de reenvió indica a cuál de las interfaces del enlace de salida del router será reenviado el paquete. Dependiendo del protocolo de la capa de red, este valor de la cabecera del paquete podría ser la dirección de destino del paquete o una indicación de la conexión a la que pertenece el paquete. El algoritmo de enrutamiento determina los valores que se introducen en las tablas.

**FUNCIONES BÁSICAS DE LOS PROTOCOLOS**

Se destacan 2 categorías:

* Direccionamiento.
* Servicios de transmisión.

**DIRECCIONAMIENTO**

Abarca una serie de cuestiones, incluyendo:

* **Nivel de direccionamiento.**
* **Alcance del direccionamiento.**
* **Identificadores de conexión.**
* **Modo de direccionamiento.**

-El **nivel de direccionamiento** es el nivel en la arquitectura de comunicaciones en el cual una entidad es designada. Generalmente se **asocia una dirección única con cada sistema final (estación de trabajo) y cada sistema intermedio (router)**. Esta dirección es, en general, una **dirección en el nivel de red**. En el caso de la **arquitectura TCP/IP, es la dirección IP**, En la **arquitectura OSI, es el punto de acceso al servicio de red (NSAP)**. La dirección en el nivel de red se utiliza para encaminar una PDU a través de una o varias redes hasta el sistema indicado por la dirección en el nivel de red contenida en la PDU.

Una vez que los datos alcanzan el sistema destino, éstos deben ser dirigidos a algún proceso o aplicación en el sistema. Cada aplicación recibe un identificador único denominado puerto en la arquitectura TCP/IP y punto de acceso al servicio (SAP, *Ser*v*ice Access Point*) en la arquitectura OSI.

-El **alcance del direccionamiento** es una cuestión relacionada a la dirección. La dirección IP es una direcciónglobal. Las características clave de una dirección global son las siguientes:

* **Ausencia de ambigüedad global:** una dirección global identifica únicamente un sistema. Sepermiten los sinónimos. Esto es, un sistema puede tener más de una dirección global.
* **Aplicabilidad global (e2e):** es posible que desde cualquier dirección global se identifique cualquier otra dirección global, ubicada en cualquier sistema, mediante la utilización de la dirección global del otro sistema.

Esto permite a una colección de redes encaminar datos procedentes de cualquier sistema conectado a una red hacia otro sistema conectado a cualquier otra red.

**-Los identificadores de conexión** aparecen con las **transferencias orientadas a conexión** en lugar de transferencias no orientadas a conexión. En transferencias **no orientadas a conexión** se utiliza un identificador global para cada transmisión de datos. En el caso de las **transferencias orientadas a conexión**, es deseable en algunas ocasiones usar solamente un identificador de conexión durante la fase de transferencia de datos**. Ventajas del uso del identificador de dirección:**

* **Reducción de la sobrecarga:** los identificadores de conexión son más cortos que los identificadores globales.
* **Encaminamiento:** en la configuración de una conexión se puede definir una ruta fija. El identificador de la conexión sirve para identificar la ruta en los sistemas intermedios.
* **Multiplexación:** una entidad puede desear disfrutar de más de una conexión simultáneamente. Así, las PDU entrantes deben ser identificadas por su identificador de conexión.
* **Uso de información de estado:** una vez que una conexión está establecida, los sistemas finales pueden mantener información de estado concerniente a la misma.

**- En el modo de direccionamiento**:

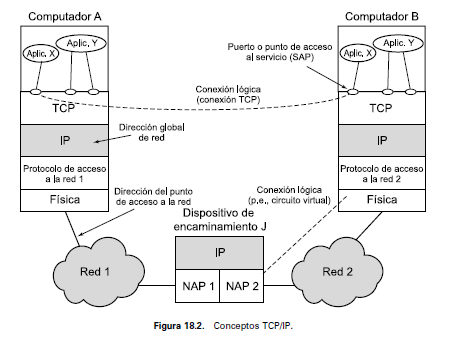
En su forma más común, una dirección se refiere a un sistema individual o a un puerto; en este caso nos referimos a una dirección individual o **unidifusión** (*unicast*).

Es también posible que una dirección se puede referir a más de una entidad o puerto. Tal dirección identifica simultáneamente a múltiples receptores para los datos. Una dirección para múltiples receptores puede ser de **difusión** (*broadcast*), destinada a todas las entidades dentro de un dominio, o de **multidistribución** (*multicast*), para un subconjunto específico de entidades.

**SERVICIOS DE TRANSMISIÓN**

Entre ellos:

* **Prioridad:** ciertos mensajes, como los de control, pueden necesitar llegar a la entidad destino con un retardo mínimo. Así, la prioridad podría ser asignada en función del mensaje o en función de la conexión.
* **Calidad de servicio:** ciertas clases de datos pueden requerir un umbral de rendimiento mínimo o de retardo máximo.
* **Seguridad:** los mecanismos de seguridad, restringiendo el acceso, pueden ser invocados.



**PRINCIPIOS DE LA INTERCONEXIÓN ENTRE REDES- REQUISITOS**

Los requisitos globales para un sistema de interconexión entre redes son los que siguen a continuación:

**1.** Proporcionar un enlace entre redes. Como mínimo, se necesita conexión física y de control del enlace.

**2.** Proporcionar el encaminamiento y entrega de los datos entre procesos en diferentes redes.

**3.** Proporcionar un servicio de contabilidad que realice un seguimiento de la utilización de las diferentes redes y dispositivos de encaminamiento y mantenga información de estado.

**4.** Proporcionar los servicios mencionados de forma que no se requiera la modificación de la arquitectura de red de cualquiera de las redes interconectadas. El sistema de interconexión entre redes se debe acomodar a las diversas diferencias existentes entre las distintas redes. Algunas de estas diferencias son:

* **Diferentes esquemas de direccionamiento:** se debe proporcionar un esquema de direccionamiento de red global.
* **Diferente tamaño máximo de paquete:** ofrecer fragmentación.
* **Diferentes mecanismos de acceso a la red.**
* **Diferentes valores de expiración de los temporizadores.**
* **Recuperación de errores:** El servicio de interconexión de redes no debería depender de la capacidad de recuperación de errores de las redes individuales.
* **Informes de estado:** el sistema de interconexión proporcione información de la actividad de interconexión a los procesos.
* **Técnicas de encaminamiento:** El sistema de interconexión entre redes debe ser capaz de encaminar los datos adaptativamente entre las estaciones de las diferentes redes.
* **Control de acceso del usuario:** cada red tendrá su propia técnica de control de acceso de los usuarios. Estas técnicas se deben solicitar por el sistema de interconexión según se necesite.
* **Conexión, sin conexión:** las redes individuales pueden proporcionar un servicio orientado a conexión o no orientados a conexión. Es deseable que el servicio entre redes no dependa de la naturaleza del servicio de conexión de las redes individuales.

**ENFOQUES SOBRE LA ARQUITECTURA:**

una característica clave en una arquitectura de interconexión de redes es si el modo de operación es orientado a la conexión o no orientado a la conexión.

**Funcionamiento orientado a la conexión:**

En este modo de funcionamiento se supone que cada red proporciona un servicio en forma de conexión. Esto es, se establece una conexión lógica a nivel de red entre cualquier par de sistemas finales conectados a la misma red. Primero se establece la conexión, y a continuación se produce el intercambio de datos. Opción de operación:

1. Los IS (Sistemas de Interconexión) se utilizan para conectar dos o más subredes: cada IS aparece como un ES (sistema final) en cada una de las redes a la que está conectado.
2. Cuando el ES A quiere intercambiar datos con el ES B, se establece una conexión lógica entre ellos que consiste en la concatenación de una secuencia lógica de conexiones a través de subredes. Esta secuencia es tal que forma un camino desde el ES A al ES B.
3. Las conexiones lógicas individuales dentro de una red están realizadas por varios IS. Cualquier tráfico que llega a un IS en una conexión lógica se retransmite en una segunda conexión lógica y viceversa.

Un IS orientado a conexión realiza las siguientes funciones claves:

* Retransmisión: las unidades de datos que llegan de una red vía el protocolo de la capa de red se retransmiten a otra red. El tráfico se conduce a través de conexiones lógicas que están enlazadas por los IS.
* Encaminamiento: cuando se va a establecer una conexión lógica extremo a extremo, consistente en una secuencia de conexiones lógicas, cada IS en la secuencia debe tomar una decisión de encaminamiento que determina el siguiente salto de secuencia.

**Funcionamiento no orientado a la conexión:**

Mientras que el modo de funcionamiento orientado a conexión se corresponde con el mecanismo de circuito virtual de una red de conmutación de paquetes, el modo de operación no orientado a conexión se corresponde con el mecanismo de datagramas de una red de conmutación de paquetes.

Cada PDU de red se trata independientemente y se encamina desde el ES origen al ES destino a través de una serie de dispositivos de encaminamiento y redes. Para cada unidad de datos transmitida por A, A realiza una decisión sobre que dispositivos de encaminamiento debería recibir la unidad de datos. La unidad de datos salta a través del conjunto de redes de un dispositivo de encaminamiento al siguiente hasta que alcanza la subred destino.

En cada dispositivo de encaminamiento se toma una decisión de encaminamiento relativa al siguiente salto. Así, diferentes unidades de datos pueden viajar por diferentes rutas entre el ES origen y destino.

Todos los ES y todos los dispositivos de encaminamiento comparten un protocolo de la capa de red común conocido genéricamente como protocolo de interconexión de redes.

**MODELOS DE SERVICIO DE RED**

Define **las características del transporte terminal a terminal** de los paquetes entre los sistemas terminales emisor y receptor. Entre los **servicios específicos** que la capa de red podría proporcionar se incluyen:

• **Entrega garantizada:** Este servicio garantiza que el paquete terminará por llegar a su destino.

**• Entrega garantizada con retardo limitado:** garantiza la entrega del paquete en un límite de retardo.

**• Entrega de los paquetes en orden.** Llegan al destino en el orden en que fueron enviados.

• **Ancho de banda mínimo garantizado.** Simula un enlace de transmisión con una velocidad de bit específica entre los host emisor y receptor.

**• Fluctuación Máxima garantizada:** este servicio garantiza que el intervalo de tiempo transcurrido entre la transmisión de dos paquetes sucesivos en el emisor es igual al intervalo de tiempo que transcurre entre su respectiva recepción en el destino.

**• Servicios de seguridad.** Uso de clave, cifrado de carga útil, etc.

La capa de red de Internet proporciona un único servicio conocido como **servicio de mejor esfuerzo**. Con un servicio de mejor esfuerzo, la temporización relativa entre paquetes no está garantizada, tampoco está garantizado que los paquetes se reciban en el orden que fueron emitidos y tampoco se garantiza la entrega de los paquetes transmitidos.

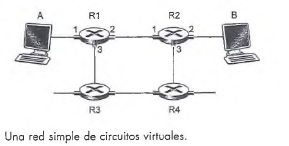
**REDES DE CIRCUITOS VIRTUALES Y DE DATAGRAMAS**

Una capa de red también puede proporcionar un **servicio no orientado a conexión** o un **servicio orientado a la conexión**. Un servicio de la capa de red orientado a la conexión comienza con un proceso de acuerdo entre los hosts de origen y de destino; y un servicio de la capa de red sin conexión no realiza ninguna tarea preliminar de acuerdo. En las principales arquitecturas, la capa de red proporciona bien un servicio sin conexión host a host o un servicio orientado a la conexión host a host, pero no ambos. Las redes de computadoras que solo proporcionan un servicio de conexión en la capa de red se conocen como **redes de circuitos virtuales** (VC); las redes que solo proporcionan un servicio sin conexión en la capa de red se denominan **redes de datagramas.**

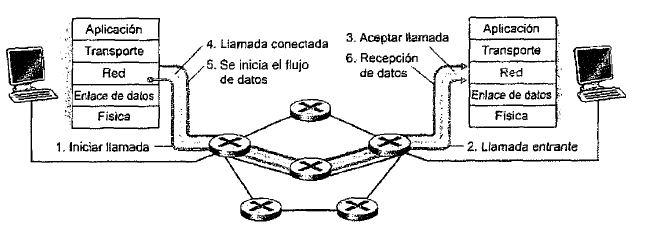
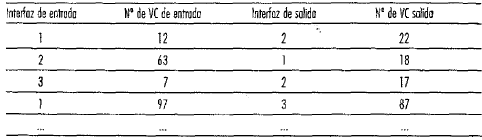
**Modelo de direccionamiento - Redes de circuitos virtuales (Orientado a la conexión)**

Un circuito virtual consta de:

* **Una ruta** entre los hosts de origen y de destino
* **números de VC**, un numero para cada enlace a lo largo de la ruta
* **entradas en la tabla de reenvió de cada router** existente a lo largo de la ruta.

Un paquete que pertenece a un circuito virtual transportara un número de VC en su cabecera. Dado que un circuito virtual puede tener un número de VC diferente en cada enlace, cada router interviniente tiene que sustituir el número de VC de cada paquete que le atraviesa por un nuevo número de VC. Este nuevo número de VC se obtiene de la tabla de reenvío. Cuando se configura un nuevo VC se agrega a la tabla de reenvío

Los routers de la red tienen que mantener **información de estado** **de la conexión** para las conexiones activas. Específicamente, cada vez que se establece una conexión nueva en un router, tiene que añadirse una nueva entrada de conexión a la tabla de reenvió del router; y cada vez que una conexión se libera, la entrada debe borrarse de la tabla.



En un circuito virtual existen tres fases identificables:

• **Configuración del VC.**La capa de transporte especifica la dirección del receptor y espera a que la red configure el circuito virtual. **La capa de red:**

* Determina la **ruta entre el emisor y el receptor.**
* Determina el **número de VC para cada enlace de la ruta.**
* **Añade una entrada a la tabla de reenvió de cada router existente a lo largo de la ruta.**
* **También puede reservar recursos a lo largo de la ruta del VC.**

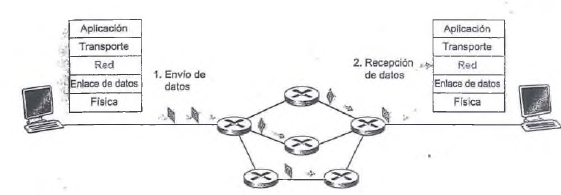
•**Transferencia de datos.** Una vez que se ha stablecido el circuito virtual, los paquetes se transmiten por el mismo.

**• Terminación del VC.** Inicia cuando el emisor (o el receptor) informa a la capa de red de su deseo de terminar el circuito virtual. La capa de red informará al sistema terminal del otro lado de la red de la terminación de la llamada y actualizará las tablas de reenvió de cada uno de los routers de la ruta, para indicar que ese circuito virtual ya no existe.

**Modelo de direccionamiento - Redes de datagramas (No orientada a la conexión)**

En una **red de datagramas,** cada vez que un sistema terminal desea enviar un paquete marca el paquete con la dirección del sistema terminal de destino y luego introduce el paquete en la red.

**No se configura ningún circuito virtual**, por ende, los **routers no mantienen ninguna información de estado** de los mismos. Cuando un paquete se transmite desde un origen a un destino pasa a través de una serie de routers. Cada uno de estos routers utiliza la dirección de destino del paquete para reenviar dicho paquete. **Cada router tiene una tabla de reenvió** que asigna **direcciones de destino a interfaces de enlace**; cuando un paquete llega a un router, este utiliza la dirección de destino del paquete para buscar la interfaz del enlace de salida apropiado en la tabla de reenvió. Después, el router reenvía el paquete a esa interfaz del enlace de salida.



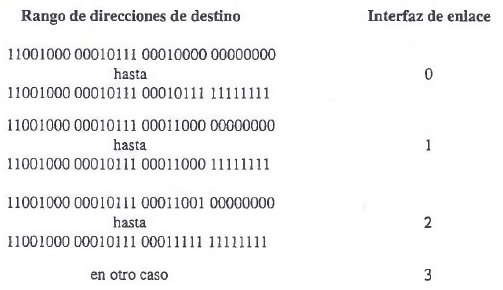
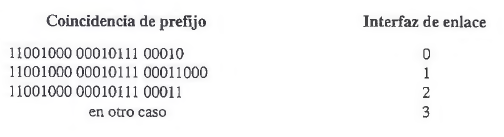
Si todas las direcciones de destino tienen una longitud de 32 bits, y el router tiene enlaces numerados, y que los paquetes deben ser reenviados a las interfaces de enlace de acuerdo a un rango de direcciones se obtiene una tabla de reenvió con pocas entradas, por ejemplo:

Tabla de reenvío resultante



Con este tipo de tabla de reenvió, el router busca la coincidencia de un **prefijo** de la dirección de destino del paquete con las entradas de la tabla; si existe una coincidencia, el router reenvía el paquete a un enlace asociado con esa coincidencia. Por ejemplo, si la dirección de destino es 11001000 00010111 00010110 10100001; como el prefijo de 21 bits de esta dirección coincide con la primera entrada de la tabla, el router reenvía el paquete a la interfaz de enlace 0.

Una dirección de destino puede corresponderse con más de una entrada. Cuando existen varias coincidencias, el router aplica la **regla de coincidencia** **con el prefijo más largo;** es decir, busca la entrada más larga de la tabla con la que exista una coincidencia y reenvía el paquete a la interfaz de enlace asociada con el prefijo más largo.

Los routers en las redes de datagramas mantienen información del estado de reenvío en sus tablas de reenvío. Sin embargo, la frecuencia a la que esta información del estado de reenvío cambia es lenta. De hecho, en una red de datagramas las tablas de reenvío son modificadas por los algoritmos de enrutamiento.

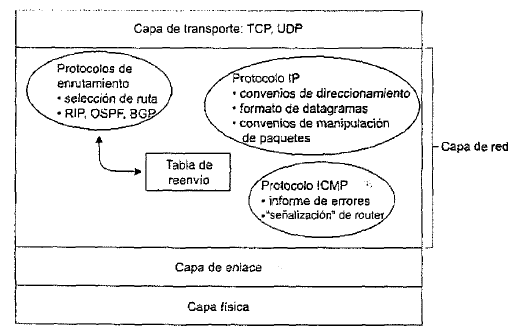
Puesto que las tablas de reenvío en las redes de datagramas pueden ser modificadas en cualquier instante, una serie de paquetes enviados desde un sistema terminal a otro puede seguir caminos distintos a través de la red y pueden llegar desordenados.

**PROTOCOLO DE INTERNET (IP): REENVÍO Y DIRECCIONAMIENTO EN INTERNET**

Actualmente hay dos versiones en uso de IP: protocolo IP versión 4, que habitualmente se denomina IPv4 [RFC 791]. La versión 6 de IP [RFC 2460; RFC 4291], protocolo propuesto para sustituir a IPv4.

La capa de red de Internet tiene tres **componentes** principales.

* El **protocolo IP**.
* El componente de **enrutamiento**, el cual determina la ruta que sigue un datagrama desde el origen al destino.
* El **protocolo de control** que permite informar de la existencia de errores en los datagramas y contesta a las solicitudes de determinada información de la capa de red. En este caso el protocolo de información de control y de errores de la capa de red de Internet, el protocolo ICMP (*Internet* *Control Message Protocol).*



**Formato del datagrama IP:**

A los paquetes de la capa de red se los refiere como datagramas.

La sintaxis y semántica del datagrama IPv4:

* **Numero de versión:** Estos 4 bitsindican el número de la versión del protocolo IP del datagrama.
* **Longitud de la cabecera Internet (IHL):** indica donde comienzan los datos del datagrama.
* **Tipo de servicio (8 bits):** permite diferenciar entre distintos tipos de datagramas IP.
* **Longitud del datagrama:** longitud total del datagrama IP en bytes.
* **Identificador, indicadores, desplazamiento de fragmentación:** intervienen en la fragmentación IP, el identificador permite identificar de forma única un datagrama, el de desplazamiento indica el lugar del fragmento dentro del datagrama.
* **Tiempo de vida TTL:** tiempo que se le permite al datagrama permanecer en la red.
* **Protocolo:** se emplea cuando un datagrama alcanza su destino final. Indica el protocolo de la capa de transporte al que se le pasaran los datos del datagrama.
* **Suma de comprobación de cabecera:** código de detección de errores aplicable a la cabecera.
* **Dirección IP de origen y destino**: dirección IP del origen que crea el datagrama y dirección IP del que lo recibe.
* **Opciones:** contiene las opciones solicitadas por el usuario que envía los datos.
* **Relleno:** asegura que la cabecera del datagrama tenga una longitud de 32 bits.
* **Datos:** es el campo de datos del datagrama IP suele contener el segmento de la capa de transporte que va a entregarse al destino.

**CUESTIONES DE DISEÑO**

Éstas son:

* **Encaminamiento.**
* **Tiempo de vida de los datagramas**.
* **Fragmentación y reensamblado.**
* Control de errores.
* Control de flujo.

**Encaminamiento**

El encaminamiento se hace manteniendo de una tabla de encaminamiento en cada dispositivo de encaminamiento y en cada sistema final. En esta tabla se da, para cada red posible de destino, el siguiente dispositivo de encaminamiento al que se deberá enviar el datagrama IP.

La tabla puede ser estática o dinámica. La estática puede contener rutas alternativas en caso que un router no esté disponible, la dinámica es más flexible para los errores y congestión. En Internet, por ejemplo, cuando un dispositivo de encaminamiento se desconecta, todos sus vecinos emitirán un informe de estado, permitiendo a otros dispositivos de encaminamiento y estaciones que actualicen sus tablas de encaminamiento.

Las tablas de encaminamiento también se pueden utilizar para ofrecer otros servicios de inter conexión entre redes, como seguridad y prioridad.

Un servicio relacionado con el encaminamiento es el registro de la ruta. Para registrar la ruta, cada dispositivo de encaminamiento incorpora su dirección internet a una lista de direcciones que lleva el datagrama.

**Tiempo de vida de los datagramas**

Si se utiliza un encaminamiento dinámico, existe la posibilidad de que un datagrama viaje indefinidamente a través del conjunto de redes. Esto no es aconsejable, Para evitar problemas, cada datagrama se puede marcar con un tiempo de vida. Una vez que ha transcurrido este tiempo de vida, el datagrama se descarta.

Se puede implementar esta función con un contador de saltos cada vez que un datagrama pasa a través de un dispositivo de encaminamiento, se decremento el contador; O mediante una medida de tiempo, los dispositivos de encaminamiento deben conocer el tiempo transcurrido desde que el datagrama cruzó por última vez un dispositivo de encaminamiento, para conocer cuánto tiene que decrementar el campo de tiempo de vida. Esto requeriría algún mecanismo global de sincronización.

**Fragmentación y reensamblado**

Las redes individuales en un conjunto de redes pueden especificar tamaños máximos de paquetes diferentes. Los dispositivos de encaminamiento pueden necesitar fragmentar los datagramas de entrada en unidades más pequeñas, llamadas segmentos o fragmentos, antes de transmitirlos en la red siguiente. Si los datagramas se pueden fragmentar se deben reensamblar.

En IP, los fragmentos de los datagramas se reensamblar en el sistema final destino. La técnica de fragmentación de IP usa los siguientes campos en la cabecera IP:

* Identificador de la unidad de datos (ID).
* Longitud de los datos.
* Desplazamiento.
* Indicador de más datos.

El *ID* es un medio de identificar de forma única un datagrama originado en el sistema final. La *longitud de los datos* indica la longitud del campo de datos de usuario, expresado en octetos, y el campo *desplazamiento* es la posición de un fragmento de los datos de usuario en el campo de datos en el datagrama original, en múltiplos de 64 bits.

El sistema final origen crea un datagrama con una longitud de datos igual a la longitud entera del campo de datos, con *desplazamiento* = 0 y el indicador de *más datos* establecido a 0 (falso).

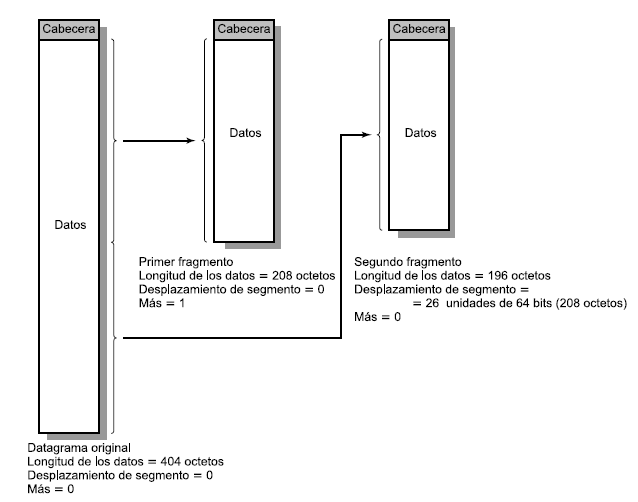
Para fragmentar un datagrama grande en dos piezas, un módulo IP en un dispositivo de encaminamiento realiza las siguientes tareas:

**1.** Crea dos nuevos datagramas y copia los campos de la cabecera del datagrama original en los datagramas nuevos.

**2.** Divide el campo de datos de usuario en dos porciones aproximadamente iguales con límites de 64 bits, situando una porción en cada datagrama nuevo. La primera porción debe ser un múltiplo de 64 bits (8 octetos).

**3.** Establece la *longitud de datos* del primer datagrama a la longitud de los datos insertados y establece a 1 (cierto) el indicador de *más datos*. El campo *desplazamiento* no se cambia.

**4.** Establece la *longitud de datos* del segundo datagrama a la longitud de los datos insertados, y añade la longitud de la primera porción de datos dividida por 8 al campo *desplazamiento*. El indicador de *más datos* permanece igual.

Para reensamblar un datagrama debe haber suficiente espacio de memoria temporal en el momento de reensamblar. Conforme los fragmentos con el mismo ID llegan, se insertan los campos de datos en la posición correcta en la memoria temporal hasta que el campo datos entero se reensambla, lo que se consigue cuando existe un conjunto contiguo de datos comenzando con un *desplazamiento* de cero y terminando con datos de un segmento con el indicador de *más datos* puesto a falso.

Considerando que IP no garantiza la entrega para saber cuándo abandonar un reensamblado se puede iniciar un temporizador cuando se recibe el primer fragmento o tener en cuenta el tiempo de vida del datagrama en cualquier caso cuando este tiempo expira se descarta el reensamblado.

**Fragmentación del datagrama IP**

No todos los protocolos de la capa de enlace pueden transportar paquetes de la capa de red del mismo tamaño. Por ejemplo, las tramas Ethernet pueden transportar hasta 1.500 bytes de datos, ATM 53 octetos.

La cantidad máxima de datos que una trama de la capa de enlace puede transportar se conoce como **unidad máxima de transmisión** (MTU*).* Como cada datagrama IP se encapsula dentro de una trama de la capa de enlace para ir de un router al siguiente, la MTU del protocolo de la capa de enlace impone un límite a la longitud de un datagrama IP. Esto no es un problema, pero sí que cada uno de los enlaces existentes a lo largo de la ruta entre el emisor y el destino puede utilizar diferentes protocolos de la capa de enlace y cada uno de estos protocolos puede utilizar una MTU diferente.

La solución consiste en fragmentar los datos del datagrama IP en dos o más datagramas IP más pequeños, encapsular cada uno de los datagramas IP más pequeños en una trama de la capa de enlace distinta y enviar dichas tramas a través del enlace de salida. Cada uno de estos datagramas más pequeños se conoce como **fragmentos**.

Los fragmentos tienen que ser **reensamblados** antes de llegar a la capa de transporte del destino. **Para mantener el núcleo de la red simple, los sistemas terminales reensamblan los datagramas en lugar de los routers de red.**

Cuando un host de destino recibe una serie de datagramas del mismo origen, tiene que determinar si algunos son fragmentos de algún otro datagrama original más grande. Si algunos datagramas son fragmentos, tiene que determinar cuándo recibió el último fragmento y como debe ensamblar los fragmentos para formar el datagrama original.

Para que el host de destino pueda llevar a cabo estas tareas de reensamblado, se incluyeron los campos ***identificación, indicador* y *desplazamiento de fragmentación***en la cabecera del datagrama IP.

Cuando se crea un datagrama, el host emisor marca el datagrama con un número de identificación (ID), direcciones de origen y de destino. Cuando un router necesita fragmentar un datagrama, cada datagrama fragmento resultante se marca con la dirección de origen, destino y el ID del datagrama original. Cuando el destino recibe una serie de datagramas procedentes del mismo host emisor, puede examinar los números de identificación de los datagramas para determinar cuáles de ellos son fragmentos de un datagrama más largo.

**Puesto que IP es un servicio no fiable, es posible que uno o más fragmentos no lleguen a su destino.** Para que el host de destino se asegure de que ha recibido el último fragmento del datagrama original, el último fragmento tiene un bit indicador puesto a 0, mientras que los demás fragmentos tienen el bit indicador puesto a 1. Para que el host de destino pueda reensamblar los fragmentos en el orden apropiado, se utiliza el campo **desplazamiento** para especificar qué posición ocupa dentro del datagrama IP original el fragmento.

En el destino, la carga útil del datagrama se pasa a la capa de transporte solo después de que la capa IP haya reconstruido completamente el datagrama IP original. Si uno o más de los fragmentos no llegan al destino, el datagrama incompleto se descarta y no se pasa a la capa de transporte.

**DIRECCIONAMIENTO IPV4**

¿Cómo los hosts y los routers están conectados en la red? Normalmente un host dispone de un único enlace hacia la red, cuando IP en el host desea enviar un datagrama, lo hace a través de este enlace. El límite entre el host y el enlace físico se denomina **interfaz**.

Un router tiene varias interfaces, una para cada uno de los enlaces. Puesto que todos los hosts y todos los routers son capaces de enviar y recibir datagramas IP, **IP requiere que cada interfaz de host y de router tenga su propia dirección IP.**

Las **direcciones IP tienen una longitud de 32 bits** (4 bytes), 2^32direcciones posibles. Estas direcciones se expresan utilizando la **notación decimal con punto**, en la que cada byte de la dirección se escribe en formato decimal y separada mediante un punto del resto de los bytes de la dirección.

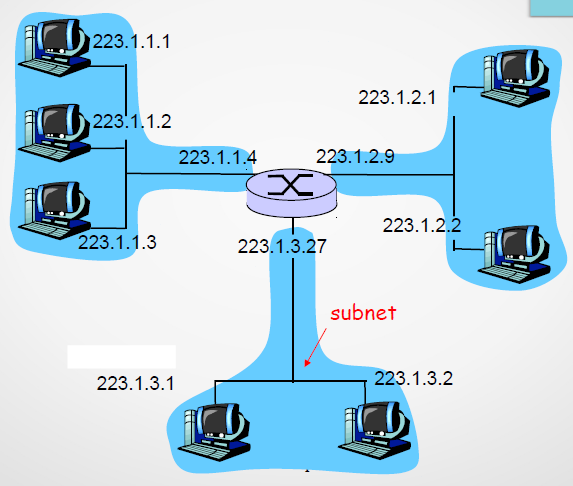
Por ejemplo, la dirección IP 193.32.216.9. El 193 es el numero decimal equivalente a los 8 primeros bits de la dirección; el 32 es el equivalente decimal de los segundos 8 bits de la dirección, y así sucesivamente. Por tanto, la dirección 193.32.216.9 en notación binaria se expresa como sigue:

11000001 00100000 11011000 00001001

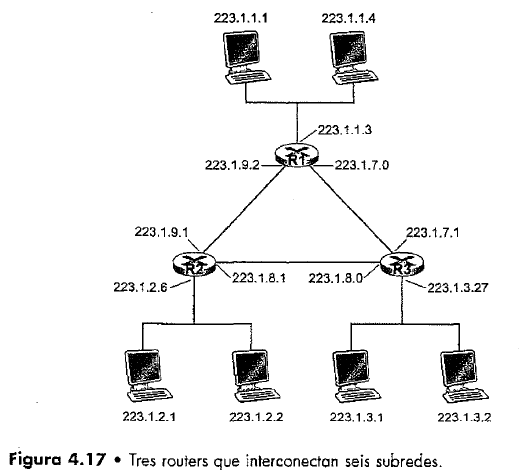
Cada una de las interfaces de un host o de un router de Internet tiene que tener asociada una dirección IP que es **globalmente única.** No obstante, estas direcciones no se pueden elegir al azar. **Una parte de la dirección IP de una interfaz estará determinada por la subred** a la que está conectada.

Por ejemplo, un router (3 interfaces) para interconectar siete hosts. Los tres hosts de la parte superior izquierda y la interfaz del router a la que están conectados, todos ellos tienen una dirección IP con el formato 223.1.1.xxx. Es decir, los 24 bits más a la izquierda de la dirección IP de todos ellos son iguales. En términos de IP, esta red que interconecta tres interfaces de host y una interfaz de router forma una **subred**.

El direccionamiento IP asigna una dirección a esta subred: 223.1.1.0/24, donde **la notación /24, que en ocasiones se denomina mascara de subred**, indica que los 24 bits más a la izquierda de la magnitud de 32 bits definen la dirección de subred. Cualquier host adicional conectado a la subred 223.1.1.0/24 requeriríauna dirección de la forma 223.1.1.xxx.

La definición IP de una subred no está restringida a los segmentos Ethernet que conectan varios hosts a una interfaz de un router.

Considerando la Figura 4.17, que muestra tres routers interconectados. Cada router tiene tres interfaces, una para cada enlace y una para el enlace de difusión que conecta directamente el router a una pareja de hosts.

 ¿Qué subredes hay presentes? Hay tres: 223.1.1.0/24, 223.1.2.0/24 y 223.1.3.0/24, y son similares a las subredes de la Figura 4.15. Pero fíjese en que, en este ejemplo, también existen tres subredes adicionales: una subred, 223.1.9.0/24, para las interfaces que conectan los routers RI y R2; otra subred, 223.1.8.0/24, para las interfaces que conectan los routers R2 y R3; y una tercera subred, 223.1.7.0/24, para las interfaces que conectan los routers R3 y R1.

En un sistema interconectado general de routers y hosts, podemos utilizar la siguiente receta para definir las subredes existentes en el sistema:

**Para determinar las subredes, desconecte cada interfaz de su host o router, creando islas de redes aisladas, con interfaces que acaban en los puntos terminales de las redes aisladas. Cada una de estas redes aisladas se dice que es una subred.**

La estrategia de asignación de direcciones en Internet se conoce como **Enrutamiento entre dominios sin clase (CIDR)**, CIDR generaliza la noción de direccionamiento de subred. La dirección **IP de 32 bits se divide en dos partes** y de nuevo se expresa en notación decimal con punto como *a.b.c.d/x*, **donde *x* indica el número de bits de la primera parte de la dirección.**

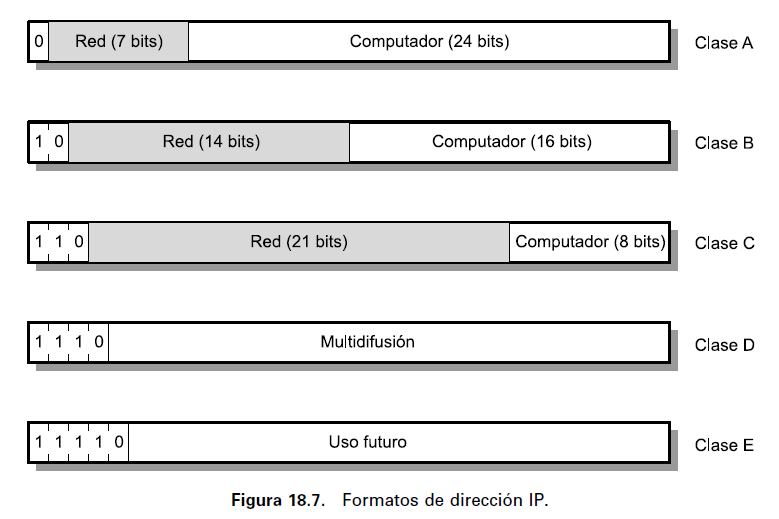
Los ***x* bits más significativos de una dirección en el formato *a.b.c.d/x* constituyen la parte de red** de la dirección IP (**prefijo** *de red* de la dirección).

Los routers externos a la red de la organización solo tienen en cuenta estos *x* primeros bits del prefijo cuando reenvía un datagrama cuya dirección de destino está dentro de la organización. Esto reduce el tamaño de la tabla de reenvió de los routers.

Los **32-x bits restantes de una dirección pueden emplearse para diferenciar los dispositivos internosde la organización**, teniendo todo el mismo prefijo de red. Estos son los bits que habrá que considerar para reenviar paquetes en los routers *internos* de la organización.

Estos bits de menor peso pueden tener (o no) una estructura en **subred adicional**, como la que hemos visto anteriormente. Por ejemplo, suponga que los 21 primeros bits de la dirección CIDR a.b.c.d/21 especifican el prefijo de red de la organización. Los restantes 11 bits identifican a los hosts de la organización. Estos 11 bits de más a la derecha se pueden emplear para dividir en subredes la organización. Por ejemplo, *a.b.c.d/24* podría hacer referencia a una subred.

Antes de que se adoptara el enrutamiento CIDR, la parte de red de una dirección IP estaba restringida a longitudes de **8, 16 o 24 bits**, un esquema de direccionamiento conocido como **direccionamiento con clases,** ya que las subredes con direcciones de 8, 16 y 24 bits se conocían, respectivamente, como redes de clase A, B y C.

**Clases**

**Clase A (/8)**

– Pocas redes, cada una con muchas computadoras

– Comienzan con el 0 (el 0=00000000 o el 127=01111111 se encuentran reservadas para subred y difusión)

– Rango desde 1.x.x.x hasta 126.x.x.x

**Clase B (/16)**

– Número medio de redes, número medio de computadoras

– Comienzan con el 10, total de direcciones de red 2˄14=14384

– Rango desde 128.x.x.x hasta 191.x.x.x

- soporte a 65.634 hosts (2˄16 - 2)

**Clase C (/24)**

– Muchas redes, cada una con pocas computadoras

– Comienza con 110, total de direcciones de red 2 ˄21.

– 192.x.x.x hasta 223.x.x.x

- Puede acomodar hasta 256 - 2 = 254 hosts (dos de las 2˄8 = 256 direcciones están reservadas la de los bits de todos ceros 00000000 es la subred y la de todos unos 11111111 es la de difusión).

La dirección IP de difusión 255.255.255.255. Cuando un host envía un datagrama cuya dirección de destino es 255.255.255.255, el mensaje se entrega a todos los hosts existentes en la misma subred.

No todas las direcciones de red están disponibles para su uso

– El rango de direcciones IPv4 es 0.0.0.0 a 255.255.255.255

– Direcciones experimentales: 240.0.0.0 a 255.255.255.254

– Direcciones multicast: 224.0.0.0 a 239.255.255.255

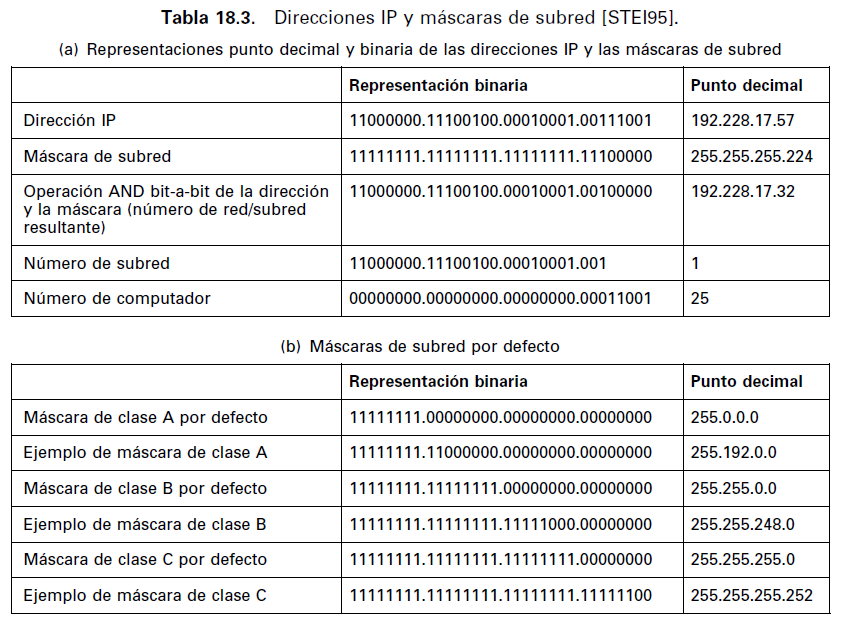
– Direcciones de enlace local: 169.254.1.0 – 169.254.254.255

– Direcciones IP privadas:

**10.0.0.0 a 10.255.255.255 (10.0.0.0/8)**

**172.16.0.0 a 172.31.255.255 (172.16.0.0/12)**

**192.168.0.0 a 192.168.255.255 (192.168.0.0/16)**



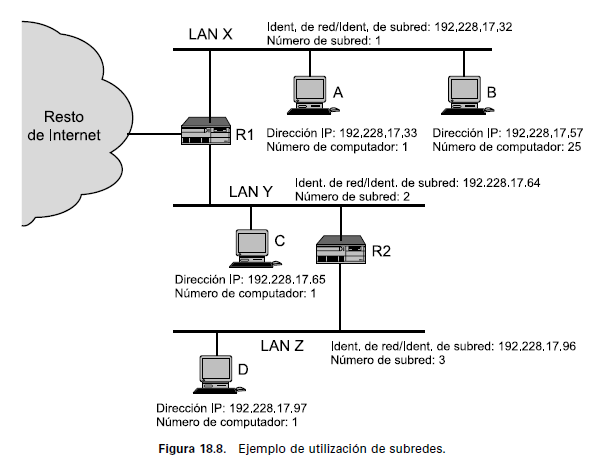
**Subredes y máscaras de subred**

Para permitir que los dispositivos de encaminamiento internos al sitio funcionen correctamente, a cada LAN se le asigna un número de subred. La dirección internet se divide en un número de subred y un número de computador para acomodar este nuevo nivel de direccionamiento.

Dentro de una red dividida en subredes, los dispositivos de encaminamiento locales deben encaminar sobre un número de red formado por la parte de *red* de la dirección IP y el número de subred. Las posiciones a nivel de bit que contienen este número de red se indican mediante **la máscara de dirección**. El uso de esta máscara de dirección permite a un computador determinar si un datagrama de salida va destinado a otro computador en la misma LAN– o a otra LAN (se envía a un dispositivo de encaminamiento).

La Tabla 18.3a muestra los cálculos que se realizan, la operación AND entre la dirección IP y la máscara de subred, con la utilización de una máscara de subred. El efecto de la máscara de subred es borrar la parte del campo de computadora que indica la computadora en una subred. Lo que permanece es el número de red y el número de subred. La máscara de subred por defecto para una clase de direcciones es una máscara nula (18.3b), produce el mismo número de red y de computador que en el caso de una dirección sin subredes.

Ejemplo de aplicación

Se muestra un complejo local consistente en tres LAN y dos dispositivos de encaminamiento. Para el resto del conjunto de redes este complejo es una red única con una dirección de clase C de la forma 192.228.17.*x*, donde los tres octetos más a la izquierda son el número de red y el octeto más a la derecha contiene un número de computador *x*. Ambos dispositivos de encaminamiento R1 y R2 se configuran con una máscara de subred con el valor 255.255.255.224 (v*éase* la Tabla 18.3a). Por ejemplo, si un datagrama con una dirección destino 192.228.17.57 llega a R1 desde el resto del conjunto de redes o desde la LAN Y, R1 aplica la máscara de subred para determinar que esta dirección hace referencia a una dirección de la subred 1, la cual es la LAN X, y si es así enviarlo a la LAN X. De forma similar, si llega un datagrama con esa dirección destino a R2 desde la LAN Z, R2 aplica la máscara y determina a partir de su base de datos que el datagrama destinado a la subred 1 se debe enviar a R1. Los computadores también utilizan la máscara de subred para tomar decisiones de encaminamiento.

**Ejemplo para IP 192.168.0.10/24**

Máscara 255.255.255.0

– Dirección de red 192.168.0.0

– Dirección de broadcast 192.168.0.255

– Identificador de computadora

IP: 11000000 10101000 00000000 000001010

Mascara: 11111111 11111111 11111111 00000000

**TRADUCCIÓN DE DIRECCIONES DE RED (NAT)**

Con el crecimiento de las redes de hogar u oficina, pareciera que cuando se desea instalar una LAN, el ISP debería asignar un rango de direcciones para cubrir todas sus máquinas. Si la subred creciera, el bloque de direcciones seria enorme, además de que el ISP puede ya haber asignado las direcciones adyacentes a su rango. Existe una forma más fácil de asignar direcciones mediante la **traducción de direcciones de red (NAT).**

Funcionamiento de un router NAT: Este router, se encuentra en una vivienda u oficina, tiene una interfaz que es parte de la red doméstica. El direccionamiento dentro de la red doméstica, las cuatro interfaces de la red tienen la misma dirección de subred 10.0.0/24. El espacio de direcciones 10.0.0.0/8 es una de las partes del espacio de direcciones IP que está reservado para una **red privada** o para un **ámbito** con direcciones privadas.

Un **ámbito** *con* direcciones **privadas**hace referencia a una red cuyas direcciones solo tienen significado para los dispositivos internos de dicha red, es decir solo dentro de la misma red. Los dispositivos de una red doméstica pueden enviarse paquetes entre si utilizando el direccionamiento 10.0.0.0/24. Sin embargo, los paquetes reenviados *hacia fuera* de la red doméstica, hacia Internet, no pueden usar estas direcciones, porque existen miles de redes que emplean ese bloque de direcciones.

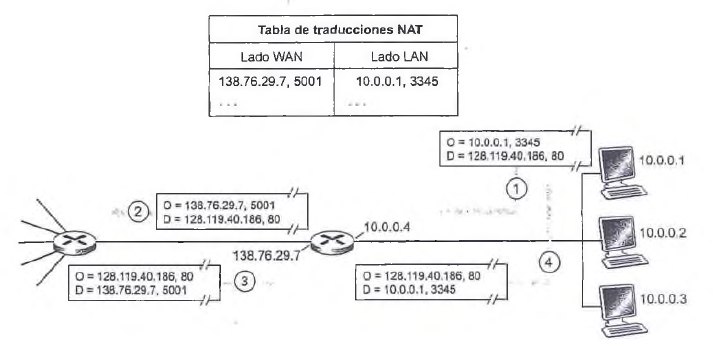
Es decir, las direcciones 10.0.0.0/24 solo tienen significado dentro de una red doméstica dada.

¿Cómo se direccionan los paquetes cuando se envían por internet o se reciben de internet?

Para direccionar los paquetes que se envían a Internet o se reciben de Internet, donde las direcciones tienen que ser únicas, el **router NAT** no figura comoun router hacia el exterior, sino que se comporta como **un único dispositivo con una dirección IP única***.* Todo el tráfico que sale del router domestico hacia Internet tiene una dirección IP de origen de la interfaz del router conectada hacia internet (138.76.29.7) y todo el tráfico que entra en el tienen que tener la misma dirección para destino (138.76.29.7). El router NAT oculta los detalles de la red doméstica al mundo exterior.

El router obtiene su dirección del servidor DHCP del ISP y el router ejecuta un servidor DHCP para proporcionar direcciones a las computadoras, dentro del espacio de direcciones de la red doméstica controlada por el router NAT-DHCP.

Todos los datagramas que llegan al router NAT procedentes de la WAN tienen la misma dirección IP de destino (la de la interfaz WAN del router NAT), el router para saber a qué host interno debería reenviar un datagrama dado usa una **tabla de traducciones NAT** almacenada en el router NAT, e incluye los números de puerto, así como las direcciones IP en las entradas de la tabla.

Ejemplo, si el host 10.0.0.1 solicita una página web en un servidor web (puerto 80) con la dirección IP 128.119.40.186. El host origen (10.0.0.1) asigna el número de puerto de origen (3345) y envía el datagrama a la LAN. El router NAT recibe el datagrama, genera un nuevo número de puerto de origen (5001) para el datagrama, sustituye la dirección IP de origen por su dirección IP de la red WAN (138.76.29.7) y sustituye el número de puerto de origen original (3345) por el nuevo número de puerto de origen (5001).

El campo número de puerto es 16 bits, el protocolo NAT puede dar soporte a 60.000 conexiones simultaneas utilizando la única dirección IP WAN del router.

En el router NAT añade **una entrada a su tabla de traducciones**. El servidor web responde con un datagrama cuya dirección de destino es la dirección IP del router NAT y cuyo número de puerto de destino es el que generó el router (5001). Cuando este datagrama llega al router NAT, este indexa la tabla de traducciones NAT utilizando la dirección IP de destino y el número de puerto de destino para obtener la dirección IP (10.0.0.1) y el número de puerto de destino (3345) apropiados para el host de la red doméstica. A continuación, el router reescribe la dirección de destino y el número de puerto de destino del datagrama y lo reenvía a la red doméstica.

¿Por qué no NAT?

-Los números de puerto deben emplearse para direccionar procesos, no para direccionar host.

-Los routers están pensados para procesar paquetes solo hasta la capa 3.

-El protocolo NAT viola lo que se denomina enfoque terminal a terminal, es decir, que los hosts deben comunicarse directamente entre sí, sin que los nodos intermedios modifiquen las direcciones IP y los números de puerto.

-Se debería migrar a IPv6.

Un problema de NAT es que **interfiere con las aplicaciones P2P**, en una aplicación P2P, cualquier par A participante debería poder iniciar una conexión TCP con cualquier otro par B participante. El problema es que si el par B está situado detrás de un traductor NAT no puede actuar como un servidor y aceptar conexiones TCP. Este problema de NAT puede resolverse si el par A no está situado detrás de un traductor NAT. En este caso, el par A puede contactar primero al par B a través de un par C intermedio, que no esté situado detrás de un traductor NAT y con el que B tenga establecida una conexión TCP activa. El par A puede entonces pedir al par B, a través de C, que inicie una conexión TCP directamente con el par A. Una vez que la conexión TCP directa P2P se ha establecido entre los pares A y B, estos podrán intercambiar mensajes o archivos. Esta técnica, conocida con el nombre de **inversión de la conexión** es utilizada por muchas aplicaciones P2P para **NAT transversal (NAT** *Traversal).*

**UNIVERSAL PLUG AND PLAY (UPnP)**

El mecanismo de NAT transversal es proporcionado por Universal Plug and Play (UPnP), que es un protocolo que permite a un host descubrir y configurar un traductor NAT próximo. **UPnP requiere que tanto el host como el traductor NAT sean compatibles con UPnP.**

Con UPnP, una aplicación que se ejecuta en un host puede solicitar una correspondencia NAT entre su tupla (*dirección IP privada, número* *de puerto privado)* y la tupla (*dirección IP pública, número de puerto público)* para algún número de puerto público solicitado. Si el traductor NAT acepta la solicitud y crea la correspondencia, entonces los nodos del exterior pueden iniciar conexiones TCP con (*dirección* *IP privada, número de puerto privado).* Además, UPnP permite a la aplicación conocer el valor de (*dirección IP pública, número de puerto público),* de manera que la aplicación pueda anunciarse al mundo exterior.

Por ejemplo, suponga que su host, situado detrás de un traductor NAT compatible con UPnP, tiene la dirección privada 10.0.0.1 y está ejecutando una app (BitTorrent) en el puerto 3345. Suponga también que la dirección IP pública del traductor NAT es 138.76.29.7. Naturalmente, su aplicación BitTorrent desea poder aceptar conexiones de otros hosts, para poder intercambiar fragmentos con ellos. En esta situación, la aplicación BitTorrent de su host pide al traductor NAT que cree un “túnel” que asigne (10.0.0.1, 3345) a (138.76.29.7, 5001). (La aplicación elige el número de puerto público 5001.) La aplicación BitTorrent de su host podría también anunciar a su tracker que está disponible en (138.76.29.7, 5001). De esta manera, un host externo que este ejecutando BitTorrent puede contactar con el tracker y determinar que su aplicación BitTorrent está ejecutándose en (138.76.29.7, 5001). El host externo puede enviar un paquete SYN TCP a (138.76.29.7, 5001). Cuando el traductor NAT recibe el paquete SYN, cambiara la dirección IP de destino y el número de puerto del paquete a (10.0.0.1, 3345) y lo reenviara a través de NAT.

En resumen, UPnP permite a hosts externos iniciar sesiones de comunicación con hosts conectados a través de un traductor NAT utilizando TCP o UDP. UPnP proporciona una solución efectiva y robusta de NAT transversal.

**Como se obtienen y distribuyen los bloques de direcciones**

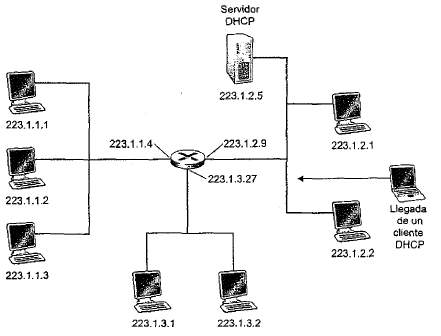
Para obtener un bloque de direcciones IP que pueda ser utilizado dentro de la subred de una organización, un administrador de red tiene que contactar con su ISP, el cual le proporcionara direcciones extraídas de un bloque de direcciones mayor que ya habrá sido asignado al ISP. Por ejemplo, al ISP puede tener asignado el bloque de direcciones 200.23.16.0/20. A su vez, el ISP podría dividir el bloque de direcciones en ocho bloques de direcciones contiguos del mismo tamaño y asignar cada uno de estos bloques de direcciones a hasta ocho organizaciones a las que puede prestar servicio.

Existe una entidad autoritativa global que gestiona el espacio de direcciones IP y asigna bloques de direcciones a los ISP y otras organizaciones, esta es la **ICANN** (Corporación de Internet para los números y nombres asignados). Su fin no es solo el de asignar direcciones IP, sino también gestionar los servidores raíz DNS, asignar nombres de dominio. La organización ICANN asigna direcciones a los registros regionales de Internet (por ejemplo, ARIN, RIPE, APNIC y LACNIC, que forman la Organización de Soporte de Direcciones de ICANN [ASO-ICANN 2009]), y gestionan la asignación/administración de direcciones dentro de sus regiones.

**OBTENER UNA DIRECCIÓN DE HOST: PROTOCOLO DE CONFIGURACIÓN DINÁMICA DE HOST (DHCP)**

Luego de obtener un bloque de direcciones, normalmente, un administrador configura manualmente las direcciones IP de un router, también los hosts se pueden configurar manualmente, pero esta tarea se lleva cabo utilizando el **Protocolo de configuración dinámica de host (DHCP)**.

DHCP permite a un host obtener automáticamente una dirección IP. Un administrador de red puede configurar DHCP para que un host reciba la misma dirección IP cada vez que se conecte a la red, o que se le asigne una dirección **IP temporal** diferente cada vez que se conecte a la red. Además, DHCP permite que un host obtenga información como su máscara de subred y la dirección del router del primer salto [Gateway o puerta de enlace] y la dirección de su servidor DNS local.

Es un protocolo **plug-and-play**. A medida que los hosts se unen a la red y salen de ella, el servidor DHCP actualiza su lista de direcciones IP disponibles. Cada vez que un host se une a la red, el servidor DHCP asigna una dirección de su conjunto actual de direcciones disponibles; cada vez que un host abandona la red, su dirección es devuelta al conjunto.

**DHCP es un protocolo cliente-servidor**. Un cliente es un host recién llegado que desea obtener información de configuración de la red. En el caso más simple, cada subred tendrá un servidor DHCP. Si en la subred no hay ningún servidor, es necesario un agente de retransmisión DHCP (ej router) que conozca la dirección de un servidor DHCP para dicha red.

Ej, servidor DHCP conectado a la subred 223.1.2/24, con el router actuando como agente de retransmisión para los clientes recién llegados que se conectan a las subredes 223.1.17/24 y 223.1.3/24.

Para un host recién llegado a una red, el protocolo DHCP es un proceso de cuatro pasos. Sudirl (“su dirección Internet”) indica la dirección que se asigna al cliente que acaba de llegar.

**Los cuatro pasos son los siguientes:**

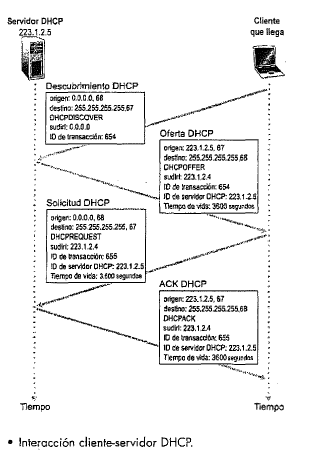
**• Descubrimiento del servidor DHCP.**Un host recién llegado debe encontrar un servidor DHCP, esto lo hace con un mensaje de descubrimiento DHCP, que envía un cliente dentro de un paquete UDP al puerto 67. El paquete UDP se encapsula en un datagrama IP. El cliente DHCP crea el datagrama IP que contiene su mensaje de descubrimiento DHCP junto con la dirección IP de difusión 255.255,255.255 y una dirección IP de origen de “este host” igual a 0.0.0.0. El cliente DHCP pasa el datagrama IP a la capa de enlace, la cual difunde esta trama a todos los nodos conectados a la subred.

**• Oferta(s) del servidor DHCP.** Un servidor DHCP que recibe un mensaje de descubrimiento DHCP responde al cliente con un mensaje de oferta DHCP, que se difunde a todos los nodos de la subred utilizando la dirección IP de difusión 255.255.255.255, esta respuesta se difunde porque en la subred pueden existir varios servidores DHCP, el cliente puede elegir entre varias ofertas. Cada mensaje de oferta de servidor contiene el ID de transacción del mensaje de descubrimiento recibido, la dirección IP propuesta para el cliente, la máscara de red y el tiempo de vida de la dirección IP.

**• Solicitud DHCP.** El cliente elige entre las ofertas de servidor y responderá a la oferta seleccionada con un mensaje de solicitud DHCP, devolviendo los parámetros de configuración.

**• ACK DHCP.** El servidor contesta al mensaje de solicitud DHCP con un mensaje ACK DHCP, que confirma los parámetros solicitados.

Una vez que el cliente recibe el mensaje de reconocimiento ACK DHCP, la interacción se completa y el cliente puede utilizar la dirección IP asignada por DHCP durante el tiempo acordado.



**PROTOCOLO DE MENSAJES DE CONTROL DE INTERNET (ICMP)**

ICMP proporciona un medio para transferir mensajes desde los dispositivos de encaminamiento y otros computadores a un computador. ICMP proporciona información de realimentación sobre problemas del entorno de la comunicación:

* cuando un datagrama no puede alcanzar su destino.
* cuando el dispositivo de encaminamiento no tiene la capacidad de almacenar temporalmente para reenviar el datagrama
* cuando el dispositivo de encaminamiento indica a una estación que envíe el tráfico por una ruta más corta.

En la mayoría de los casos, el mensaje ICMP se envía en respuesta a un datagrama. ICMP se encuentra encima de IP, ya que los mensajes ICMP son transportados dentro de los datagramas IP por lo tanto no se garantiza su entrega y su uso no se puede considerar fiable. Todos los mensajes ICMP empiezan con **una cabecera de 64 bits** que consta de los siguientes campos:

* **Tipo (8 bits):** especifica el tipo de mensaje ICMP.
* **Código (8 bits):** especifica parámetros del mensaje.
* **Suma de comprobación (16 bits):** suma de comprobación del mensaje ICMP.
* **Parámetros (32 bits):** se usa para especificar parámetros más largos.

**Tipos de mensajes ICMP:**

El mensaje **destino inalcanzable**, situaciones:

Un dispositivo de encaminamiento puede devolver este mensaje si no sabe cómo alcanzar la red destino.

* Un router conectado a una de estas redes puede ser capaz de determinar si un computador es inalcanzable y devolver este tipo de mensaje.
* El propio computador de destino puede devolver este mensaje si el protocolo de usuario o algún punto de acceso al servicio de un nivel superior no están alcanzables.
* Si el datagrama especifica una ruta dada por la fuente que no se puede usar, se devolverá un mensaje.
* Si un dispositivo de encaminamiento debe fragmentar un datagrama, pero el indicador de no fragmentación está establecido, se devuelve también el mensaje «destino inalcanzable».

Un dispositivo de encaminamiento devolverá el mensaje **tiempo excedido** si el tiempo de vida del datagrama ha expirado. Un computador enviará este mensaje si no puede completar el reensamblado dentro del margen de tiempo disponible.

El mensaje de **ralentización del origen** proporciona una forma de control de flujo. Este mensaje lo pueden enviar tanto un router como un computador destino a un computador origen solicitando que reduzca la tasa a la que envía el tráfico al destino. Cuando se recibe este tipo de mensaje, un computador origen debe disminuir la tasa de datos a la que envía el tráfico al destino especificado hasta que no reciba más mensajes de ralentización del origen. El mensaje de ralentización del origen lo puede generar tanto un dispositivo de encaminamiento como un computador que deba descartar datagramas debido a que su memoria temporal está llena. En este caso, el dispositivo de encaminamiento o el computador enviará un mensaje de ralentización del origen por cada datagrama que se descarta. Además, un sistema se puede anticipar a la congestión y enviar este tipo de mensaje cuando su memoria esté a punto de llegar a su capacidad máxima.

Un dispositivo de encaminamiento envía un mensaje de **redirección** a un computador conectado directamente a un dispositivo de encaminamiento para informarle de una ruta mejor para un destino particular.

Los mensajes **eco** y **respuesta a eco** proporcionan un mecanismo para comprobar que la comunicación entre dos entidades es posible. El receptor de un mensaje de eco está obligado a devolver el mensaje en un mensaje de respuesta a eco. Al mensaje de eco se le asocia un identificador y un número de secuencia que coinciden con los de paquete de respuesta a eco.

**DIRECCIONAMIENTO IPv6**

**Motivaciones para desarrollar IPv6**

* IPv4 no fue dimensionado para el crecimiento que tuvo Internet, el espacio de direcciones IP de 32 bits estaba comenzando a agotarse, a causa de las nuevas subredes y nodos IP que estaban conectándose a Internet
* El remanente del espacio de direcciones IPv4 desciende por debajo del 5%
* El proyecto arranco IPng en el año 1992 con un solo objetivo:

– Mas Direcciones! – Sin Identificación QoS

– Seguridad opcional – Autoconfiguración

**Motivaciones para desplegar IPv6**

* Para miles de millones de nuevos dispositivos, como teléfonos celulares, PDAs, dispositivos de consumo, coches, etc.
* Para tecnologías de acceso “always-on”, como xDSL, cable, ethernet, etc.
* Para miles de millones de nuevos usuarios, como China, India, etc.

– Espacio ampliado de direcciones 128 bits, lo que da 2^128 direcciones, 6x10^23 direcciones /m2 de la superficie de la tierra.

– 340,282,366,920,938,463,463,374,607,431,768,211,456

– Con asignación jerárquica pesimista: 1.546 direcciones/m2

– 3,911,873,538,269,508,102 direcciones/m2

* **¿Por qué no NAT?**

– No funciona con gran número de “servidores”, es decir, dispositivos que son “llamados” por otros (Teléfonos IP)

– Comprometen las prestaciones, robustez, seguridad y manejabilidad de Internet

– Rompe el principio e2e para las comunicaciones

* Facilidad para la gestión/delegación de las direcciones
* Espacio para más niveles de jerarquía y para la agregación de rutas
* Habilidad para las comunicaciones extremo-a-extremo con IPsec (porque no necesitamos NATs)
* Capacidades expandidas de direccionamiento
* Simplificación del formato de la cabecera
* Soporte mejorado de extensiones y opciones
* Capacidad de etiquetado de flujos
* Capacidades de autenticación y encriptación

**IPv6 incluye las siguientes mejoras sobre IPv4:**

**Un espacio de direcciones ampliado:** IPv6 utiliza direcciones de 128 bits en lugar de las direcciones de 32 bits de IPv4.

**Un mecanismo de opciones mejorado:** las opciones de IPv6 se encuentran en cabeceras opcionales separadas situadas entre la cabecera IPv6 y la cabecera de la capa de transporte. Esto simplifica y acelera el procesamiento que realiza un router.

**Autoconfiguración de direcciones:** esta capacidad proporciona una asignación dinámica de direcciones IPv6.

**Aumento de la flexibilidad en el direccionamiento:** IPv6 incluye el concepto de una dirección monodifusión, mediante la cual un paquete se entrega sólo a un nodo seleccionado de entre un conjunto de nodos.

**Funcionalidad para la asignación de recursos:** IPv6 habilita el etiquetado de los paquetes como pertenecientes a un flujo de tráfico particular para el que el emisor solicita un tratamiento especial.

**Estructura IPv6: IPv4 vs IPv6**

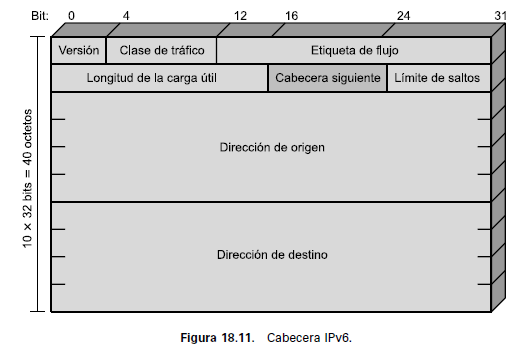
Una unidad de datos del protocolo IPv6 tiene el formato general siguiente:



La única cabecera que se requiere se denomina **cabecera IPv6**. Esta tiene una longitud fija de 40 octetos. Se han definido las siguientes cabeceras de extensión:

* **Opciones salto a salto:** define opciones especiales que requieren procesamiento en cada salto
* **Encaminamiento:** lista los nodos intermedios por los que pasa el paquete.
* **Fragmentación:** información de fragmentación y ensamblado.
* **Autenticación:** valida el mensaje y asegura la integridad de los datos.
* **De encapsulamiento de la carga de seguridad**
* **De las opciones para el destino:** ofrece confidencialidad y protege contra la interceptación.



**Cabecera de IPv6:** Tiene una longitud de 40 bytes con los siguientes campos:

**Versión (4 bits):** número de la versión del protocolo Internet; el valor es 6.

**Clase de tráfico (8 bits):** identificar y distinguir entre clases o prioridades de paquete IPv6. Este

**Etiqueta de flujo (20 bits):** etiquetar aquellos paquetes para los que requiere un tratamiento especial en los routers

**Longitud de la carga útil (16 bits):** longitud del resto del paquete IPv6 excluida la cabecera.

**Cabecera siguiente (8 bits):** identifica el tipo de cabecera que sigue inmediatamente a la cabecera IPv6

**Límite de saltos (8 bits):** el número restante de saltos permitidos para este paquete. Se decrementa en 1 en cada nodo que reenvía el paquete.

**Dirección origen (128 bits):** dirección del productor del paquete.

**Dirección destino (128 bits):** dirección de destino deseado del paquete.

**Formato del datagrama IPv6:**

Los cambios más importantes introducidos en IPv6 son evidentes en el formato de su datagrama:

* **Capacidades ampliadas de direccionamiento.** La dirección paso de 32 a 128 bits.
* **Una cabecera de 40 bytes simplificada.** algunos de los campos de IPv4 se han eliminado, longitud fija de 40 bytes.
* **El campo de fragmentación se retiró**, ya que solo se permite fragmentación y reensamblado en el origen y el destino.
* **Suma de comprobación fue retirado de la cabecera** ya que es ofrecida por protocolos de nivel superior
* **El campo de opciones se eliminó de la cabecera**
* El campo de la **longitud de la cabecera** se eliminó ya que la longitud de la cabecera ahora es fija.
* El campo **protocolo se reemplazó por el campo cabecera siguiente**.
* Se agregaron **cabeceras de extensión** para darle más funcionalidad al datagrama.
* Se reemplazó **el tiempo de vida del datagrama por un límite de saltos.**

**Formato de Direcciones IPv6**

* **8 campos de 16 bits de largo**
* **Separados por dos puntos “:”**
* **Cada campo está representado por 4 caracteres hexadecimales (0-f)**
* **No son sensibles a mayúsculas o minúsculas. Ej:**

**– 2001:0000:1234: 0000:0000:C1C0: ABCD: 0876**

**– 2001:0000:1234: 0000:0000:c1c0: abcd: 0876**

**– 2001:0:1234: 0:0:c1c0: abcd: 876**

**– 2001:0:1234:c1c0: abcd: 876**

**IPv6 permite tres tipos de direcciones:**

**Unidifusión:** un identificador para una interfaz individual. Un paquete enviado se entrega a la interfaz con esta dirección.

**Monodifusión (*anycast*):** un identificador para un conjunto de interfaces, un paquete enviado a una dirección monodifusión se entrega a una de las interfaces identificadas por esa dirección

**Multidifusión (*multicast*):** un identificador para un conjunto de interfaces. Un paquete enviado a una dirección multidifusión se entrega a todas las interfaces identificadas por esa dirección.

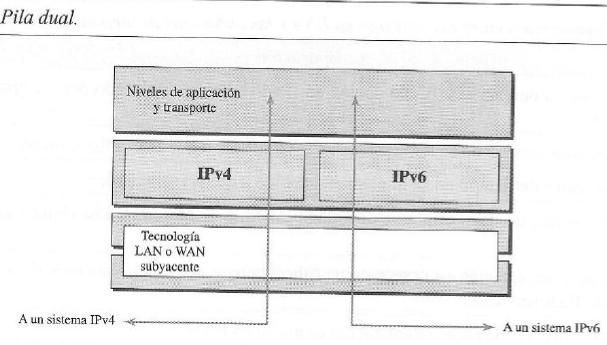
**TRANSICIÓN DE IPV4 A IPV6**

La transición no puede ocurrir de repente, debe ser suave para evitar cualquier problema, se han desarrollado 3 estrategias:

* Pila dual.
* Túneles.
* Traducción de cabeceras.

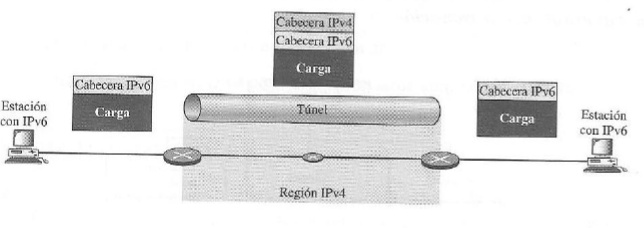
**Pila dual**

Antes de migrar del todo a la versión 6, cada estación debería tener una pila dual de protocolos. Cada estación que ejecuta Ipv6 debe ejecutar simultáneamente IPv4 e IPv6 hasta que se utilice IPv6, así podrá usar los datagramas de ambas versiones. Para determinar que versión usar cuando se envía un paquete a un destino, el emisor consulta el DNS. Si el DNS devuelve una dirección IPv4, el emisor envía un paquete IPv4, si devuelve una dirección IPv6, envía un paquete IPv6.



**Túneles**

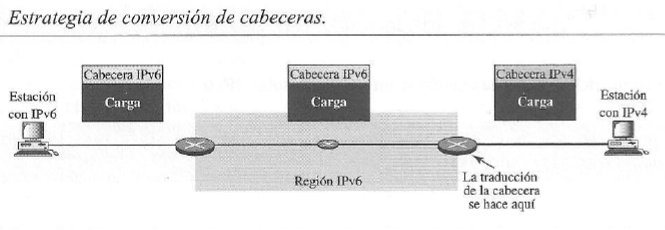
Se usa cuando 2 computadoras que usan IPv6 tienen que comunicarse entre sí y los paquetes deben atravesar una región donde los routers que usa son IPv4. Para pasar por esta región, el paquete debe tener una dirección IPv4. Por lo tanto, el paquete IPv6 se encapsula en un paquete IPv4 cuando entra a la región y se extrae cuando la deja. Para indicar que el paquete IPv4 transporta un paquete IPv6, el valor del protocolo se fija a 41.

El nodo IPv6 del lado emisor del túnel toma el datagrama IPv6 y lo incluye en el campo de datos de un datagrama IPv4. Este datagrama IPv4 se direcciona entonces hacia el nodo IPv6 del lado receptor del túnel y se envía al primer nodo del túnel. Los routers IPv4 del túnel enrutan este datagrama IPv4 sin saber que el datagrama IPv4 contiene un datagrama IPv6. El nodo IPv6 del lado de recepción del túnel termina recibiendo el datagrama IPv4 determina que el datagrama IPv4 contiene un datagrama IPv6, extrae el datagrama IPv6.

Traducción de cabeceras

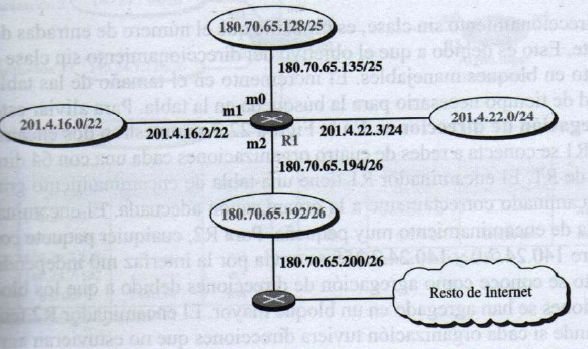
La traducción de cabeceras es necesaria cuando la mayor pare de internet se ha movido a IPv6, pero algunos sistemas siguen con IPv4. El emisor quiere usar IPv6, pero el receptor no comprende IPv6. En este caso el formato de la cabecera debe cambiarse mediante un proceso de traducción. La cabecea del paquete IPv6 se convierte a una cabecera de IPv4. Reglas de traducción:

* Una dirección de IPv6 se pasa a Ipv4 extrayendo los 32 bits de la derecha.
* Se descarta el campo prioridad de IPv6.
* El tipo de servicio en IPv4 se pone a cero.
* Se calcula e inserta la suma de comprobación de IPv4.
* Se ignora la etiqueta de flujo de IPv6.
* Las cabeceras de extensión se convierten en opciones de la cabecera de IPv4.
* Se calcula e inserta la longitud de la cabecera de IPv4.
* Se calcula e inserta la longitud en el campo.



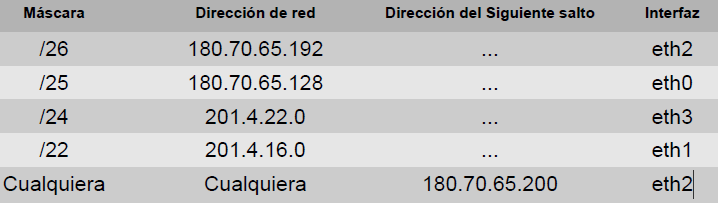
**Filmina N°8 algoritmos de enrutamiento**

**Tablas de enrutamiento**

En la función de **reenvío** de un router, cuando recibe un paquete por un enlace de entrada este tiene que reenviar el paquete por el enlace de salida apropiado. Esta función requiere que cada router mantenga una tabla de enrutamiento. Cuando se recibe un paquete el router busca en esta tabla para encontrar el camino al destino final.

Para el proceso de reenvío, la tabla de enrutamiento necesita tener una fila de información para cada enlace involucrado. En la tabla se busca de acuerdo a la dirección de red (IP) y la máscara. Ejemplo de tabla.

Tabla para R1



Por ejemplo, si llega un paquete de destino 180.70.65.140 a R1 se realiza lo siguiente:

1. Se aplica la máscara /26 a la dirección de destino, como resultado se obtiene 180.70.65.128 que no coincide con la dirección de red correspondiente.
2. Se aplica la segunda mascara /25 y el resultado es 180.70.65.128 que coincide con la dirección de red. La **dirección del siguiente salto** y el número de interfaz eth0 se pasan a ARP para su procesamiento.

Las tablas de enrutamiento se mantienen en cada host o router, tienen una entrada para cada destino, para encaminar los paquetes IP. Esta tabla puede ser estática o dinámica. Las tablas estáticas son cargadas a mano, no se pueden actualizar automáticamente. Una tabla dinámica se actualiza en forma periódica usando un protocolo de enrutamiento como RIP, OSPF o BGP. Cuando hay un cambio los protocolos actualizan todas las tablas de los routers de manera automática.

Los principales elementos de una tabla de enrutamiento son la máscara, dirección de red, dirección del siguiente salto, interfaz del enlace.

**Utilidades para información de las tablas de enrutamiento**

* **Netstat –nr:** da el contenido de la tabla de enrutamiento, r indica que es la tabla de enrutamiento.
* **Ipconfig eth0:** da información específica de la interfaz sobre la dirección IP y la dirección física.

**Algoritmos de enrutamiento**

La capa de red tiene que determinar la ruta que seguirán los paquetes desde los emisores a los receptores. Todo host está conectado directamente a un router, el **router predeterminado** para el host (**router del primer salto**). Cuando unhost envía un paquete, este se transfiere a su router predeterminado. El routerpredeterminado del host de origen como **router de origen** y al router predeterminado delhost de destino como **router de destino.**  El **problema de enrutar** un paquete es enrutar el paquete desdeel router de origen al router de destino.

El propósito de un algoritmo de enrutamiento es**: determinar una “buena” ruta desde el router de origen al router de destino**. Es decir, la del **mínimo costo**. No es simple debido a problemas del mundo real, políticas utilizadas, etc.

Para formular los problemas de enrutamiento se usan **grafos**. Un grafo G= (N, E) es un conjunto N de nodos y una colección E de aristas, donde cada arista es una pareja de nodos de N.

Los nodos son routers y las aristas enlaces físicos entre los routers. Una arista tiene un valor que es su costo. Para cualquier arista (*x, y*)se define *c {x,y)* como el coste de la arista entre los nodos *x* e *y.* Si el par (x, y) no pertenece al grafo*,* c(x, y) =∞. El objetivo de un algoritmo de enrutamiento es identificar las rutas de coste mínimo entre los orígenes y los destinos. El coste de una ruta (x1, ..., *xp)* es la suma del coste de todas las aristas a lo largo de la ruta; es decir, c(x1, x2) + c(x2, x3) +...+ *c (xp-1, xp)*.

El costo asignado depende de cada algoritmo de enrutamiento, en RIP (protocolo de información de encaminamiento) trata a todas las redes por igual, el costo de pasar a través de una red es un salto. El protocolo abierto de primer camino más corto (OSPF) permite al administrador asignar un costo para pasar a través de una red. El protocolo BGP (protocolo de pasarela frontera) el costo es la política que puede ser fijada por el administrador.

**Una forma de clasificar los algoritmos de enrutamiento es:**

• **Un algoritmo de enrutamiento global** calcula la ruta de costo mínimo entre origen y destino usando el conocimiento global y completo acerca de la red. Es decir, el algoritmo toma como entradas la conectividad entre todos los nodos y todos los costes de enlace. Esto requiere que el algoritmo obtenga esta información antes de realizar el cálculo. Un algoritmo global dispone de toda la información acerca de la conectividad y de los costes de los enlaces. En la práctica, los algoritmos con información de estado global a menudo se denominan algoritmos de estado de enlaces (LS*),* ya que el algoritmo tiene que ser consciente del coste de cada enlace de la red.

• **En un algoritmo de enrutamiento descentralizado**, el cálculo de la ruta de coste mínimo se realiza de manera iterativa y distribuida. Ningún nodo tiene toda la información acerca del coste de todos los enlaces de la red. Al principio, cada nodo solo conoce los costes de sus propios enlaces. Después, a través de un proceso iterativo de cálculo e intercambio de información con sus nodos vecinos (es decir, los nodos que están en el otro extremo de los enlaces a los que el mismo está conectado), cada nodo calcula gradualmente la ruta de coste mínimo hacia un destino o conjunto de destinos. Un ejemplo es el algoritmo vector de distancias (DV), cada nodo mantiene un vector de estimaciones de los costes a todos los demás nodos de la red.

Una segunda forma general de clasificar los algoritmos de enrutamiento es en:

• **Estáticos:** las rutas cambian muy lentamente con el tiempo, con frecuencia como resultado de una intervención humana (por ejemplo, una persona que edita manualmente la tabla de reenvió de un router).

– Información previamente recopilada

– En función de la capacidad de la línea, tráfico promedio, etc.

– Tablas cargadas por el administrador estáticamente en cada router

– No es posible responder a situaciones cambiantes

– Utilizada en pequeñas redes

• **Dinámico:** modifican las rutas de enrutamiento a medida que la carga de tráfico o la topología de la red cambian. Puede ejecutarse periódicamente o como respuesta directa a cambios en la topología o en el coste de los enlaces. Responden mejor a los cambios de la red, son más susceptibles a problemas como los bucles de enrutamiento.

– Información recopilada en tiempo real.

– En función de la información actual de la red recibida de otros routers.

– Mecanismo autoadaptativo.

– Soporte a fallos (Caída de un enlace o de un router).

Una tercera forma de clasificar los algoritmos de enrutamiento es según sean **sensibles o no a la carga**. En un algoritmo sensible a la carga, los costes de enlace varían de forma dinámica para reflejar el nivel actual de congestión en el enlace subyacente. Si se asocia un coste alto con un enlace que actualmente este congestionado, el algoritmo de enrutamiento tendera a elegir rutas que eviten tal enlace congestionado. Los algoritmos de enrutamiento actuales de Internet (RIP, OSPF y BGP) no son sensibles a la carga, ya que el coste de un enlace no refleja su nivel actual de congestión.

**Encaminamiento interdominio e intradominio**

Una internet puede ser tan grande que un protocolo de enrutamiento no puede actualizar todas las tablas de los routers. Por eso una internet se divide en **sistemas autónomos. Un sistema autónomo:**

* Es un grupo de redes y routers controlados por una única administración.
* El encaminamiento dentro de un AS se denomina **encaminamiento intradominio (IGP).** Puede haber más de un protocolo en un AS para manejarlo.
* El encaminamiento entre sistemas autónomos se denomina **encaminamiento Inter dominio (EGP).** Solo un protocolo de encaminamiento lo maneja.

****Protocolos de intradominio:

* Basado en el vector de distancia **protocolo de información de encaminamiento o RIP.**
* Basado en el estado de enlace **protocolo abierto del primer camino más corto u OSPF.**

Protocolo de Inter dominio:

* Basado en el vector camino, **protocolo de pasarela frontera o BGP.**

**Algoritmo de enrutamiento de estado de enlaces (LS)**

**La topología de la red y el coste de todos los enlaces son conocidos**, están disponibles como entradas para el algoritmo LS, cada nodo puede usar el algoritmo de Dijkstra. Esto se consigue haciendo que cada nodo difunda paquetes del estado de los enlaces a *todos* los demás nodos de la red, con cada paquete de estado de enlace conteniendo las identidades y los costes de sus enlaces conectados. El resultado es que todos los nodos tienen una visión completa e idéntica de la red. Cada nodo usa la misma topología para crear la tabla de encaminamiento, pero esta es única para cada nodo. **Cada nodo puede entonces ejecutar el algoritmo LS** y **calcular el mismo conjunto de rutas de coste mínimo que cualquier otro nodo**. La topología debe ser dinámica, representando el último estado de cada nodo y cada enlace. Si hay cambios en cualquier punto de la red la topología debe actualizarse en cada nodo.

Construcción de las tablas de encaminamiento:

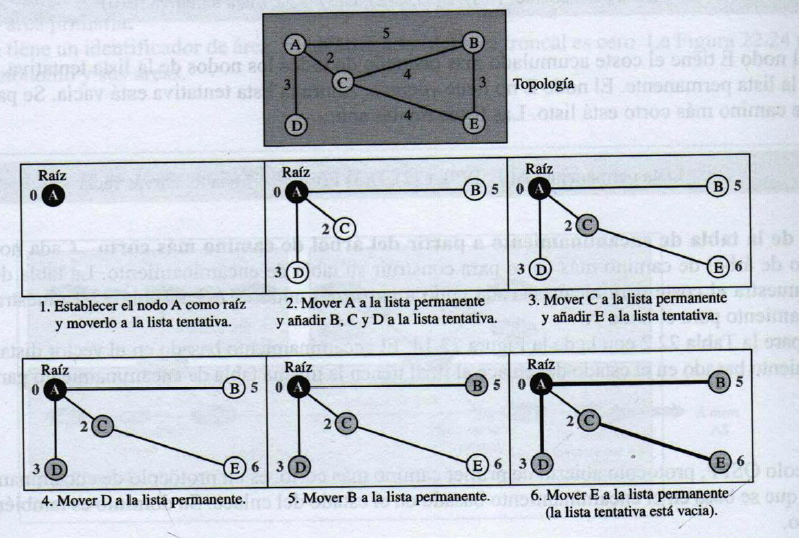
Se usan 4 acciones para que cada nodo tenga la tabla de encaminamiento que muestra al nodo de menor costo al resto de nodos:

1. Creación de los estados de los enlaces por cada nodo en un paquete de estado del enlace (LSP).
2. Distribución de los paquetes de estado de enlaces, denominado inundación.
3. Formación del árbol de camino más corto para cada nodo (cálculo de la ruta mínima)
4. Calculo de una tabla de encaminamiento basado en el árbol de camino más corto.

**Creación del paquete de estado del enlace:** contiene la identidad del nodo, lista de enlaces, número de secuencia y la edad. Estos paquetes se generan en dos ocasiones, cuando hay un cambio en a la topología o de forma periódica.

**Inundación de paquetes LSP:** se disemina un paquete LSP a todos los nodos no solo a los vecinos. El nodo que creo el paquete lo envía por todas las interfaces de salida, cada nodo que lo reciba compara con la copia que pueda tener y si es más nuevo actualiza el suyo y lo difunde por todos los enlaces excepto por el que llego.

**Formación del árbol de camino más corto:** este es un árbol en el que el camino entre el nodo raíz y otro nodo cualquiera es el más corto. Para cada nodo se necesita el árbol con ese nodo como raíz. El algoritmo de Dijkstra crea el árbol a partir de un grafo. El algoritmo divide los nodos en 2 grupos: tentativo y permanente. Al inicio el nodo raíz pasa a la lista tentativa, luego entra en un bucle hasta que la lista tentativa este vacía, si no está vacía mueve el nodo de camino más corto de la lista tentativa a la permanente, el primero va a ser el nodo raíz, luego añade cada nodo vecinos del nodo movido a la lista tentativa si no están en ella y si están con un coste acumulado mayor lo reemplaza con el nuevo y vuelve a evaluar el bucle, Para encontrar el camino más corto en cada etapa se necesita el costo acumulado desde el nodo raíz a cada nodo.

**Calculo de la tabla de encaminamiento a partir del árbol:** cada nodo usa el protocolo del árbol de camino más corto para armar su tabla. La tabla muestra el costo para alcanzar cada nodo a partir de la raíz.

Ventajas

– Comunicación con todos los nodos de la red

– Velocidad de convergencia más rápida que RIP

– Mensajes autenticados

– Posibilidad de estructura jerárquica en grandes dominios

– Cada nodo comprueba por sí mismo la ruta más corta

Desventajas

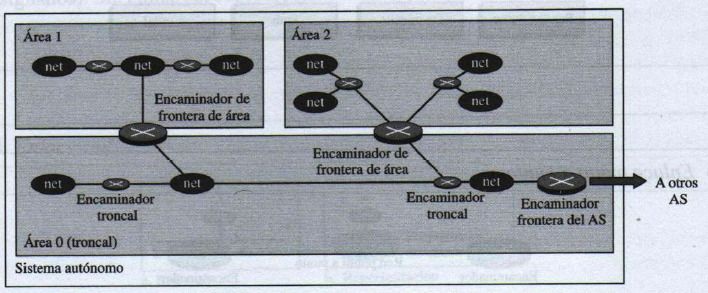
– Comunicación con todos los nodos de la red

– Un nodo que falla puede causar confusión informando mal los costos del enlace.

– Sobrecarga de la red

**OSPF- PROTOCOLO ABIERTO DE PRIMER CAMINO MAS CORTO**

Es de intradominio se basa en el encaminamiento en el estado del enlace. Su dominio es un AS. OSPF divide un AS en áreas, un **área** es una colección de redes, estaciones routers conectados dentro de un AS. Todas las redes dentro de un área deben estar conectadas.

Los routers dentro de un área inundan el área con información de encaminamiento. En el borde de un área, los routers se denominan **routers de frontera de área** resumen la información sobre el área y al envían a otras áreas. Hay un área especial en un AS denominada **troncal**, todas las áreas de un AS deben estar conectadas a la troncal. Los routers de la troncal se denominan routers troncales. Cada área tiene asignada un identificador.

El protocolo **OSPF** permite al administrador asignar un costo a cada router. Este costo puede basarse en un tipo de servicio y el router puede tener varias tablas una para cada tipo. Cada conexión se denomina enlace. Se definieron 4 tipos de enlaces:

* **Punto a punto:** conecta 2 routers, los costos son los mismos en ambos extremos.
* **De transito:** es una red con varios routers conectados a él. Los datos entran y salen a través de cualquier router. La red no puede funcionar como router por lo que un router toma esa responsabilidad y se denomina designado. Hay costo desde cualquier nodo al designado, pero no en viceversa.
* **Stub:** es una red que se conecta a solo una red. Los paquetes entran y salen de la red a través de un mismo router.
* **Virtual:** cuando se rompe un enlace entre 2 routers se crea este enlace usando un camino más largo y a través de varios routers.

Resumen realizado siguiendo las filminas de Direccionamiento y utilizando teoría del libro Redes de Computadoras. 5ta Edición. James F. Kurose & Keith W. Ross - capítulo 4.

