

### **Unidad 5: La Capa de Red**

La capa de red se ocupa de llevar los paquetes desde el origen hasta el destino. Llegar al destino puede requerir muchos saltos por enrutadores intermedios. Esta función ciertamente contrasta con la de la capa de enlace de datos, que solo tiene la meta modesta de mover marcos de un extremo del alambre al otro. Por tanto, la capa de red es la capa mas baja que maneja la transmisión de extremo a extremo.

Para lograr su cometido, la capa de red debe conocer la topología de la subred de comunicaciones ( es decir los routers) y escoger las trayectorias apropiadas a través de ella de modo de evitar la sobrecarga de algunas de ellas mientras otras estén sin trabajo.

Servicios proporcionados a la capa de transporte

La interfase entre la capa de red y la capa de transporte tiene mucha importancia ya que frecuentemente es la interfaz entre la portadora y el cliente, es decir, el límite de la subred.

Los servicios de la capa de red se diseñaron con las siguientes metas en mente:

1. Los servicios deben ser independientes de la tecnología de la subred
2. La capa de transporte debe estar aislada de la cantidad, tipo y topología de las subredes presentes.
3. Las direcciones de red disponibles para la capa de transporte deben seguir un plan de numeración uniforme, aun a través de varias LAN y WAN.

Como estas metas son bastante amplias, sin contradecirlas se han establecido dos criterios en lo que debería ser la capa de red, por un lado, el bando representado por la comunidad de internet, sugiere que la capa de red debería brindar un servicio sin conexión; ser de baja complejidad, ocuparse de transportar paquetes de un lado a otro considerando que al red es inherentemente inestable y que las capas superiores se harán cargo del control de errores, es decir desplazan la complejidad hacia arriba. Por otro lado, el bando representado por las compañías telefónicas y afines, consideran que la capa de red debería brinda un servicio orientado a conexión sobre redes confiables. Los dos ejemplos de estas orientaciones son el TCP/IP de internet (servicio sin conexión) y ATM (servicio orientado a conexión).

Organización interna de la capa de red

Básicamente hay dos filosofías diferentes de organización de las funciones de la capa de red, coincidentes con las dos posturas mencionadas anteriormente, una que usa conexiones y otra que funciona sin conexiones:

Circuitos Virtuales

Los circuitos virtuales generalmente se usan en subredes cuyo servicio primario está orientado a conexión. En ellos, se escoge y se recuerda una ruta de la máquina origen a la máquina destino como parte del establecimiento de la conexión, tal como sería un sistema telefónico. Al liberarse la conexión, también deja de existir el circuito virtual.

Datagramas

En contraste, en una red de datagramas no se determinan rutas por adelantado, aun si el servicio fuera orientado a conexión. Cada paquete enviado se enruta de manera independiente de sus antecesores.

Si bien las subredes de datagramas tienen mas trabajo que hacer, generalmente también son mas robustas y se adaptan a las fallas y congestionamientos con mayor facilidad que las de circuitos virtuales.

Comparación de ambas organizaciones

Tal como veremos a continuación, ambas políticas tienen ventajas y desventajas según la función que cumplan y el enfoque utilizado e inclusive podríamos encontrar puntos de encuentro entre ellas.

ASUNTO	Subred de datagramas	Subred de Circ. Virt.
Establecimiento del circuito	No es necesario	Es requerida
Direccionamiento	Cada paquete contiene completas las direcciones de origen y destino	Cada paquete contine un número corto de CV
Información de estado	La subred no contiene información de estado	Cada CV requiere espacio en la tabla de estados de la subred.
Enrutamiento	Cada paquete se enruta independientemente	La ruta es escogida cuando se establece el CV. Todos los paquetes siguen esa ruta.
Efecto de fallas de un router	Ninguno a excepción de algunos paquetes perdidos durante la caída	Terminan todos los CV que pasan a través del router
Control de congestionamientos	Difícil	Sencillo si existen

		suficientes buffers para cada CV por adelantado.
--	--	--

Es importante indicar explícitamente que el servicio ofrecido (con o sin conexión) es independiente de la estructura de la subred (circuitos virtuales o datagramas) y en teoría son posibles las cuatro combinaciones aunque es claro que son razonables las implementaciones de CV con un servicio orientado a conexión y DATAGRAMAS con un servicio sin conexión.

#### Algoritmos de enrutamiento

El algoritmo de enrutamiento es aquella parte del software de la capa de red encargada de decidir la línea de salida por la que se transmitirá cada paquete de la entrada. Si la subred usa datagramas internamente, esta decisión debe hacerse cada vez que llega un paquete a la entrada. En cambio, si la subred usa CV, las decisiones de enrutamiento se toman solamente al establecerse la comunicación.

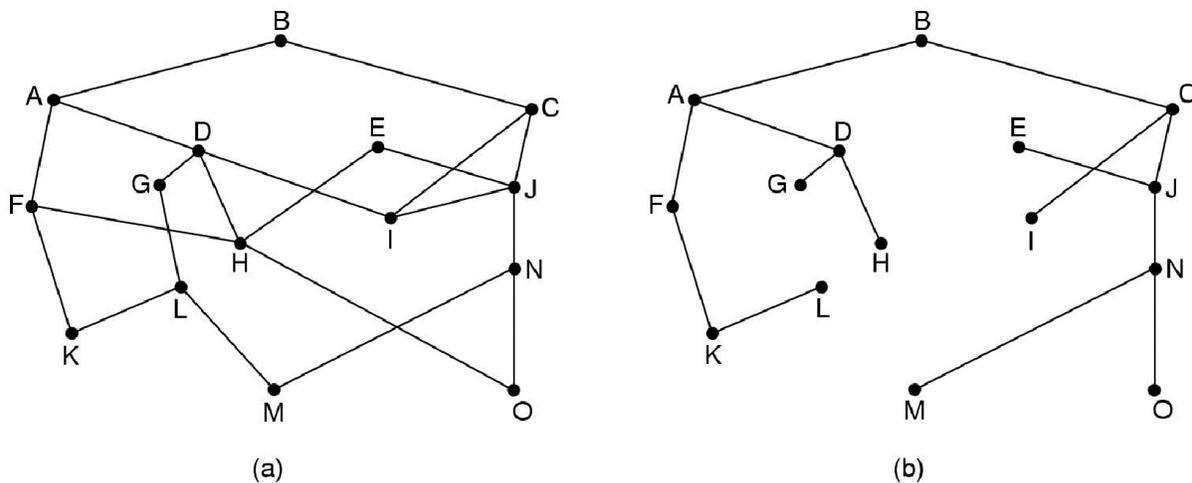
Un buen algoritmo de enrutamiento debe ser capaz de manejar los cambios de topología de la red y el exceso de tráfico sin requerir el aborto de todas las actividades y el rearranque de la red. Así también, este algoritmo debe ser equitativo para no dar prioridad a ciertos nodos si esto no es requerido.

Los algoritmos de enrutamiento pueden agruparse en dos clases: adaptativos y no adaptativos: en el segundo caso, la decisión de que ruta se usará para llegar del origen al destino se calcula por adelantado, fuera de línea, y se carga en los routers al iniciar la red, también son llamados enrutamientos estáticos. Los algoritmos adaptables, en contraste, cambian sus decisiones de enrutamiento para reflejar los cambios de topología, y generalmente también de tráfico cada cierto tiempo definido.

#### Principio de optimización

Es útil señalar que es posible hacer un postulado general sobre las rutas óptimas sin importar la topología ni el tráfico de red. Este postulado se conoce como principio de optimización, y establece que, si el router  $J$  está en la trayectoria óptima del router  $I$  al router  $K$ , entonces la trayectoria óptima de  $J$  a  $K$  también está en la misma ruta.

Como consecuencia de esto podemos ver que el grupo de trayectorias óptimas de todos los orígenes a un destino dado forma un árbol con raíz en el destino llamado árbol de descenso. La métrica de distancia de este árbol es el número de escalas que debe hacer el paquete y es posible que el árbol no sea único sino que hay otros posibles con métricas iguales.



Dado que es un árbol, no contiene ciclos, por lo que cada paquete será entregado con un número de escalas finito y limitado. En la práctica, la solución no es tan fácil, ya que los enlaces y routers pueden caerse y reactivarse durante su operación normal.

Veremos a continuación algunos ejemplos de algoritmos estáticos y adaptables:

#### Enrutamiento por trayectoria mas corta

La idea de esta técnica es armar un grafo de la subred en la que cada nodo representa un router y cada arco una línea de comunicación (o enlace). Para escoger una ruta entre un par dado, el algoritmo simplemente encuentra en el grafo la trayectoria mas corta entre ellos. Es importante destacar que es lo que se considera como *trayectoria mas corta* ya que esto depende de la métrica que tomemos, es decir que el algoritmo llegará a distintas conclusiones si la métrica usada es la cantidad de saltos entre routers o si la métrica tiene en cuenta la distancia en kilómetros entre ellos o si consideramos el retardo promedio.

El algoritmo más conocido se debe a Dijkstra (1959), en el, cada nodo se etiqueta con su distancia más corta al nodo origen. Inicialmente no se conocen las trayectorias, por lo que todos los nodos tienen la etiqueta infinito. A medida que avanza el algoritmo y se encuentran trayectorias, las etiquetas pueden cambiar, reflejando mejores trayectorias. Las etiquetas pueden ser tentativas o permanentes, una vez que se descubre que no puede hallarse una trayectoria mejor, la etiqueta pasa de tentativa a permanente y no se vuelve a modificar hasta que cambie el grafo.

#### Algoritmo de inundación

Otro algoritmo estático es el de inundación, en el que cada paquete de entrada se envía por cada una de las líneas de salida, excepto por aquella por la que llegó. Esto, evidentemente genera gran cantidad de paquetes duplicados. Este efecto se ve disminuido por la implementación de algunos trucos como el *contador de escalas* contenido en la cabecera de cada mensaje. Este contador, inicializado por el origen en un valor que puede ser la máxima distancia de la subred o el máximo tamaño de la trayectoria entre el origen y el destino; disminuye uno en cada escala haciendo desaparecer el paquete cuando llega a cero.

Una variación de este método, un poco más práctica, es la **inundación selectiva**. En este algoritmo, los routers no envían los paquetes de entrada por todas las salidas sino sólo por aquellas que van aproximadamente en la dirección correcta.

La inundación no es para nada práctica salvo para aquellas aplicaciones donde se puede producir una caída importante en el número de los routers y la información es muy importante ya que tiene la ventaja de producir una gran cantidad de redundancia.

#### Enrutamiento basado en flujo

Hasta ahora solo hemos observado algoritmos que tienen en cuenta la topología para decidir las rutas pero en ciertos casos, si algún enlace está saturado de tráfico podría ser conveniente enviar los paquetes por una trayectoria más larga pero con menos congestión.

El concepto de este algoritmo se basa en la posibilidad de predecir cual será el tráfico a través de las líneas y el retardo promedio en cada momento. El problema se reduce entonces a estudiar una matriz con los valores de retardo promedio de toda la subred y encontrar el camino con el retardo promedio mínimo en cada momento (intervalo de tiempo).

#### Enrutamiento por estado de enlace

El concepto de este algoritmo es muy sencillo y puede postularse en cinco partes:

1. Descubrir a sus vecinos y conocer sus direcciones de red.
2. Medir el retardo o costo para cada uno de sus vecinos.
3. Construir un paquete que indique todo lo que acaba de aprender.
4. Enviar este paquete a todos los routers conocidos.
5. Calcular la trayectoria más corta a todos ellos por el método de Dijkstra.

#### Enrutamiento para hosts móviles

El tema de redes con usuarios móviles es una de las complicaciones más importantes a las que se enfrentan los desarrolladores de los protocolos actuales. Para comprender el funcionamiento de estos debemos hacer algunas aclaraciones:

- Las redes llamadas inalámbricas están formadas por partes alambradas que conectan ciertos nodos celulares que son los que se encargan de recibir a los usuarios móviles.
- Hay usuarios móviles que en ciertas circunstancias permanecen conectados a la red forma alambrada.
- Los usuarios móviles tienen una estación base en la red alambrada donde son reconocidos y validados.
- Cada área posee uno o más agentes foráneos que llevan el registro de todos los usuarios que visitan el área.
- Además poseen un agente de base que lleva el registro de todos los usuarios móviles que actualmente están visitando otra área.

Al entrar un usuario nuevo en un área, ya sea al conectarse a ella (via la LAN) o simplemente entrar a una célula, su computadora debe registrarse con el agente foráneo de ese lugar. El procedimiento suele ser así:

1. Periódicamente, cada agente foráneo difunde un paquete que anuncia su existencia y dirección. Un host móvil recién llegado puede esperar uno de estos mensajes, pero si no llega ninguno con suficiente rapidez, el host móvil puede difundir un paquete que diga: ¿Hay agentes foráneos por ahí?
2. El host móvil se registra con el agente foráneo, dando su dirección base, su dirección actual de capa de enlace de datos y cierta información de seguridad.
3. El agente foráneo se pone en contacto con el agente de base del host móvil y le dice: "uno de tus hosts está por aquí". El mensaje del agente foráneo al agente de base contiene la dirección de red del agente foráneo, así como la información de seguridad, para convencer al agente de base de que el host móvil en realidad está ahí.

4. El agente de base examina la información de seguridad, que contiene una marca de tiempo, para comprobar que fue generada en los últimos segundos. Si está conforme, indica al agente foráneo que proceda.
5. Cuando el agente foráneo recibe el reconocimiento del agente de base, hace una entrada en sus tablas e informa al host móvil que ahora está registrado.

#### Enrutamiento por difusión

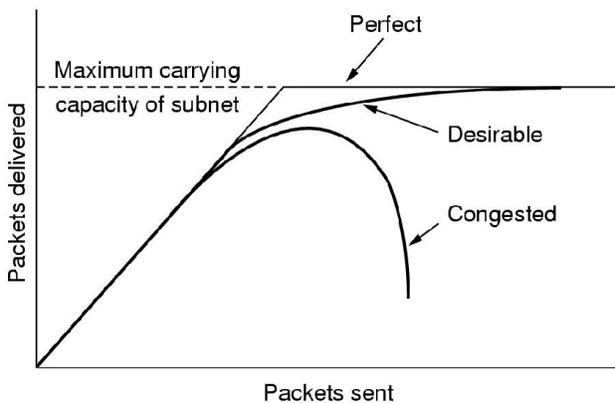
En algunos casos en donde es deseable que un host distribuya información en forma de broadcast a todos los destinos se utiliza este algoritmo llamado por difusión, similar al de inundación, solo que requiere que el host emisor tenga la lista de los host destino para no generar una replicación infinita de los paquetes.

#### Enrutamiento por multidifusión

Es algo similar al anterior con la diferencia que la lista de hosts destino es lo suficientemente corta como para que no justifique el envío a todos los nodos de la red. Este algoritmo tiene la posibilidad de organizar grupos de nodos de manera que tiene diversas listas de hosts bien identificadas.

#### CONGESTIÓN

Cuando hay demasiados paquetes presentes en una parte de la subred, hay una degradación del desempeño. Esta situación es denominada *congestionamiento*. En la figura siguiente se muestra este síntoma:



Cuando la cantidad de paquetes descargados en la subred por los hosts está dentro de su capacidad de conducción, todos se entregan y la cantidad de entregados es igual o proporcional al número de enviados. Sin embargo, a medida que aumenta el tráfico, los routers ya no pueden manejarlo y comienzan a perder paquetes. Esto tiende a empeorar las cosas ya que los pedidos de retransmisión saturan la subred y casi no hay entrega de paquetes.

Vale la pena indicar explícitamente la diferencia entre control de congestionamiento y control de flujo. El control de congestionamiento tiene que ver con el asegurarse de que la subred pueda transportar todo el tráfico ofrecido. En contraste, el control de flujo tiene que ver con la capacidad que tiene el host de destino para recibir todos los paquetes que le envió el origen y que pasan la subred sin problemas. La razón por la que se confunde los conceptos es porque un emisor puede recibir mensajes que lo intiman a reducir el flujo, tanto porque la subred está congestionada como porque el host de destino no puede manejar la cantidad de paquetes que le envían y muchos algoritmos tienen las funciones para ambos controles.

Comenzaremos el estudio examinando un modelo general para manejarlo y posteriormente veremos como prevenirlo.

#### Principios generales del control de congestionamiento

Vamos a utilizar parte de la teoría de control para encarar este modelo. Este enfoque conduce a la división de las soluciones en dos grupos: de ciclo abierto y de ciclo cerrado. Esencialmente las soluciones de ciclo abierto intentan resolver el problema con un buen diseño sin intentar ajustar el mismo posteriormente a medio camino. Las herramientas para llevar a cabo el control de ciclo abierto incluyen decidir cuando aceptar tráfico nuevo, cuando descartar paquetes, etc. Todas tienen en común el hecho de que son decisiones independientes del estado actual de la red.

En contraste, las soluciones de ciclo cerrado se basan en el concepto de un ciclo de realimentación. Este enfoque tiene tres partes cuando se aplica al control de congestionamiento:

1. Supervisar el sistema para detectar cuando y donde ocurren congestionamientos.
2. Pasar esta información a los lugares en donde pueden llevarse a cabo acciones.
3. Ajustar la operación del sistema para corregir el problema.

Pueden usarse varias métricas para supervisar la subred en busca de congestionamientos como: porcentaje de paquetes descartados por falta de espacio en los buffers, la longitud promedio de las colas, la

cantidad de paquetes con tiempo de vida agotado, el retardo promedio de los paquetes, etc. En todos los casos, un aumento de las cifras indica un aumento del congestionamiento.

Ahora debemos realizar la realimentación a los hosts o enrutadores correspondientes para que puedan tomar alguna medida. Hay varias maneras entre las que se destacan que el router que detecte el problema envíe un paquete anunciando la congestión o la reserva de un bit o un campo en el encabezado de los paquetes para que el router lo llene cuando rebasa algún umbral.

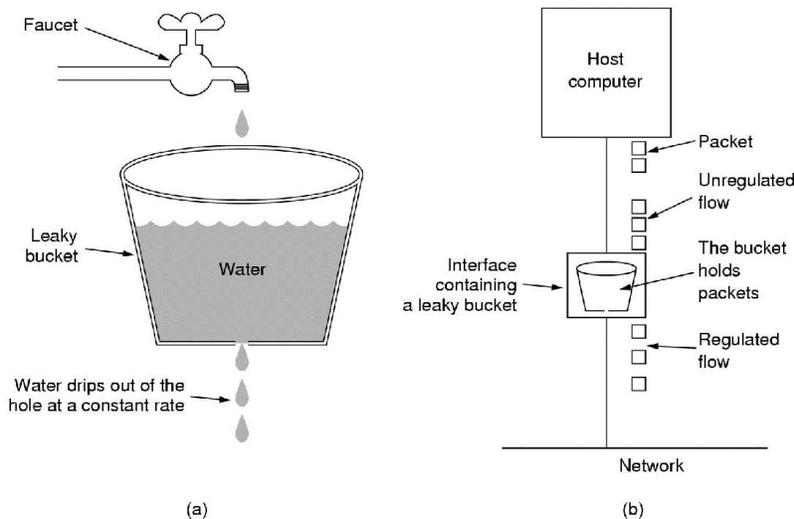
Los algoritmos de control de congestionamiento de ciclo abierto suelen clasificarse en: actuantes en el origen y actuantes en el destino.

Los algoritmos de control de congestionamiento de ciclo cerrado también se clasifican en de realimentación implícita y realimentación explícita, en el segundo caso, se envían paquetes de regreso desde el punto de congestionamiento para avisar al origen; en los algoritmos implícitos, el origen deduce la existencia de un congestionamiento haciendo observaciones locales, como el tiempo necesario para el regreso de los paquetes de reconocimiento.

La presencia de congestionamiento significa que la carga es (temporalmente) mayor en alguna parte del sistema, automáticamente surgen dos soluciones, o aumentar los recursos o reducir la carga, en algunos casos es posible aumentar los recursos temporalmente pero en la mayoría de ellos, la única solución es disminuir la carga que implica por ejemplo negar el servicio a algunos usuarios o degradarlo proporcionalmente a todos, etc.

#### Algoritmo de la cubeta con goteo

Imaginemos una cubeta con un pequeño agujero en el fondo.



Sin importar la rapidez con que entra el agua en la cubeta, el flujo de salida tiene una tasa constante,  $\rho$ , cuando hay agua en la cubeta y 0 (cero) cuando la cubeta está vacía. También una vez que se llena la cubeta, cualquier agua adicional que entra se derrama por los costados y se pierde.

Puede aplicarse el mismo concepto a los paquetes. Conceptualmente, cada host está conectado a la red mediante una interfaz que contiene una cubeta con goteo. Si llega un paquete a la cola cuando está llena se descarta el paquete. Esto no es otra cosa que un sistema de encolamiento con un tiempo de servicio constante.

El host puede poner en la red un paquete por pulso de reloj. Este mecanismo convierte un flujo desigual de paquetes (ráfagas) de los procesos de los usuarios en un flujo continuo de paquetes hacia la red, moderando estas ráfagas y reduciendo en buena medida las posibilidades de congestionamiento.

Existen además otros algoritmos más sofisticados que permiten regular el flujo de la cubeta para liberar más paquetes si es necesario y la subred lo permite.

#### INTERCONEXIÓN DE REDES

Hasta ahora hemos supuesto implícitamente que hay una sola red homogénea y que cada máquina usa el mismo protocolo en cada capa. Por desgracia, este supuesto es demasiado optimista. Existen muchas redes diferentes que sin embargo pueden interconectarse formando una interred.

Vamos a imaginar una caja negra que se ocupará de realizar las tareas necesarias para que redes distintas puedan entenderse; podemos encontrar 4 tipos de estas cajas negras, de acuerdo a su función específica:

Capa1: Los HUBS o repetidores copian bits individuales entre distintos segmentos de cable. (LAN)

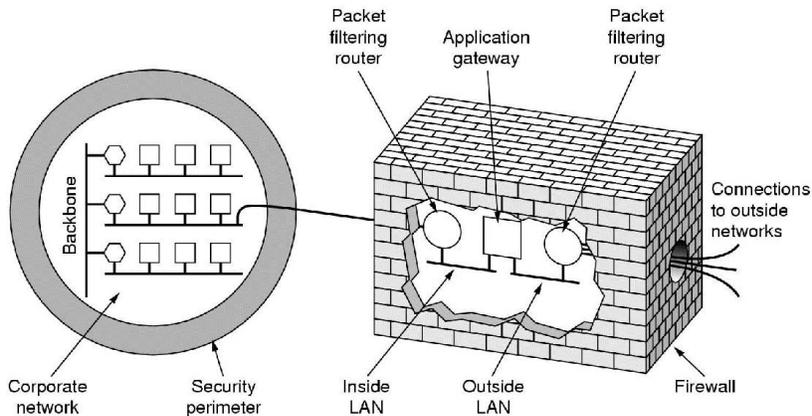
Capa2: Los BRIDGES o puentes al igual que los SWITCH almacenan y reenvían marcos de enlace entre las LAN

Capa3: Los ROUTERS multiprotocolo reenvían paquetes entre redes disímiles.

Capa4: Los GATEWAYS o pasarelas de transporte conectan corrientes de bytes en la capa de transporte.

Arriba de la 4: Las GATEWAYS o pasarelas de aplicación permiten el trabajo conjunto por encima de la capa 4.  
Firewalls

Los firewalls son simplemente una adaptación moderna del viejo sistema de seguridad medieval: un foso profundo alrededor del castillo. Este diseño obligaba a cualquiera que entrara o saliera a pasar por un solo puente levadizo, donde podía ser inspeccionado por la policía de E/S. En las redes, este puente levadizo es reemplazado por el firewall tal como se muestra en la figura siguiente:



El firewall en esta configuración tiene dos componentes: dos routers que filtran paquetes y un gateway de aplicación. Existen configuraciones más sencillas, pero la ventaja de este diseño es que cada paquete debe transitar por dos filtros y una pasarela para entrar o salir.

La primera mitad del sistema son los routers que poseen un sistema de inspección de paquetes que tienen que cumplir con ciertas condiciones para poder pasar al otro lado, caso contrario se descartan.

La segunda mitad del mecanismo es el gateway de aplicación que, como su nombre indica, actúa a nivel de las aplicaciones inspeccionando los paquetes para aplicaciones específicas como ser correo electrónico o Web.

La capa de Red en Internet

La capa de red en la internet está representada por el conocido protocolo IP (Internet Protocol). Este, además de establecer una numeración uniforme (los números IP), toma corrientes de datos, las divide en datagramas que generalmente son de unos 1500 bytes. Cada datagrama se transmite a través de la red, posiblemente fragmentándose en unidades más pequeñas en el camino. Cuando todas las piezas llegan finalmente a destino, son reensambladas por la capa de red, dejando el flujo de datos original.

Direcciones IP

Cada host y router en Internet tiene un número identificador único en toda la red. Estos números son asignados por el NIC (*Network Information Center*) para evitar conflictos.

**Las direcciones IP son números binarios de 32 bits que por simplicidad se escriben en notación decimal con puntos**, es decir, cada uno de los 4 bytes se escribe en decimal (de 0 a 255) y se lo separa con un punto. Es decir que los números IP pueden ir teóricamente desde 0.0.0.0 hasta 255.255.255.255, lo cual sería cierto si no se reservaran algunos números para funciones específicas como por ejemplo:

- 00000000.00000000.00000000.00000000      Este host (0.0.0.0)
- 00000000.XXXXXXXXXX.XXXXXXXXXX.XXXXXXXXXX      Un host de esta red (0.X.X.X)
- 11111111.11111111.11111111.11111111      Difusión de la red (255.255.255.255)
- 01111111.cualquier cosa      Retrociclo
- 11000000. 10101000.XXXXXXXXXX.XXXXXXXXXX      Reservado para intranets (192.168.X.X)

Subredes

La numeración IP permite la subdivisión en subredes más pequeñas conocidas como A,B,C, esto se realiza utilizando un segundo valor llamado máscara de subred que indica cuantos números IP pertenecen a la subred local haciendo una operación AND booleana.