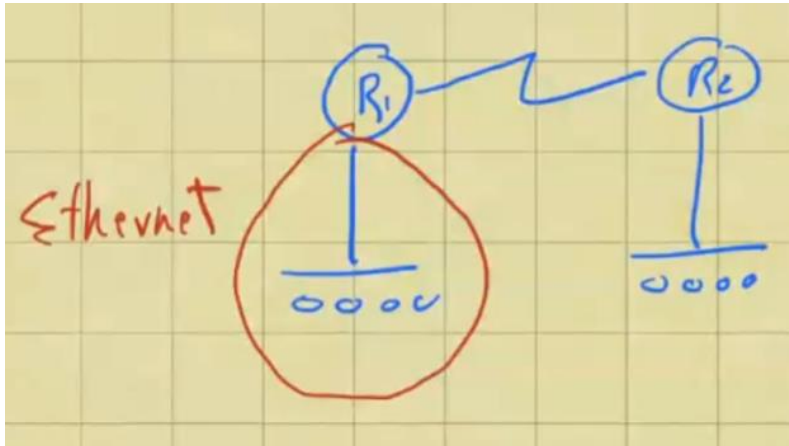


El profe nos va a mostrar conectividad

Mismo esquema de los tp

Ruteador = R , se conecta a serie de redes lan

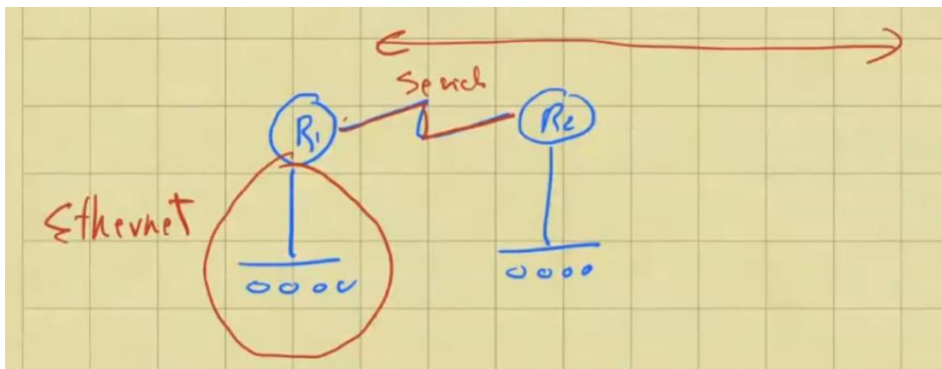
Se conecta a otro, R2, Que se conecta a una LAN .



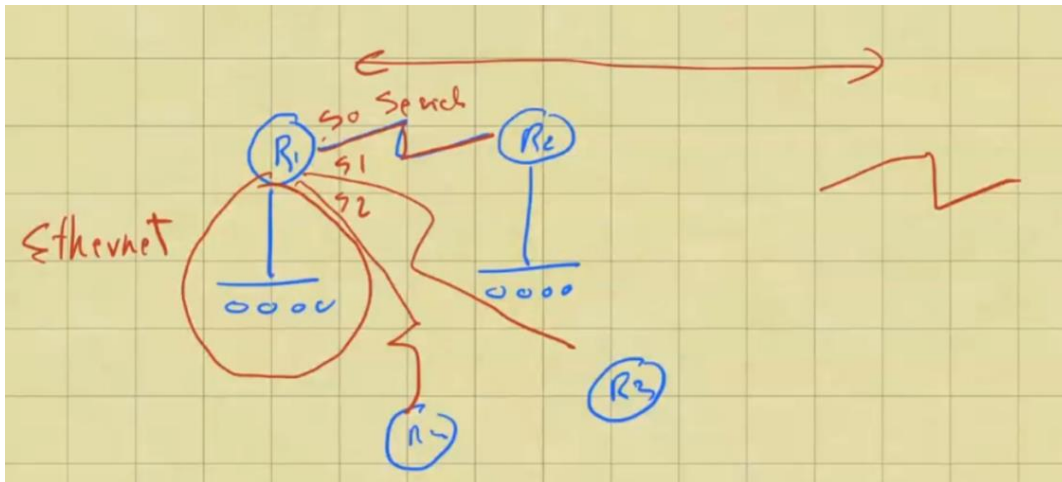
La red LOCAL es una red ethernet .

Se conecta con otro ruteador que está LEJOS (o cerca )

Entre ambos ruteadores se tira un cable llamado SERIAL



El ruteador no hace falta que este solo conectado al 2, sino a cualquiera, r3, r4, etc etc. C/u de ellos , con un enlace SERIAL ( Numero 0 , 1 , 2 .... ) Hay que acordarse cual es .

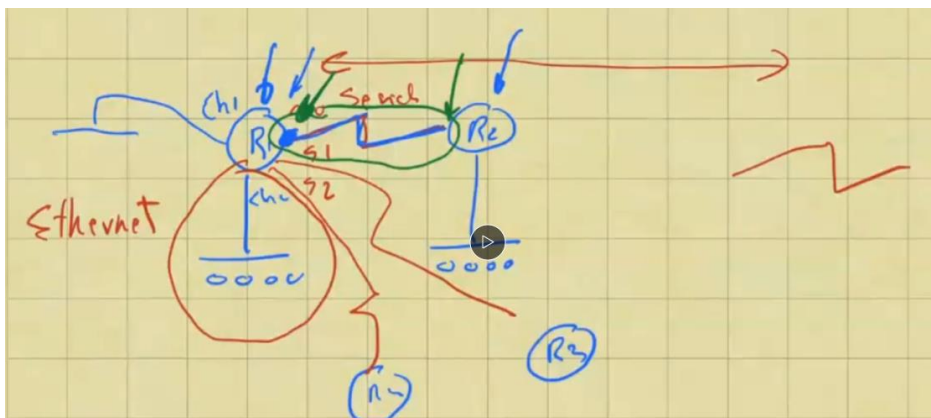


Ethernet tambien puede tener muchas salidas, eth 0, 1, 2 y 3.

Los seriales tienen que tener un clock , como característica . Alguno de los 2 ruteadores tiene que tener un clock , dice a que velocidad trabaja el enlace . Cada router tiene su propio clock .

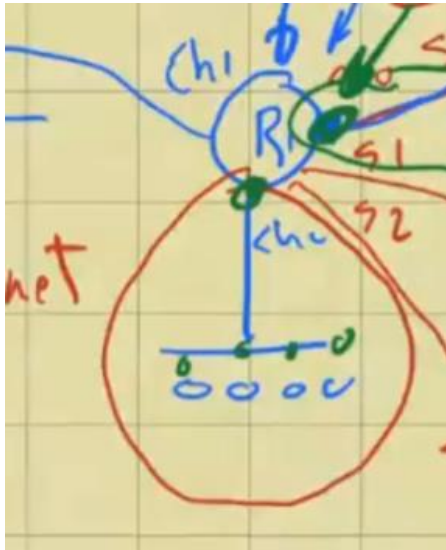
Una de las maquinas., cualquiera de ellas, da el clock para el enlace , la otra adapta su clock a este enlace de lo que le mandó la maquina correspondiente.

El serial tiene una característica especial que tiene un clock puesto en alguno de los 2 lados .



Veremos que un serial va a tener una dirección a la izquierda y también de la derecha, harán falta 2 direcciones .

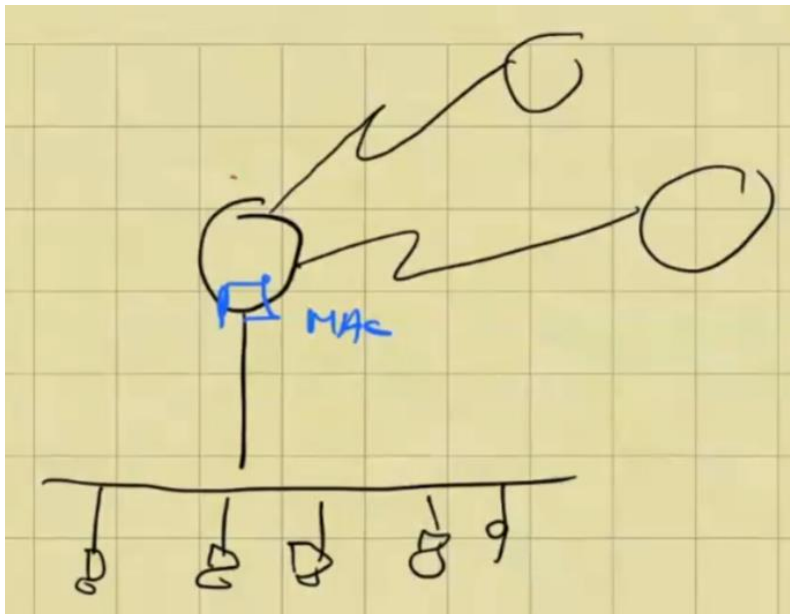
En eth, tenemos que tener una direccion de centro, y de cada una de las máquinas que esten conectadas .



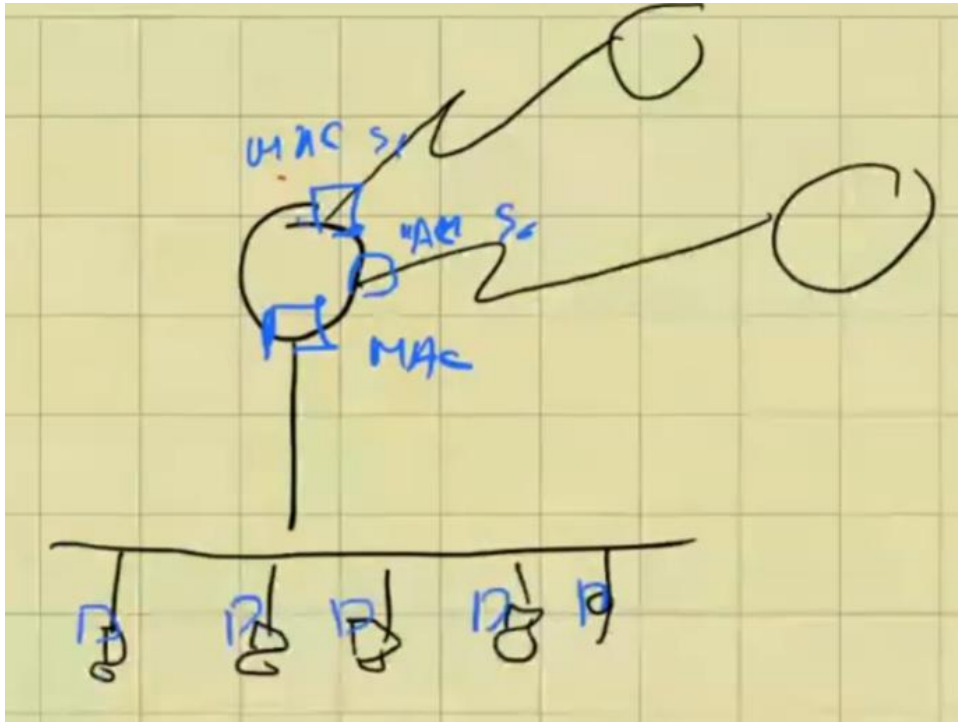
Los seriales no vienen con direcciones establecidas , se las tenemos que asignar nosotros, como hacemos en el Boson .

Lo unico que ya viene “asignado” con direcc preestablecidas son las placas de red que hay en los dispositivos, como en los routers o en las máquinas .

Tenemos un router conectado a una red LAN que conecta a varias máquinas . Conectado a su vez a otros routeadores



Tenemos placas de RED, en el router una placa con dirección MAC que tiene el router en su enlace ethernet , una placa de RED contiene su dirección MAC a su enlace SERIAL1, una placa de red que tiene su dirección MAC a su enlace SERIAL0



Configurar la interface (Boson) : cuando configuramos una “interface” con los datos pertinentes, necesitamos PRENDERLA para que ejecute sus datos . ( no shutdown ) . No hay que olvidarse de esto .

La interfaz es la configuración LÓGICA que tiene el puerto , cuando le ponemos las direcciones IP y las máscaras .

Que hacen las máscaras ?

Tenemos una dirección IP, por ejemplo:

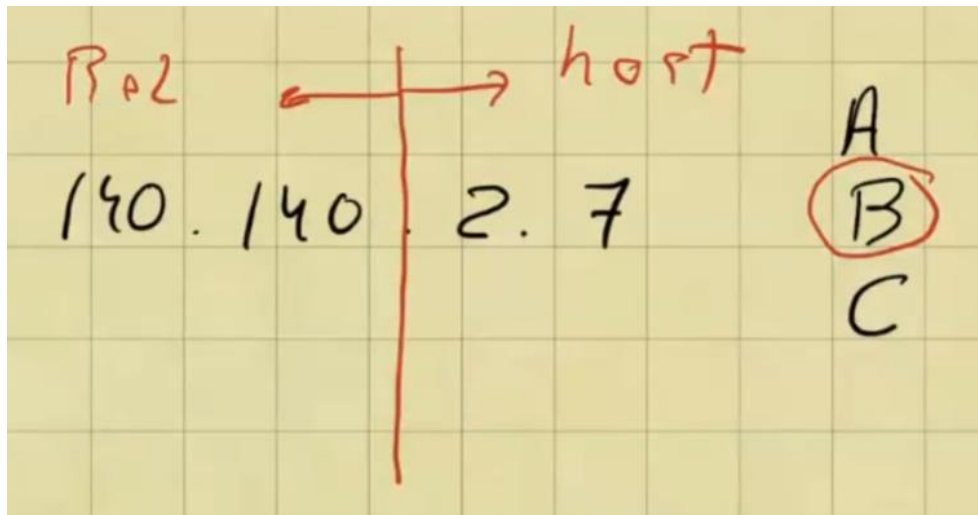
140.140.2.7

Puede ser clase A, B o C.

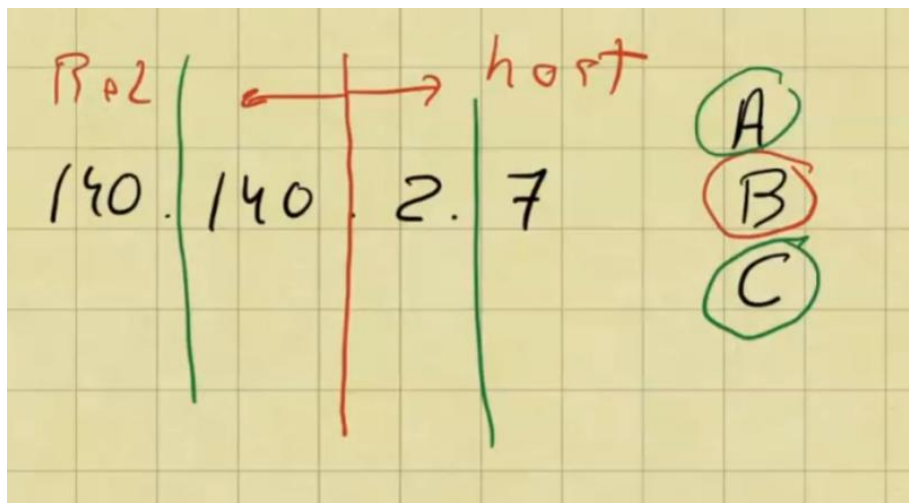
En este caso ,por ser 140 (empieza con 1), es una clase B.

Por ser clase B, se divide así :

De forma tal que los primeros 2 pares de octetos sean direcciones de RED, y los últimos 2 pares de octetos sean direcciones de HOST .



A y C se dividen así :

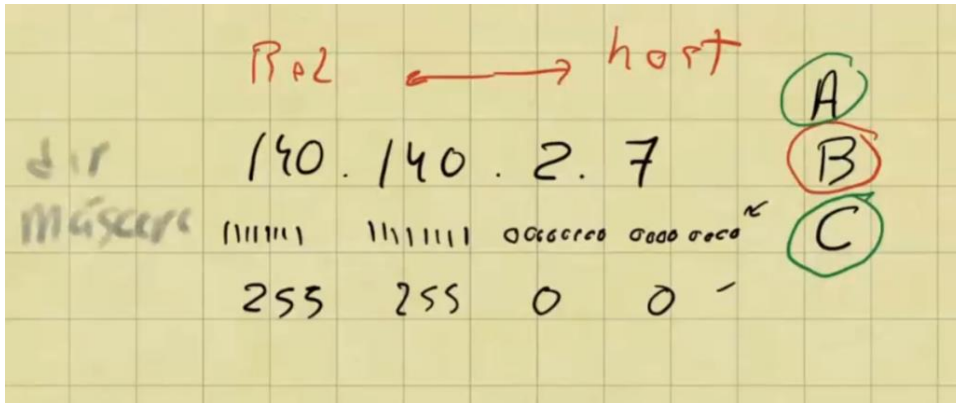


A = Primer par de octetos RED, 3 pares siguientes HOST ,  
C = 3 pares de oct RED , 1 par HOST

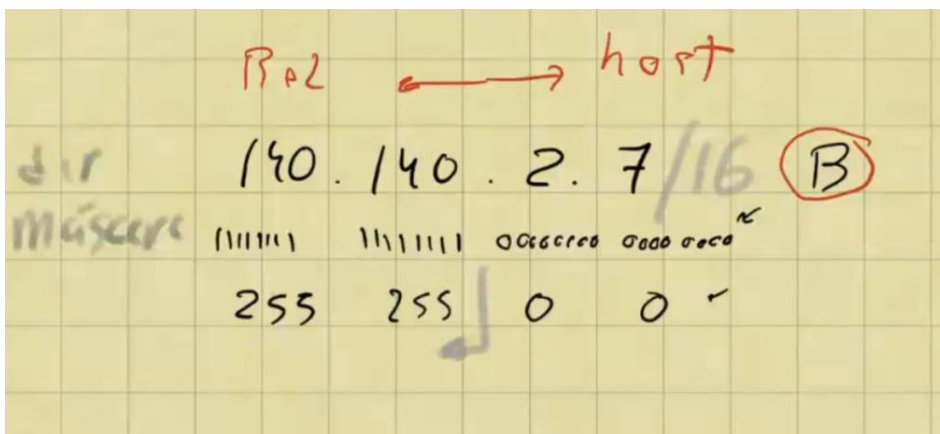
No había una división fácil para hacerlo, entonces se quiso a empezar a manejar un poco más en detalle, se pensó en la MASCARA.

La mascara es un num que va junto con la dirección IP que como es binario puede ser 0 o 1, cuando es todo 1, sería :

Cuando es UNO, significa que es RED, cuando es CERO, significa que es HOST :



Es una de la formas con las que podemos escribir, otra de las formas es con notación de barra, es decir contar cuantos 1 hay de aquí para la izquierda, en este caso 16 UNOS, entonces sería 140.140.2.7/16 (dirección IP / bits )



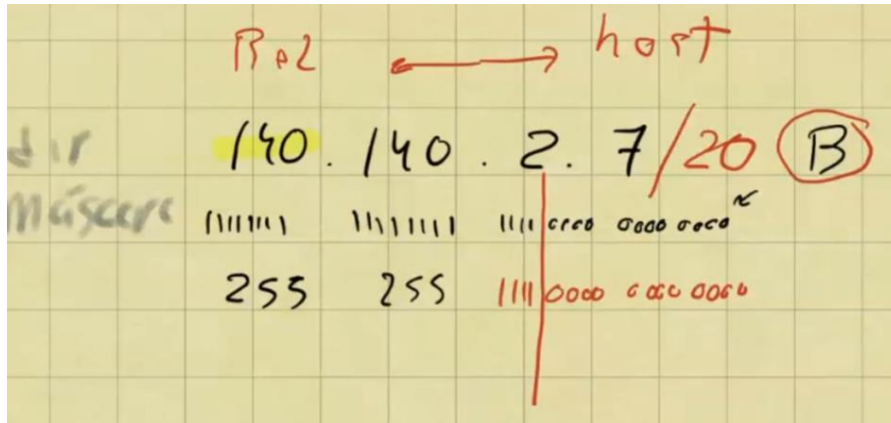
Tenemos 16 posiciones correspondientes a la RED desde la izquierda , y desde la derecha 16 posiciones correspondientes a HOST .

Porque nos damos cuenta que es clase B? Pq comienza con 140, pero también nos damos cuenta porque tenemos que la máscara divide en unos y ceros, y porque tenemos un /16 también.

Ya no me hace falta definir de esa manera, hoy en día puedo definir otro tipo de máscara, que puede ser así :

La notación de barra caería ahora acá , donde el tercer octeto comparte 4 unos y 4 ceros y NO 8 ceros .

Esta máscara ya no sería 255.255.0.0 y /16, sería :



Serían ahora 20 posiciones ,  $8 + 8 + 4$ .

Que gano con esto? Que ahora, ya la division entre RED y HOST no es justo en el medio, sino que un poquito distinta.

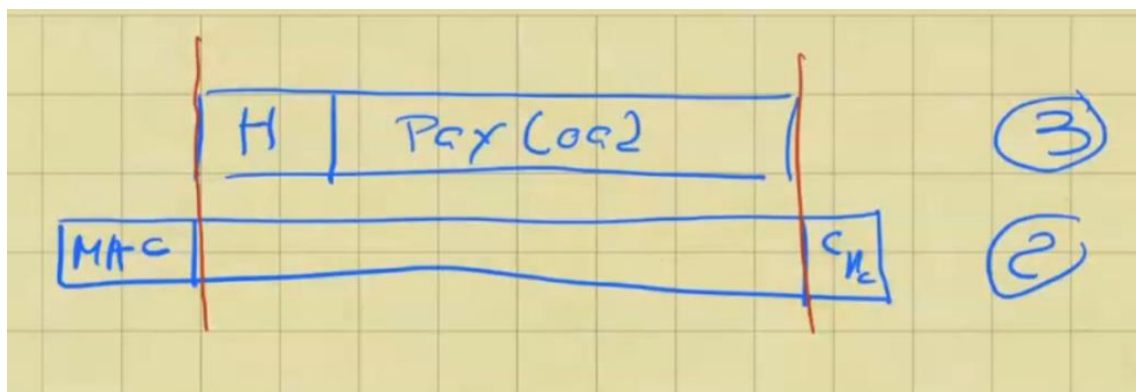
Deja de ser A , B o C, pierde sentido hablar de clases, ya que yo puedo manejarlo con distintas config de parte de red y host . SERIA UNA RED "CLASSLESS" SIN clase . A diferencia de la CLASS FULL que es A, B o C.

## IP

Dentro del IP , tenemos :

Un HEADER y un PAYLOAD (capa 3)

Debajo de esto, está la TRAMA (CAPA 2) : Aquí estarían las direcciones MAC origen y destino, al final estaría el CRC . Lo de CAPA 3 se encapsula aquí dentro :

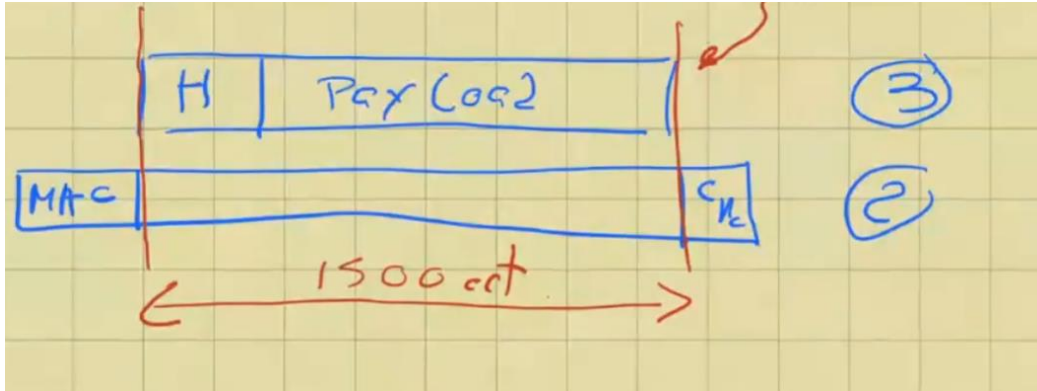




Maximo tamaño de una trama ethernet en su carga/payload = 1500 OCT .

El paquete IP puede ser MUCHO más grande , y como sucede eso, NO entra y hay que FRACCIONARLO .

Se transmite lo mismo en varios fragmentos.

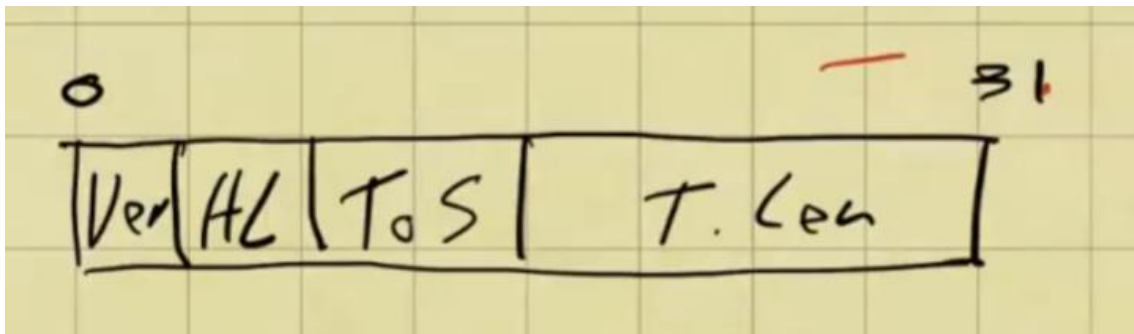


Cuando uno fracciona esto, genera fragmentos, cada fragmento tiene un HEADER, ese HEADER mide ALGO.

Si vemos el campo de HEADER de cada fragmento, se divide en:

VERSION | HLEN | ToS | TLEN

El campo VERSION tiene 4, el campo HLEN que tiene 4, el Type of Service que tiene 8, y el Total Length 'TLEN'.

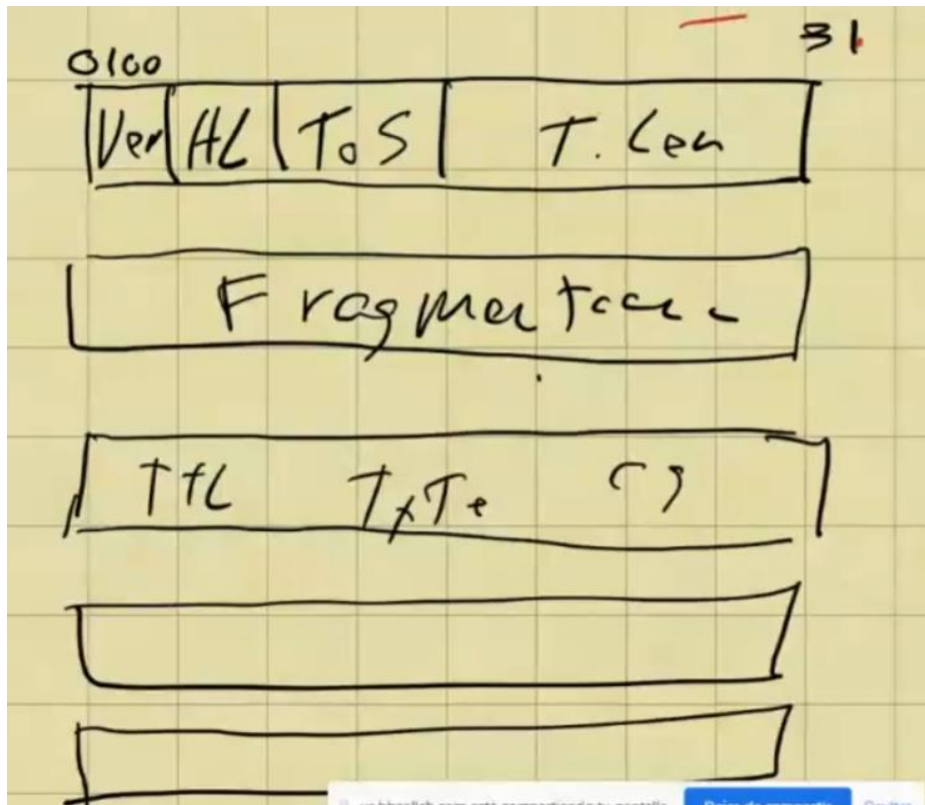


Esto va desde el bit 0 al bit 31, en total son 32 bit.

VERSION: La versión que nosotros tenemos (versión 4 : 0100 en binario)

HLEN: El tamaño que tiene el HEADER. El Header tiene un tamaño mínimo que es de 5 líneas, en las que va este campo HL (primer línea), luego viene el campo de la FRAGMENTACION (2da línea) y debajo de esto vienen los otros 3 campos que serían TTL, Type y HEADER CHECKSUM (3era línea)



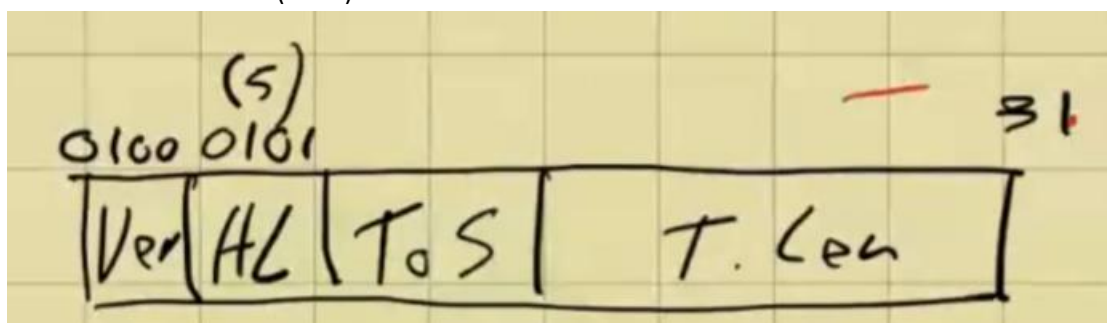


Debajo de esos 3 vienen las direcciones, origen y destino (ultimas 2 líneas que conformarían las palabras MINIMAS de un HEADER).

Esto sería mas o menos como está armado un datagrama IP que no tenga OPCIONES . Que significa? Yo le puedo dar muchas más cosas que se ponen abajo como opciones.

El HL mide en PALABRAS DE 32 BITS cuando MIDE EL HEADER (Lo primero que vimos con el payload).

Como mínimo, ya hemos dicho, tiene 5 palabras de 32 bits. Es decir, adentro de HL va el número 5 en BINARIO (0101)

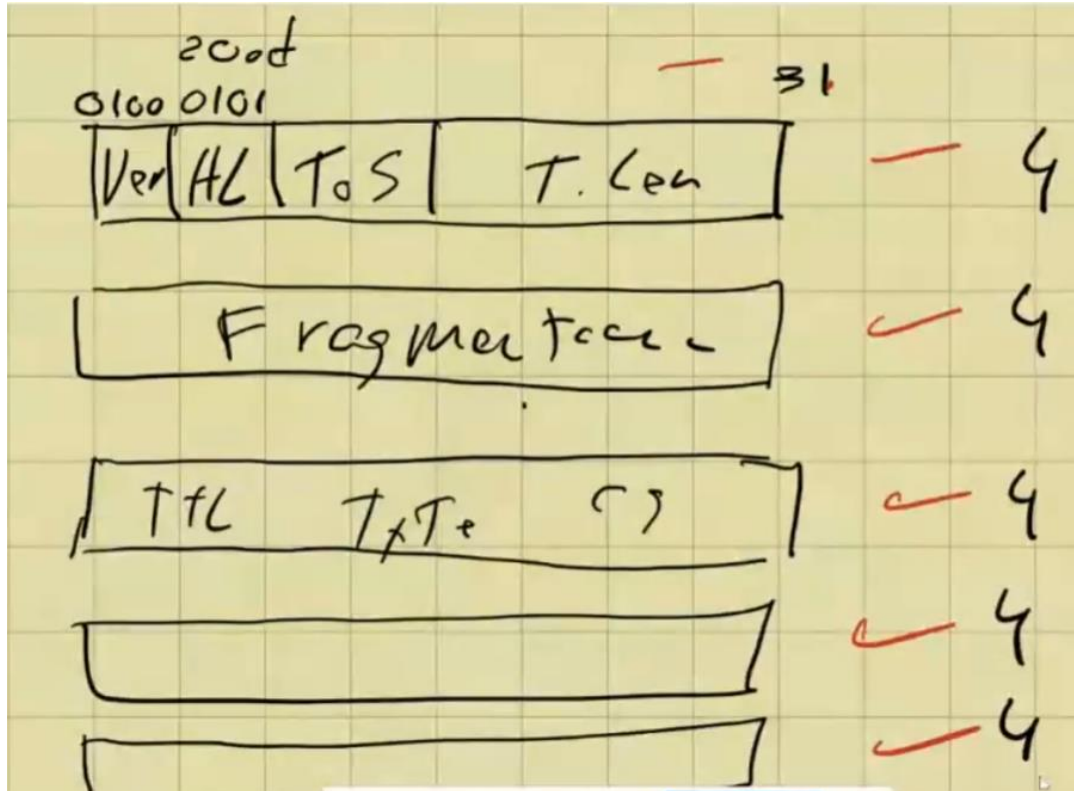


Entonces, 5 palabras de 32 bits,

Recordar que cada "LINEA" tiene 4 OCTETOS (PORQUÉ? PORQUE SON 32 BITS ,  $4 \times 8 = 32$ )  
cada OCTETO tiene 8 BITS,

entonces  $5 \times 4 = 20$  OCTETOS.

(NORMALMENTE, la gran mayoría de las veces, o casi todos, tienen el MÍNIMO, y entonces va siempre 20)

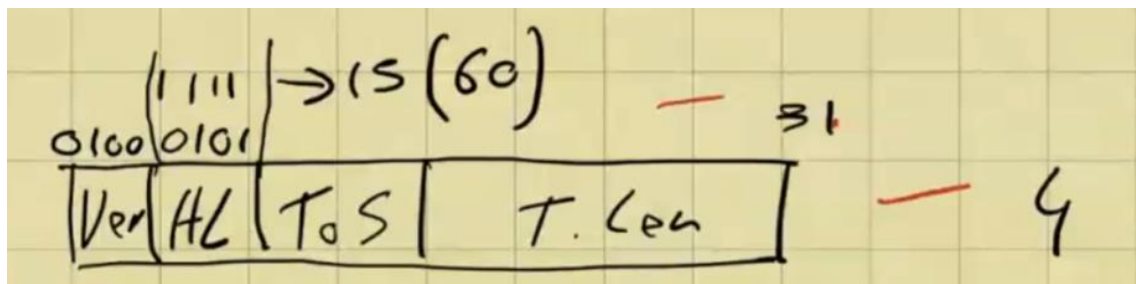


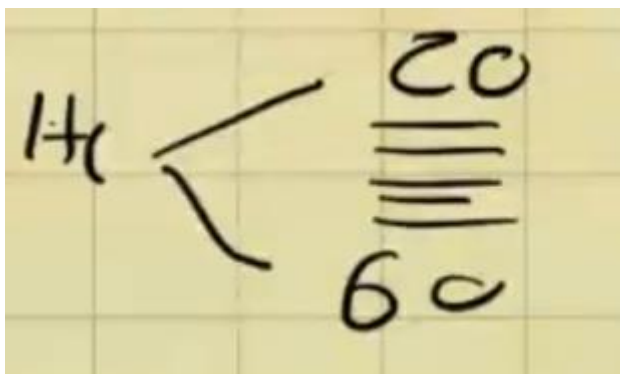
De que manera, calculandolo, podemos saber el máximo tamaño que puede medir entonces en octetos un HEADER? (HLEN)

Si 0101 es el mínimo ,el máximo a nivel binario es 1111 (15 palabras, de 32 bits)

Cada palabra tiene 4 octetos, entonces sería un máximo de 60 OCTETOS para el campo HLEN.

El Header Length puede ir entre 20 y 60 octetos.





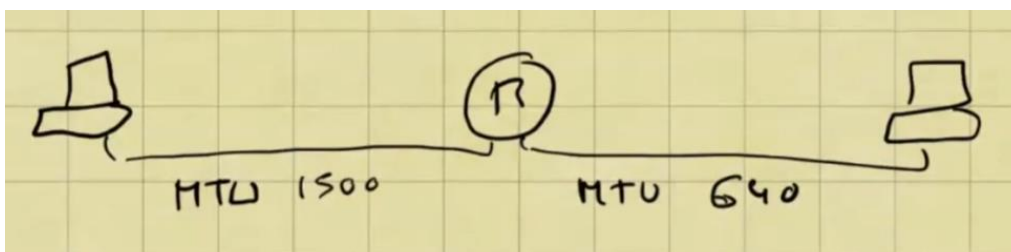
En la gran mayoría de los casos, si no se dice nada, usamos el valor 20.

Hay un pibe que preguntó con respecto al header y al payload del direccionamiento IP :  
Preguntó como hace el encabezado MAC para guiar a la trama a que haga trabajar su dirección IP :

Tenemos una máquina, un router y otra máquina, conectadas mediante un enlace.

Esos 1500 octetos máximos de la trama que comentamos antes se lo conoce como **MTU (Maximum Transfer Unit)**

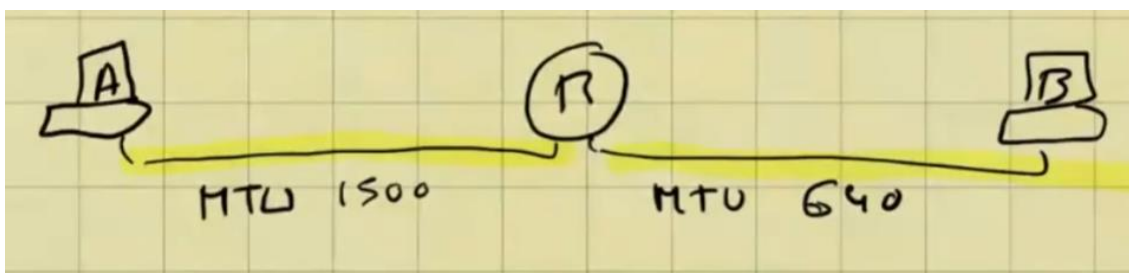
Tenemos un MTU de 1500 en la primer RED, y un MTU de 640 en la segunda RED:



Supongamos que los paquetes irán de la máquina A a la B.

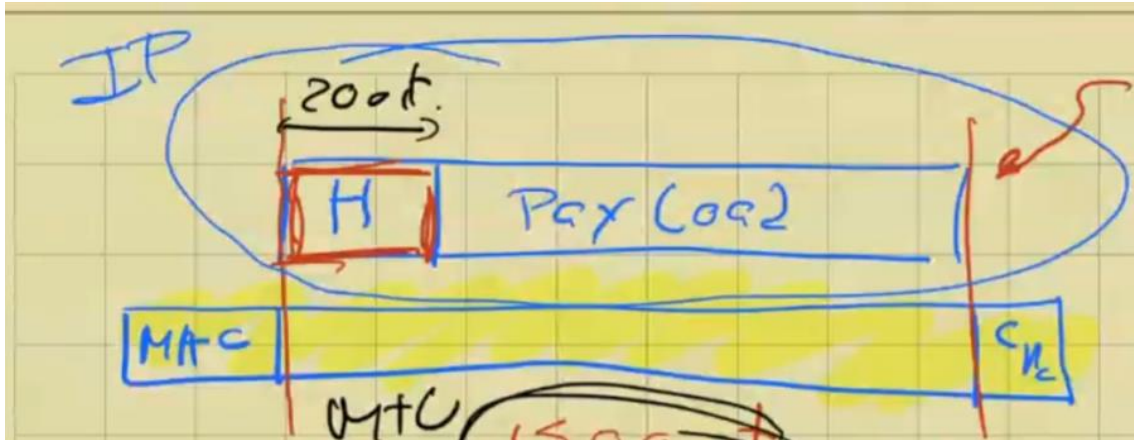
¿Cuál sería el procedimiento?

Tenemos la dirección MAC, que afecta al ENLACE LOCAL. Solo tiene validez e importa, en la dirección local, acá (sombreados con amarillo):



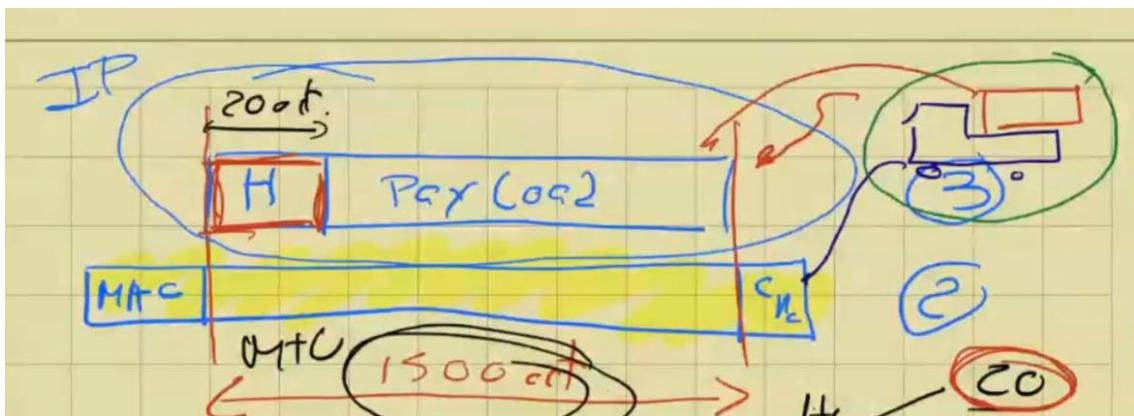
Veremos la primera

Uno, la trama, que sería lo que vemos abajo, como amarillo, es la direccion MAC



A la direccion MAC que está en la trama, la trama la pensamos como el camioncito que lleva las cosas, adentro lleva el PAYLOAD del IP.

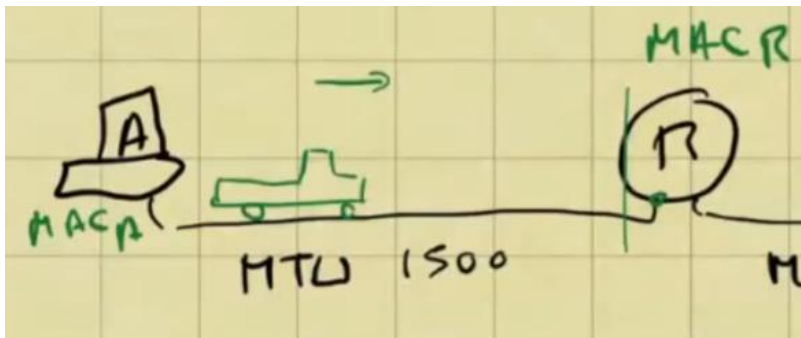
Osea, la primer parte la tomamos como un camion que lleva containers , que sería la trama (lo de abajo) y que arriba del mismo va el datagrama IP que sería el de arriba .



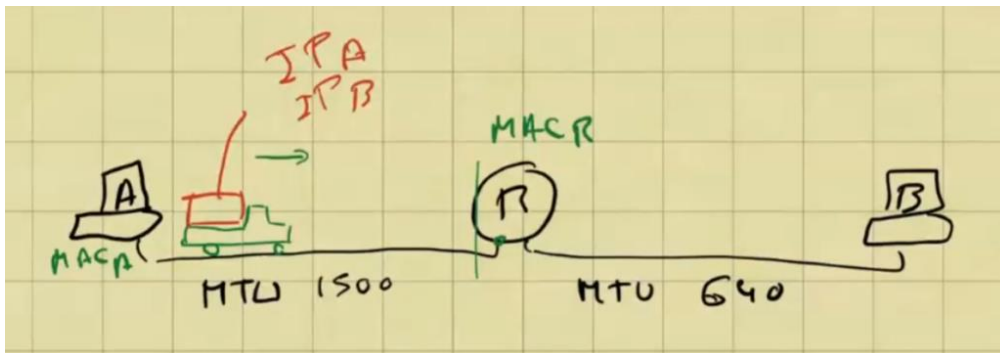
**(PUEDE VIAJAR ÚNICAMENTE EN LOS ENLACES DIRECTOS LOCALES, NO ATRAVIESA ROUTERS)**

Entonces, veremos como se arma esto.

Armamos el camioncito que fue iniciado por la trama de la direccion MAC A (MAC ORIGEN) y tiene como MAC DESTINO la MAC del ROUTER:



El camion tiene el paquete de las direcciones IP, Tiene la dirección IP de A como origen y la dirección IP de B como destino.



Nótese que la trama tiene direcciones de enlace local PURO y la dirección IP tiene las de origen a destino ya definidas.

Me quedé en 1:10hs