SUITE TCP/IP

Suite transmission Control Protocol / Internet Protocol.

Son muchos protocolos, incluido TCP/IP.

Contenido

[Arquitectura TCP/IP 2](#_Toc114161258)

[Filosofía 2](#_Toc114161259)

[IP = Servicio connectionless 3](#_Toc114161260)

[Servicio connectionless 3](#_Toc114161261)

[Datagrama IP 4](#_Toc114161262)

[Fragmentación y reensamblado 7](#_Toc114161263)

[Direccionamiento 9](#_Toc114161264)

[Subredes 10](#_Toc114161265)

[Por que surgen 12](#_Toc114161266)

[Definición 12](#_Toc114161267)

[Cantidad de subredes 12](#_Toc114161268)

[Clases de direcciones 13](#_Toc114161269)

[RFC1819 classfull 14](#_Toc114161270)

[Classless interdomain routing (CIDR) y VLSM 15](#_Toc114161271)

[Conceptos de enrutamiento (para clase que viene) 15](#_Toc114161272)

# Suite TCP/IP

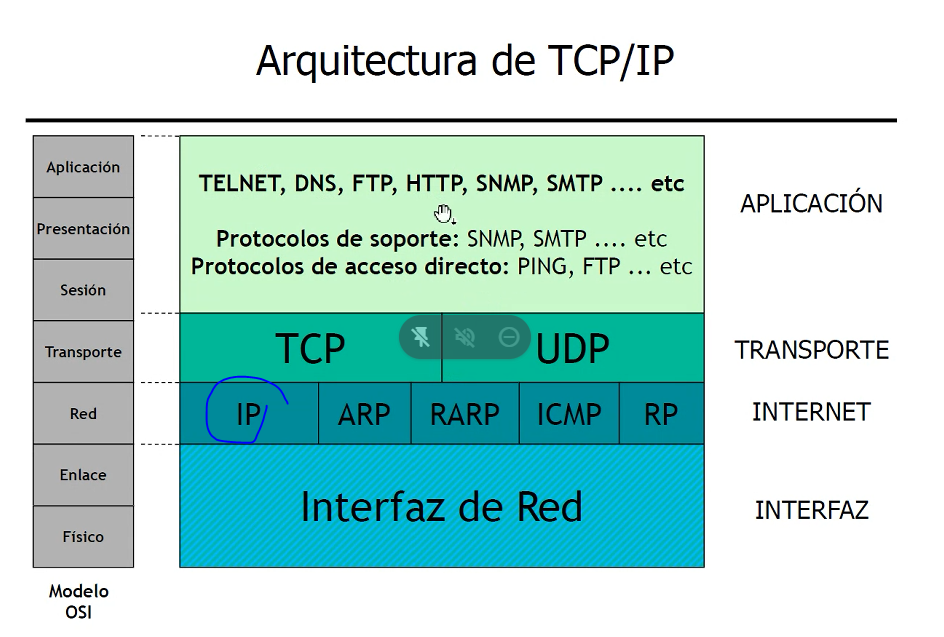
Conjunto de protocolos que se soportan sobre un ***protocolo único capa 3 (IP)***

Un protocolo de capa 3 debe ser capaz de proveer direcciones que permitan identificar a los miembros de la red y mecanismos que permitan el encaminamiento (es decir cómo llegar de un nodo a otro).

Son funciones del protocolo de red:

* Direccionamiento. Cómo identificar cada punto de la red (origen y destino).
* Encaminamiento. Cómo llegar de un punto a otro. Lo realizan los routers.   
  También: **có**mo los routers almacenan la informacion de tpologocia (como sabe este router como comunicarse con el servidor, o si existen variocaminos cual es el mejor, etc)  Cómo sabe qué camino elegir (cuál no es el camino, cuál es el mejor camino entre dos opciones, etc).

# Arquitectura TCP/IP



Con Ethernet (capa 2), podemos comunicarnos a dispositivos dentro de una misma red. Por ejemplo, un router.

Para comunicarnos con algo que está fuera de nuestra red, necesitamos el protocolo IP (capa 3).  Además, hay protocolos de soporte para IP (ARP, RARP, etc.)

En la capa de transporte (4) hay 2 protocolos: la confiable que es la TCP y la no confiable o ligera que es la UDP.

# Clasificación de protocolos

**Orientados a la Conexion.**

* Trabajan en 3 etapas:
  + 1. Establecimiento
    2. Intercambio
    3. Cierre

**NO Orientados a la Conexion (incl: IP, Ethernet)**

* Etapas. Única: Intercambio. Ejemplo: Correo.
* Características:
  + No son confiables.
  + Paquetes tratados independientemente del anterior.
  + No existe un estado en los routers sobre como se trataron ls paquetes anteriores
  + Entrega best effort\_ El software realiza un serio intento por entregar el paquete (no hay garantia de que se entrego)

# IP = Servicio connectionless

## Servicio connectionless

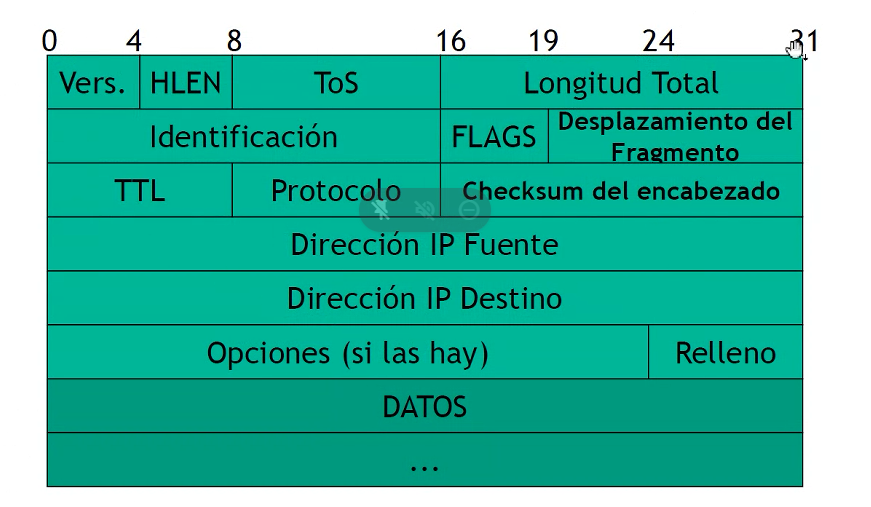
Que sea no orientado a la conexión **implica**:

* **No es confiable:** significa que **los paquetes pueden ser:**
  + **Perdidos:** puede pasar que la capa de transporte le pase una PDU a IP para que la transmita, e IP la pierda.
  + **Duplicados:** entregue dos copias del mensaje.
  + **Desordenados. No llegan en el mismo orden en que fueron enviados**
  + **Demorados:** no esta garantizado cuánto puede demorar en llegar.  
    De todas maneras IP funciona muy bien!
* **Connectionless:** **paquetes tratados independientemente.**
  + **No existe un “estado” en los routers** **acerca** de cómo fueron tratados los **paquetes anteriores**, ni que contenían. Cada paquete va a buscar como llegar al destino, no tiene un camino de routers predefinido.
* **Entrega best-effort:** el software realiza un **serio intento por entregar el paquete** (sin garantía).

## Datagrama IP

El datagrama incluye la cabecera del protocolo más los datos.

Todos los protocolos de la suite TCP/IP se representan con 32bits

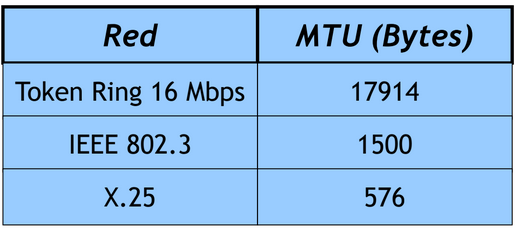


Campos:

* **Versión** (4bits): IPv4 = 0100 o IPv6 = 0110 (tip: leer en binario)
* **HLEN** - Tamaño de cabecera (4 bits). Cantidad de bloques de 32 bits que componen la cabecera. Normalmente el valor es 5 (no se suelen usar opciones). Máximo: 15.
* **Tipo de servicio** (ToS, 8bits). (Este campo está en desuso, sólo tiene sentido en una red privada) Indica la calidad de servicio deseada durante el tránsito por una red. 4 bits tienen significado, los otros 4bits son reservados para el futuro.   
  De los 4bits solo uno podía estar encendido:
  + **Bit 0: Minimize Delay** (minimizar **retardo**, envíalo por el camino que tarde menos y priorizar sobre otros).
  + **Bit 1: Maximize throughput** (enviarlo por el camino que permita la mejor **velocidad** real de transferencia de datos).
  + **Bit 2: Maximize reliability** (maximizar la **confiabilidad**, enviarlo por el camino más confiable).
  + **Bit 3: Minimize cost** (enviarlo por el camino más **barato**).
* **Identificador único del datagrama** (16bits). Si se fragmenta el datagrama, esto permite distinguir los fragmentos de un datagrama de los de otro. El originador del datagrama debe asegurar un valor único para la pareja origen-destino y el tipo de protocolo durante el tiempo que el datagrama pueda estar activo en la red.
* **Flags** (3bits). Actualmente utilizado sólo para especificar valores relativos a la fragmentación de paquetes. Los 3 bits (por orden de mayor a menor peso) son:
  + bit 0: Reservado; debe ser 0.
  + bit 1: No fragmentar. 0 = Divisible, 1 = No Divisible (DF).
  + bit 2: Más fragmentos. 0 = Último Fragmento, 1 = Fragmento Intermedio (le siguen más fragmentos) (MF).
* **Offset - Desplazamiento del fragmento (13bits):** En paquetes fragmentados indica la posición, en unidades de 64 bits, que ocupa el paquete actual (offset de la data CREO) dentro del datagrama original. El primer paquete de una serie de fragmentos contendrá en este campo el valor 0.
* **Tiempo de vida (TTL Time To Live, 8bits).** Indica el máximo número de enrutadores que un paquete puede atravesar. Cada vez que algún nodo procesa este paquete disminuye su valor en, como mínimo, una unidad. Cuando llegue a ser 0, el paquete será descartado. Se utiliza porque es un protocolo no orientado a la conexión.  
  En la práctica representa la cantidad de saltos entre routers. En 8 bits, máximo de saltos: 255.
* **Protocolo (8bits).** Indica el protocolo de las capas superiores al que debe entregarse el paquete (TCP = 6, ICMP=1 o UDP = 17). Le indica a IP a quién debe dar el mensaje. Si viene de TCP, va a TCP. Etc. (entre Ethernet e IP había un método similar)
* Suma de Control de Cabecera (Checksum - 16bits). **Sólo controla errores en la cabecera (no los datos!)** Se recalcula cada vez que algún nodo cambia alguno de sus campos (por ejemplo, el Tiempo de Vida). Si encuentra errores, descarta el datagrama, sin generar mensajes de error.
* **Dirección IP de origen** (32bits) Debe ser dada en formato de red.
* **Dirección IP de destino** (32bits). Debe ser dada en formato de red.
* **Opciones** (bits variables): Las opciones están definidas y estandarizadas pero la gran mayoría están en desuso. Aunque no es obligatoria la utilización de este campo, cualquier nodo debe ser capaz de interpretarlo.
* **Relleno** (bits variables): Utilizado para asegurar que el tamaño, en bits, de la cabecera es un múltiplo de 32 y para llegar al tamaño mínimo del datagrama. El valor más usado es el 0.
* **Datos** (bits variables): El máximo, al igual que los demás variables, sale de restar la longitud total de la ocupada.

## Fragmentación y reensamblado

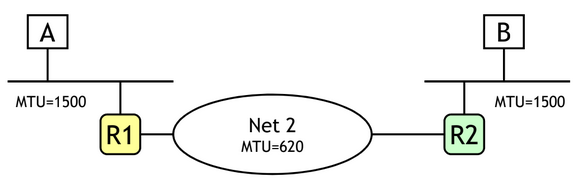
Cuando un router recibe un paquete, lo des-encapsula para leer la dirección destino y si no era para él, y **lo vuelve a encapsular pero en el protocolo/tecnología que tenga el enlace entre él y el router al que se lo debe pasar.** Cada tecnología tiene su propia **limitación respecto al tamaño del mensaje**, que es lo se conoce como **Maximum Transfer Unit (MTU).**

MTU (Maximum Transfer Unit): Cada tecnología de conmutación de paquetes, fija un límite máximo para la cantidad de datos que pueden transmitirse en una única trama.

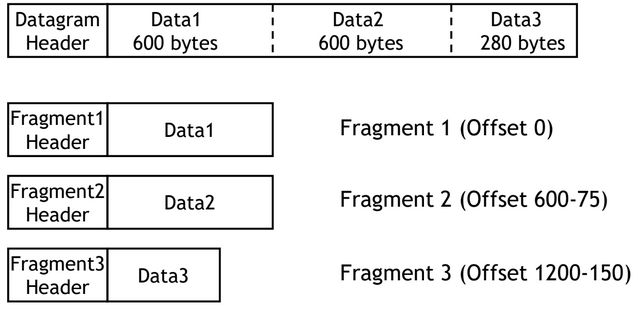
Cuanto más bytes puede enviar por mensaje, más eficiente es la transmisión (bits datos/ bits totales).

Si el router 1 que trabaja con IEEE 802.3, debe pasarle una trama de 1500bytes al router 2 que trabaja con X.25 (MTU menor), debe fragmentar esa trama en tramas más pequeñas.

### Funcionamiento



* “A” que está conectada a una interfaz con una MTU=1500. Arma un datagrama de 1500bytes, dirigido a B, lo encapsula en una trama Ethernet y se lo pasa a R1 (router).
* El router R1 lee la dirección MAC que sea para él, va a leer el EtherType (va a ver que es un datagrama IP), va a ir al campo de datos y va a retirar de ahí el datagrama IP y se lo va a entregar al módulo IP. El modulo IP va a calcular el checksum y va a ver la dirección destino y toma la decisión de por cuál interfaz lo envía.
* A continuación se encapsula el datagrama en el protocolo Net 2. Tiene un datagrama IP de 1500bytes y Net 2 tiene un MTU de 620bytes. **IP ofrece un mecanismo de fragmentación y re-ensamblado**, por lo que R1 va a fragmentar el mensaje.
* IP oculta los detalles de tecnología subyacente. **Divide los datagramas en fragmentos que deben ser re-ensamblados en la dirección destino.**



La fragmentación la realiza la capa IP (puede hacerlo un router o un terminal). Esto lo hace el primer elemento de red que tiene el problema de un mensaje demasiado grande para enviar (R1 en este caso)

El reensamblado lo realiza el destino final en la capa IP.

El fragment header contiene:

* **Dirección origen y dirección destino**. Es el mismo que el del datagrama original.
* **Campo identificación**: todos los fragmentos tienen el mismo para ser identificados como una misma unidad. Es el mismo que el del datagrama original.
* **Campo desplazamiento del fragmento** (offset): con el offset reconozco el orden de los fragmentos. Como el tamaño del datagrama original es demasiado grande, no puedo tomar el offset real, por lo que se elige un múltiplo de 8bytes más próximo al MTU del trayecto (en el ejemplo, 600 se representa con 600/8 = 75).
* **Flags.**
  + Bit 1: bit sin uso.
  + **Bit 2: no fragmentar**. Si se encuentra en 1 impide la fragmentación, por lo que si no es posible pasarlo por el MTU se descarta y se genera un mensaje ICMP (reporte de error al origen que necesita fragmentar y no lo dejan).
  + **Bit 3: más fragmentos**. Si esta en 0 es el último. Si se fragmenta un datagrama con el bit de más fragmentos en 1, ninguno de sus fragmentos va a tener el bit en 0 debido a que ese datagrama también era un fragmento.

**Cada fragmento conforma un datagrama independiente** para la red, es decir que si un router fragmentó, **el router destino no se entera**. Los tres conforman un datagrama fragmentado solamente para el destino, que es el que lee el bit “más fragmentos”.

Las 3 cabeceras tienen elementos en común:

* IP destino y origen
* El campo identificador, que vincula a los tres entre sí.
* Flags. El flag “Más fragmentos” va a estar en 1 en todos menos el último.

Offset nos van a permitir reensamblar los 3 pedazos, respecto de su posición original en el datagrama. El primer datagrama tiene offset 0.

El valor del offset no es absoluto, sino el real / 8. Así ocupa menos espacio. 🡪 por eso el tamaño de fragmento tiene que ser un múltiplo de 8, que se elige el más cercano (menor) al MTU.

Desventajas de la fragmentación

* Duplica la probabilidad de pérdida de un datagrama.
* Genera mayor carga de procesamiento en los routers.

El proceso de fragmentación se puede realizar múltiples veces sucesivas.

# Direccionamiento

Un protocolo de capa 3 debe ser capaz de proveer direcciones que permitan identificar a los miembros de la red y mecanismos que permitan el encaminamiento (es decir cómo llegar de un nodo a otro).

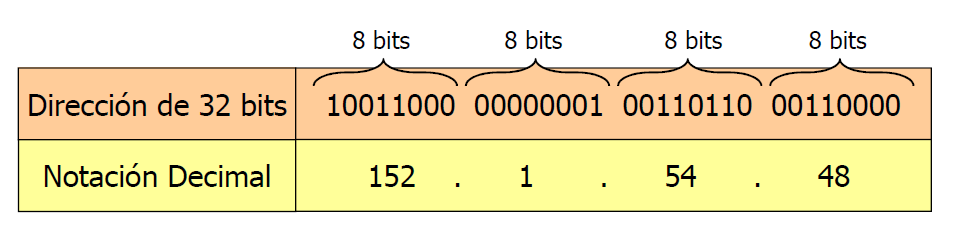
Son funciones del protocolo de red:

* Direccionamiento. Cómo identificar cada punto de la red (origen y destino).
* Encaminamiento. Cómo llegar de un punto a otro. Lo realizan los routers.

## Direcciones y máscaras

En el protocolo IP a cada host se le brinda una dirección IP con las siguientes características:

* Es **única** en internet, no hay dos direcciones IP iguales. No hay ambigüedad.
* Tiene 32bits de **longitud** (1 palabra en la cabecera IP).
* Se suelen representar como **4bytes separados por un “.”** con notación decimal (ejemplo 24.323.218.197). Cada grupo es 1 octeto, por lo que cada número está entre 0 y 255.
* Parte de la dirección IP identifica a la red, la otra parte identifica al host dentro de la red.
  + Identificador o prefijo de red: parte de la dirección IP que es igual para todos los miembros de la red. Los proveedores de red deben pedir este prefijo al organismo que los gestiona.
  + Identificador de host: parte de la dirección IP que permite identificar unívocamente al host dentro de la red. Este se solicita a la red misma. Están reservados los siguientes valores:
    - Todos 0: identificar a la red.
    - Todos 1: reservado para broadcast en la red.
* La **netmask o mascara de subred** indica qué parte de la dirección corresponde a la red y al host. La máscara tiene la misma longitud que la IP. Un bit igual a 1 significa que ese bit de la dirección IP corresponde a la red y un 0 al host. (255 son 8bits en 1. Ejemplo 255.255.255.0, recién el ultimo octeto o separación corresponde al host).  
  Siempre empieza con 1 la máscara. Los 1 tienen que ser consecutivos (no se pueden alternar números).   
  Estos números son compartidos por cada elemento de mi red.   
    
  Máximo de hosts identificables = (2^[cant de 0s en máscara]) – 2



**Ejemplo de máscara:**   
Máscara de subred 255.255.255.0 🡪 esto significa que sólo 8 bits son dedicados a identificar dispositivos en mi red. El host 0 no se puede usar, por convención, porque representa a la red como un todo. El host 11111111 representa a todos en la red, es un broadcast.

Por ende, con esta máscara de subred sólo puedo tener 2^8 – 2 dispositivos = 254. Entre estas 254 está incluido el router, como cualquier otro dispositivo conectado a este prefijo (Smart tv, impresora, tele, router, etc).

**Cómo funciona:**

Si el host quiere enviar un mensaje a una dirección destino que se encuentra en su misma red, va a averiguar la MAC address del destino, va a armar una trama de Ethernet con las direcciones MAC address e IP y va a enviar directamente el mensaje.

En caso de que no este en la misma red, no puedo averiguar la dirección MAC. Por ende, el host debe enviar el mensaje al router:

* Para ello el host averigua la dirección MAC del router, encapsula el datagrama IP que le quiero enviar al destino, dentro de una trama Ethernet cuya dirección MAC address es del router (que tiene una dirección IP).
* El router recibe la trama Ethernet con su dirección MAC como destino. El EtherType es 0800 por lo que lo hay adentro es un datagrama IP. Lo analiza si no está en su red y envía el mensaje por el camino donde se debería encontrar la dirección destino.
* El destino al recibir la trama, responde a la IP address del host si era una request a gmail por ejemplo, sino no hace nada debido a que TCP/IP no es orientado a la conexión.

# Subredes

## Definición

Las **subredes** son un método para **maximizar el espacio de direcciones** IPv4 de 32 bits y **reducir el tamaño de las tablas de enrutamiento** en una interred mayor.

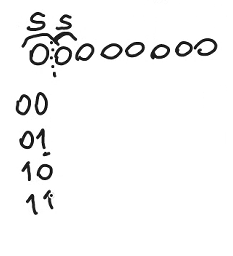
En cualquier clase de dirección, las subredes proporcionan un medio de **asignar parte del espacio de la dirección host a las direcciones de red**, lo cual permite tener más redes. La parte del espacio de dirección de host asignada a las nuevas direcciones de red se conoce como **número de subred**.

## Cómo funciona

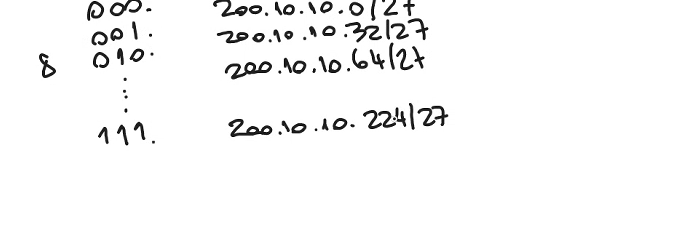
**Problema inicial:** Si tengo un rango de direcciones IP asignadas, pero necesito subdividirla en 2, tendría que cambiar la máscara de subred y darle la mitad del espacio a cada red. Esto lo haría con un único bit del espacio de hosts, y lo sumo a la máscara.

Cada una de estas redes tendría su broadcast (todos 1) y su dirección de red (todos 0)

Si necesitara más subdivisión, tendría otro problema. Podría aumentar la cantidad de bits usados para subred. Con 1 bit puedo tener 2 subredes. Con 2 bits, puedo tener 4.



|  |  |  |
| --- | --- | --- |
| Bits | Subredes | Hosts |
| 1 | 2^1 | N/2 – 2 |
| 2 | 2^2 | N/4 – 2 |



## Por qué surgen

Cuando se pide un prefijo de red a la IANA (internet assign Number authority) entidad que las asigna), la misma brinda el prefijo de red. Que las direcciones tengan el mismo prefijo de red significa que desde la .1 a la .254, los hosts son vecinos (conectados al mismo enlace físico).

Esto podría no suceder, por ejemplo en el caso de que quiera tener una red local y una remota (IANA no va a proveer a la misma empresa otro prefijo de red porque todavía tiene un montón de IPs de la anterior sin utilizar). Para solucionar este problema se “parte” la red para poder implementar dos redes, es decir obtengo subredes.

Eventualmente, IANA se queda corta. RFC define estándar para asignación de direcciones.

## Cantidad de subredes

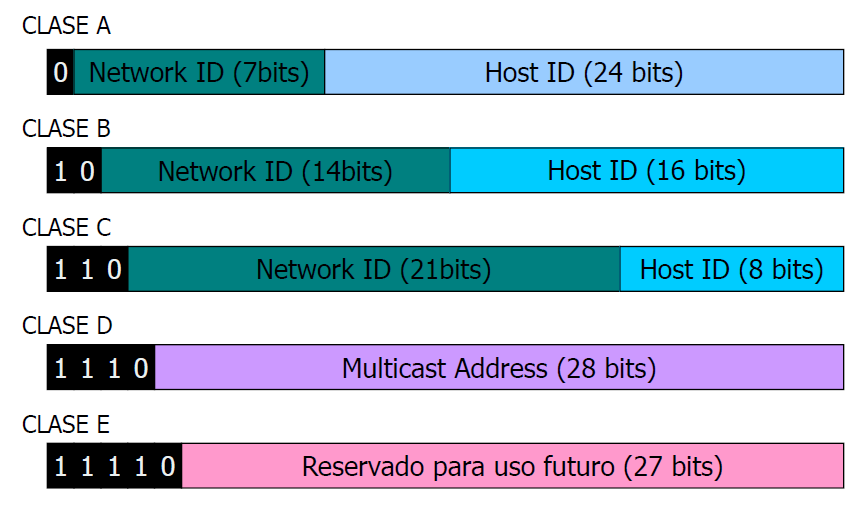
Cuando hacemos una partición la **mínima cantidad** de bits que puedo agarrar para formar subredes es **1bit**, formando **2 subredes** pero, por **estándar** (creo que igual ya no se le da bola al estándar), la **mínima cantidad** de bits del ultimo octeto que podemos agarrar para formar subredes es **2 bits**, por lo tanto obtenemos **4 subredes** (las que empiezan con 00 01 10 y 11) . Si tomo 2bits como mascara de subred, me quedan 6 bits para las direcciones que serían 64 direcciones, de las cuales puedo asignar 62 por subred (**la primera es de la subred y la ultima de broadcast**).

Para formar estas mascaras de 26bits (24 de red y 2 de subred), debo modificar la máscara de la IP a 255.255.255.192 (192 es 8unos.8unos.8unos.2unosy6ceros).

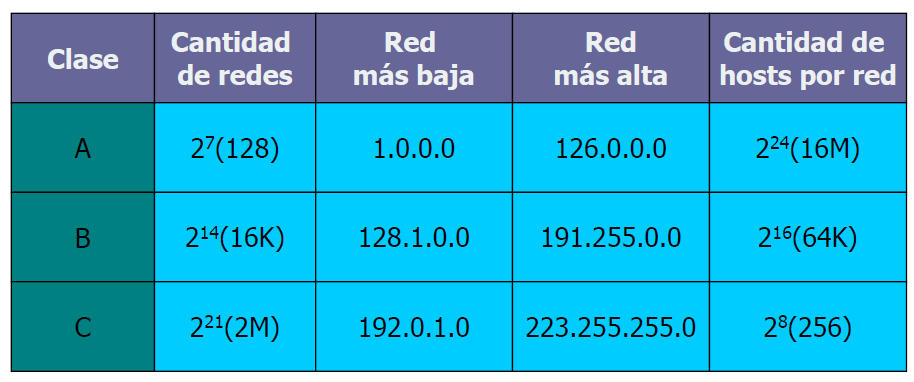
Generadas las subredes, el direccionamiento es el mismo que el explicado anteriormente, se tratan como dos redes distintas.

# Clases de direcciones

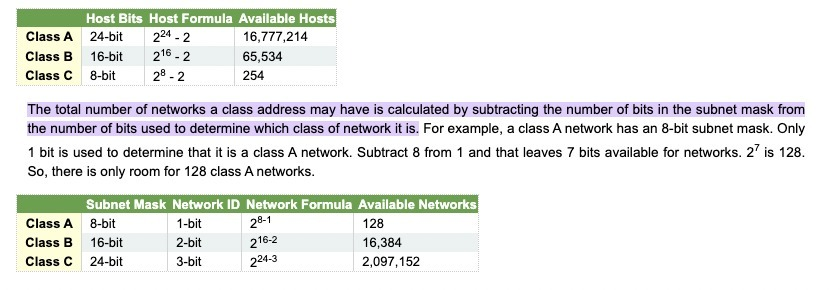
Para mejorar el uso de la tabla de direcciones, la IANA decidió separa los prefijos de red en bloques A, B, C, D y E, siguiendo un orden de mayor soporte de host a menor.



Las tres clases que se pueden utilizar **para host son la clase A, B y C**, mientras que la **D está reservada para multicast y la clase E para un uso futuro.** La capacidad de soporte de host es la siguiente:



Otro cálculo más exacto de hosts (notar que resta 2 a la cantidad de hosts, teniendo en cuenta red y broadcasting) y redes



Debido a que inicialmente las personas solicitaban IPs no para conectarse a internet, sino porque utilizaban **programas que corrían sobre el protocolo IP (uso privado)**, surgió el **RFC1819** para mejorar la asignación de las direcciones.

Para que dos direcciones IP estén dentro de una misma red, su prefijo debe ser igual.

* Clase A. redes muy grandes, con muchos nodos. Máscara 255.0.0.0 (primer octeto identifica la red, el resto identifica el host). Son muy pocas redes 2^7 (por qué? Porque el primer bit se usa para indicar que es tipo A?) y muchos hosts.
* B: 255.255.0.0
* C: 255.255.255.0

## RFC1819 classfull – Bloques reservados para uso privado

Respuesta de la IANA en los 90 cuando se empezó a quedar sin direcciones. También se diseñó IPv6, aunque no se implementó mayoritariamente (hoy en 2022 sólo algunos proveedores las brindan, en Arg hace 4 o 5 años).

### Dirección privada

Se reservó **un bloque A, uno B y uno C para direccionamiento privado** con los cuales **no se puede acceder a internet** porque **NO garantizan la unicidad** (pero si debe ser única cada IP dentro de la red privada), a muchos se les asigna esos bloques.

Esos bloques reservados para uso privado son:

* Clase A: 10.0.0.0 a 10.255.255.255 (10/8 prefix).
* Clase B: 172.16.0.0 a 172.32.255.255 (172.16/12 prefix).
* Clase C: 192.168.0.0 a 192.168.255.0 (192.168/16 prefix).

Estas direcciones no se pueden usar para comunicarse con internet. La mayoría de los routers están configurados con una IP de este rango.

### Dirección publica

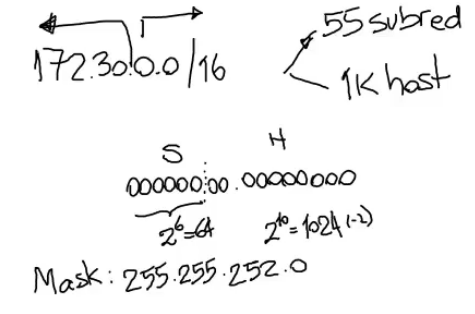
A los routers se le asigna una dirección pública, es decir con posibilidad de acceso a internet debido a que no existe la duplicidad. Esta dirección del router nos la asigna (presta) nuestro proveedor de internet que a su vez pidió a la IANA un bloque para sus clientes.

Accedemos a internet a través de la IP pública del router. La trama viene del dispositivo con la dirección IP privada del mismo, cuando la recibe el router, reemplaza esta dirección privada de mi dispositivo por la IP pública del router y envía la trama. El router se guarda el registro de esta operación para que, si recibe una respuesta, envié la misma al host correspondiente.

### Dirección loopback

El **bloque 127.0.0.0** de la clase A esta reservado para looping, es decir, **todo lo que se le envié a ese puerto, me lo devuelve**. También se conoce como **localhost**.

Ejemplo de ejercicio subred



# Classless interdomain routing (CIDR) y VLSM

**Notación CIDR:**

Ejemplo: IP del formato 200.10.10.0/24 🡪 /24 equivale a máscara 255.255.255.0 (24 unos, y 8 ceros)

La notación me indica la cantidad de 1 que hay al principio de la máscara.

Permite distinguir prefijo de red de dirección de host.

**Esto venía del resumen de antes:**

Intenta resolver el problema de tamaño fijo de los bloques de las clases.

Consiste en olvidar la classfull (la división por clases) y poder darle a un cliente la cantidad de direcciones que pida asignándole un segmento de ese tamaño (por ejemplo si pide 1000 direcciones, se le asigna una dirección de red de /22 quedando 10 bits para los hosts). Es decir, consiste en mascaras de tamaño variable (VLSM variable length subnet mask).

Se implementa este método para todos los bloques restantes (libres) a los ya asignados con el método anterior.

# Conceptos de enrutamiento (para clase que viene)

Default Gateway: normalmente es el router. Máscara y default Gateway son conceptos de ENCAMINAMIENTO, no de direccionamiento. Me dice dónde estoy y cómo salgo.   
La dirección del Gateway puede ser cualquiera, pero es práctica común que sea la primera (o menos seguido, la última) del rango de esa red.

Si se configura mal el Gateway, no se puede salir de la red y conectarse a otras (internet). Idem con la máscara.

# Actuaildad, comentario conflictivo

Hoy en día, el mundo es classless (no tengo clase A, B, C, D).

No se puede hacer un broadcast fuera de tu red. Sólo se puede enviar a los vecinos de red. Las redes lo descartan.

(Creo uq ese refiere mandar un broadcast a una suprared de mi subred, que no puede englobar más cosas).

En wireshark esto se ve porque ningún mensaje va a 192.168.0.255 que venga desde afuera de la red .