Introducción a IP versión 4

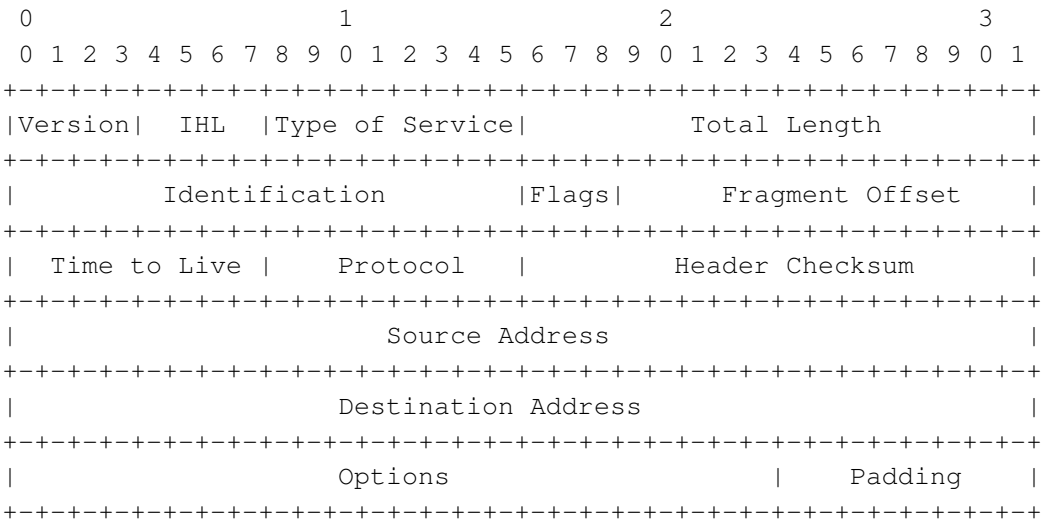
Introducción a IPv4

IPv4 (Internet Protocol versión 4) es el protocolo de nivel de red usado en Internet. Junto con otros protocolos auxiliares es responsable de transferir la información del usuario por la red. El protocolo IPv4 está definido en el RFC 791.

IPv4 es un protocolo de nivel de red no orientado a conexión, no confiable. En caso de haber problemas, se espera que el nodo involucrado descarte el paquete. Debido a que un paquete debe transitar por varios nodos, posiblemente siguiendo un camino que no necesariamente es el mismo que el usado por otros paquetes, los datos enviados pueden llegar en desorden. IPv4 no intenta corregir el orden de los paquetes.
Las características de IPv4 hacen que Internet sea principalmente una red “best effort”, o sea que no provee ninguna garantía sobre el tráfico, aunque haciendo su mejor esfuerzo para asegurarse que los datos lleguen a destino.
Aunque también existe IPv6, cuando se use el término “IP” en este documento, se considerará que se está haciendo referencia a IPv4.

El formato del paquete IPv4

El protocolo IPv4 tiene un header de longitud variable. El header está formado por una parte obligatoria, de 20 bytes, seguido por una serie de opciones. Debido a limitaciones del header, las opciones deben tener una longitud múltiplo de 4 bytes, pudiendo el header crecer hasta un máximo de 60 bytes (contando parte obligatoria y opcional).

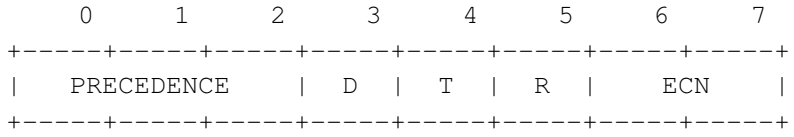


Los headers obligatorios son los siguientes (considerando big endian a los numeros mencionados)

- Version (4 bits): Es el número de versión del protocolo IP. Una constante “4”. Permite que otras versiones interactuen en la misma red sin causar conflictos. La otra versión usada comunmente es IPv6.

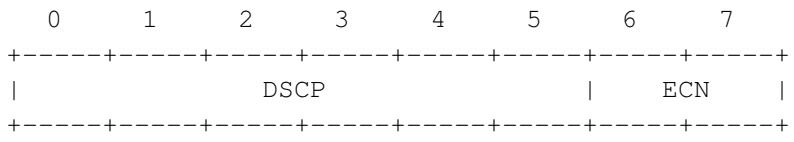
- IHL (4 bits): Es la longitud del header IPv4. Puesto que el valor cubre de 0 a 15, la medida usada es bloques de 32 bits. Debe indicar al menos 5.
- Type of service o TOS (8 bits): Es un campo de bits que indica como se debe tratar al paquete en cuestión. Usado para priorizar algunos paquetes sobre otros.
- Total length (16 bits): Es la longitud total del paquete, medida en bytes. Esto significa que un paquete IPv4 no puede tener una longitud mayor a 64 KiB.
- Identification, flags y fragment offset: Estos campos son usados en la fragmentación de paquetes IPv4 (a ver en la siguiente clase).
- Time to live, o TTL (8 bits): Es un campo usado para evitar que un paquete quede circulando indefinidamente en la red. Originalmente indicaba la cantidad de segundos que el paquete puede permanecer. Hoy en día indica la cantidad de saltos que puede realizar. Este campo se cambia al pasar por cada router, y el paquete se descarta si este valor llega a 0.
- Protocol (8 bits): Indica cual es el protocolo de la capa superior. Usado para permitir llevar múltiples protocolos sobre IPv4 (similar al ethertype de ethernet).
- Header checksum (8 bits): Es un checksum que protege al header. No brinda mucha protección y no protege a los datos. Si el checksum en un paquete no es correcto, se descarta el mismo. Es principalmente una medida para evitar la propagación innecesaria de paquetes.
- Source address y destination address (32 bits cada una): Indican la dirección de origen y destino del paquete. El origen está incluído para permitir que el receptor sepa a quien debe responder y también asiste en el mantenimiento del estado en protocolos de capas superiores.

El campo TOS en IPv4



IPv4 provee ciertos mecanismos de priorización de paquetes. La especificación original de IPv4 indica que debe considerarse un campo de 3 bits que indica la prioridad (precendence), con otros tres bits indicando el servicio deseado y dejando los últimos dos bits en cero (por estar reservados). Los bits que indican el tipo de servicio son (en orden): “minimizar la latencia” (bit D), “maximizar el throughput” (bit T) y “maximizar la confiabilidad” (bit R). Un host puede prender cualquier combinación estos bits y los routers pueden usarlos como guia para determinar el próximo salto.
Supongamos que un router está conectado a otro con dos enlaces, uno satelital con mucho throuput y mucha latencia y un cable serie, con poca latencia y poco throughput. Si un paquete IPv4 con el bit D debe ser forwardeado, es muy probable que el router decida usar el enlace serie, ya que es el que menos latencia tiene. Si el siguiente paquete tiene el bit T prendido, es de esperar que sea enviado por el enlace satelital. Todo este tratamiento es opcional, y en última instancia depende del router que hace el forwardeo (y de su configuración/programación).

Otra interpretación posible es el uso total del campo de bits para indicar distintas clases de tráfico (indicadas en el campo DSCP). Puesto que esta interpretación fue definida mucho despues de la interpretación original, es necesario definirlas de forma tal que sean compatibles con la interpretación original, a menos que haya un arreglo de antemano entre todos los nodos de la red. Esta nueva interpretación se encuentra en el RFC 2474.



Es importante aclarar que los dos bits reservados fueron asignados recientemente para control de congestión (marcados como “ECN”), por lo que ningún mecanismo de ToS debe alterarlos.

Anatomía de una dirección IPv4

Una dirección IPv4 está formada por dos partes: la parte de red y la parte de host. Esta división no es evidente en la dirección en sí, sino que está indicada por otro parámetro llamado la máscara de red. Una máscara es una serie de bits en uno, seguidos por bits en cero hasta completar 32 bits. Cuando una máscara está asociada a una dirección IPv4, los unos de la misma indican los lugares usados en la parte de red, mientras que los ceros marcan la parte de host. Es común escribir una máscara de forma similar a una dirección IPv4, aunque también se acostumbra abreviarla escribiendo el número de bits en uno que esta posee.
La razón de esta división es que no es factible conectar directamente a todos los hosts de Internet por medio del nivel de enlace, y al mismo tiempo no es posible asignar una entrada en cada tabla de ruteo a cada host posible. Para evitar esto, se define la idea de “red”: un grupo de nodos conectados directamente con un protocolo de nivel de enlace. La idea es agrupar a los hosts que se encuentren en la misma red y asignarles un bloque de direcciones. Una vez hecho esto, es posible realizar el ruteo hacia distintas redes, ahorrando el costo de hacer un ruteo individual a cada host.
Por medio de la máscara se pueden obtener dos valores importantes: la dirección de red y la dirección de broadcast. La dirección de red es la dirección cuya parte de host contiene todos ceros y sirve para hacer referencia a la red como entidad. La dirección de broadcast es la dirección con la parte de host llena de unos, y sirve para enviar un mensaje a todos los nodos de la red. Estas dos direcciones están reservadas y no pueden ser asignados a ningún host o router. Estos parámetros usualmente se encuentran en la configuración básica de un host.
La máscara a usar depende del tamaño de la red. Cuantos más unos tenga, mas pequeña es la red. Obviamente en la red debe haber suficiente lugar para los nodos que la integran, pero usar una máscara demasiado pequeña causa que se desperdicien direcciones (ya que no hay hosts en esa red que puedan usarlas, lo que significa que bajo circunstancias normales no se terminan usando). En contraposición, cuanto más grande es una máscara, menos direcciones hay en total, y más se desperdicia el espacio en direcciones de red y broadcast (hasta llegar a /30, que desperdicia 50% de las direcciones).

Dirección	Máscara (explícita)	Máscara (resumida)	Cantidad de direcciones disponibles	Dirección de red	Dirección de broadcast
157.92.27.2	255.255.0.0	/16	65533	157.92.0.0	157.92.255.255
200.80.40.197	255.255.255.240	/28	14	200.80.40.192	200.80.40.207
5.2.7.1	255.255.255.252	/30	2	5.2.7.0	5.2.7.3
201.213.16.47	255.255.255.0	/24	253	201.213.16.0	201.213.16.255
64.41.98.153	255.255.255.128	/25	126	64.41.98.128	64.41.98.255

Table 1: Distintas direcciones con máscaras y sus direcciones de red y broadcast.

Ruteo básico

El forwarding de paquetes es similar a lo visto en teoría: cada datagrama se procesa independiendetemente del resto, y se lo compara con las entradas de la tabla de ruteo. La principal diferencia es que la comparación no se realiza con destinos finales, sino con distintas redes, que engloban destinos. Para ello, cada entrada debe tener una dirección de red y una máscara, usadas para buscar, y un siguiente salto, o una indicación de que el destino es local. Como es posible que hayan varias entradas que coincidan con la dirección de destino, se seguira la ruta más precisa (la que tiene una máscara más grande).

Destino	Interfaz	Siguiente salto
157.92.0.0/16	eth0	200.80.40.193
192.168.30.0/23	ser1	192.168.0.5
157.92.75.5/32	ser1	192.168.0.5
192.168.0.4/30	ser1	Directo
200.80.40.192/28	eth0	Directo
0/0	eth0	200.80.40.193

Table 2: Ejemplo de una tabla de ruteo, el host está conectado directamente a dos redes, usando las interfaces eth0 y ser1. Es importante notar la ruta default y la precedencia que toma la tercera regla sobre la primera si ambas se aplican.

Por cada paquete que debe ser forwardeado, se realiza el siguiente algoritmo:

- Por cada entrada de la tabla
    - Aplicar (AND) la máscara de la entrada a la dirección destino.
    - Comparar el resultado del paso anterior con la dirección de red de la entrada.
    - Si hay coincidencia, guardarlo como posible ruta.
- Si hay rutas posibles a usar, buscar la de máscara más grande y tomar la acción correspondiente.
- Si no hay, descartar el paquete porque no es un destino alcanzable.

Una ruta muy especial es la “ruta default” o “default gateway”, que es una ruta cuya máscara es /0. Esta ruta se toma cuando no hay coincidencia con ninguna otra entrada en la tabla de ruteo, ya que es la menos precisa de todas.

Para determinar si un paquete está destinado a una red local, se puede usar el procedimiento indicado arriba, usando la máscara y dirección IPv4 del host para obtener la dirección de red local, repitiendo el procedimiento usado con las entradas de la tabla de ruteo. Normalmente no es necesario hacer esta verificación, ya que la tabla de ruteo contiene entradas para destinos locales.

Como manejar espacios de direcciones

Si todas las computadoras de una red estuvieran conectadas directamente por el nivel de enlace, serían válidas como direcciones de host cualquier combinación de octetos, excepto 0.0.0.0 y 255.255.255.255. Puesto que Internet esta dividida en subredes, es necesario fraccionar este espacio de acuerdo con las

necesidades particulares de sus usuarios. Sin embargo, puesto que para definir una red se emplea una máscara, no es posible hacer una partición arbitraria del espacio de direcciones. Dada una partición válida, la unica forma de volver a partir es partir ese espacio a la mitad (al mover la máscara un bit a derecha y reescribir ese espacio como dos redes distintas). Toda partición que se pueda hacer de una red en distintas subredes debería poder expresarse como una serie de particiones en dos. Al aplicar cualquier partición, es importante prestar atención a las direcciones de red de las redes resultantes. Estas deben (una vez movida la máscara) tener su parte de host llena de ceros. Un buen método consiste en mover la máscara y luego incrementar en 1 (binario) la parte de red mientras el resultado se encuentre en la red original. Sin importar el método usado, si uno toma una dirección de red y la incrementa en n (donde n es la cantidad de direcciones de esa subred), el resultado debería ser la dirección de otra red (pero no necesariamente dentro del rango asignado, si uno aplica esto a la última subred de la partición).
El procedimiento aplicado para dividir una red en dos partes (llamado subnetting) puede aplicarse en reversa para obtener una red más grande. Para que este procedimiento (supernetting) funcione, deben usarse dos subredes vacías tales que al correr la máscara un bit a izquierda se vea la misma dirección de red en ambas. Si esto sucede, es porque (al menos conceptualmente) estas dos subredes fueron creadas por la subdivisión de la red más grande anteriormente.

Resumiendo:

- Uno no debe salirse de los limites del espacio asignado.
- Es posible partir un espacio en dos, moviendo la máscara un lugar a la derecha, teniendo dos redes, una con la misma base que la original y otra que comienza en la mitad del espacio partido.
- Es posible mover la máscara varios lugares a la derecha: en ese caso, la cantidad de redes obtenidas es una potencia de dos, y todas estan espaciadas regularmente a lo largo del espacio original. Es equivalente a realizar sucesivas particiones en dos.
- Es posible volver a juntar dos redes en una más grande, siempre que ambas pertenezcan a un par que fue partido anteriormente (ambas deben indicar la misma dirección de red si se mueven sus máscaras a izquierda un lugar), y si ambas estan libres (si estas fueron partidas, el algoritmo debe aplicarse recursivamente, si hay alguna subred ocupada, la red NO esta libre).
- La máscara mas grande de red, bajo circunstancias normales, es /30.
- Es recomendable asignar primero las redes grandes para evitar fragmentar el espacio de direcciones.