Tema 4. Redes conmutadas e internet.

1. Funcionalidades

Funciones y servicios en TCP/IP

La capa de red actúa como una especie de interfaz o API para la capa de enlace, permitiendo que las distintas tecnologías de todas las redes de internet se puedan intercomunicar. Para ello, la capa de red implementa una serie de funcionalidades:

- Encaminamiento: es la función de buscar un camino entre todos los posibles en una red de paquetes cuyas topologías poseen una gran conectividad. Dado que se trata de encontrar la mejor ruta posible, lo primero será definir qué se entiende por "mejor ruta" y en consecuencia cuál es la "métrica" que se debe utilizar para medirla.

- Conmutación: La conmutación podría definirse como la manera en la que la información navega por la red (en forma de flujos, de circuitos, paquetes, etc).

- Interconexión de redes: Es un conjunto de redes (que pueden ser de tipos diferentes) que están interconectadas por medio de encaminadores, gateways, u otros dispositivos, para que de este modo puedan funcionar como una sola gran red.
- En OSI: control de congestión se encuentra en la capa de red, sin embargo, en TCP/IP se encuentra en la capa de transporte, es decir, lo hace TCP.

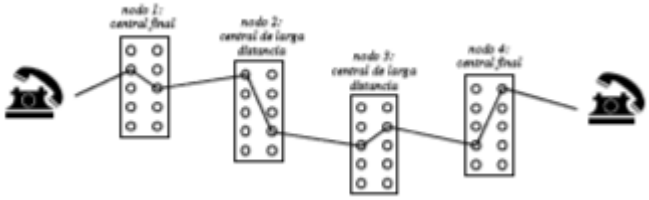
2. Conmutación

La conmutación podría definirse como una redirección en la red.

Hay dos esquemas de conmutación:

- Circuitos: Establece un circuito para la comunicación entre los dos finales. Una vez iniciada la conexión, se crea un circuito que pasa por una serie de nodos intermedios (por ejemplo, en una red de ordenadores podrían ser los routers) y siempre se mantendrá dicho circuito. De hecho, el circuito será dedicado donde los recursos estarán dedicados únicamente a dicha conexión, es decir, habrá una parte del cable reservada para dicho circuito.

  Se crea una especie de tubería para la comunicación entre esos dos finales, únicamente puede pasar información de dicha comunicación por esa tubería. La implementación de la conmutación es más sencilla pero estamos perdiendo recursos que dejamos dedicados para dicha comunicación.

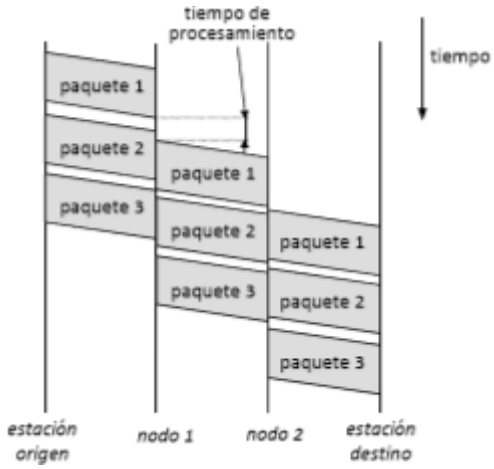


Pasos: Conexión, Transmisión, Desconexión.

Recursos dedicados: Facilita comunicaciones tiempo-real. No hay conteción.

Retraso para el establecimiento de la llamada. Poca flexibilidad para adaptarse de cambios.

- De paquetes:
  - Mediante datagramas: es la conmutación basada en paquetes propiamente dicha. En este caso, la comunicación se divide en paquetes y cada paquete es una entidad lógica independiente que viaja por internet. Un ejemplo de ello es IP, no hay conexión ya que no es necesario reservar recursos pero todos los paquetes que enviemos tendrán un retardo de procesamiento y los datagramas pueden seguir rutas diferentes.



A la hora de enviar un segundo paquete, le pasará lo mismo. Todos los paquetes deben de tener un tiempo de procesamiento en cada nodo para poder ser reenviados.

Cada paquete al ser una entidad independiente deberá de llevar información sobre su destino, al contrario que la conmutación de circuitos.

Para un determinado mensaje es más rápido la conmutación de datagramas, ya que en la basada en circuitos tenemos un retardo debido al establecimiento de conexión.

Basada en circuitos -> Enviar mucha información.

Basada en datagramas -> Enviar poca información.

- Mediante circuitos virtuales: En la conmutación basada en circuitos virtuales, usamos la misma tecnología que en la conmutación basada en datagramas, pero intentando minimizar el tiempo de procesamiento que debe tener el paquete en cada nodo. Para ello, se crean circuitos virtuales por encima de la red basada en datagramas, lo cual implica que todos los paquetes son enviados por el mismo camino consiguiendo reducir así el tiempo de procesamiento dedicado al encaminamiento.

La parte negativa de usar un único camino es la robustez. Si enviamos paquetes a través de la red con la libertad de poder elegir el camino por el cual los enviamos, la red es muy robusta, ya que si se cae un enlace se puede enviar el paquete a través de cualquier otro. El paquete tardará algo más en llegar, pero llegará al destino. Si en una red telefónica se cae uno de los enlaces por los que pasa nuestra llamada se termina la llamada y se tendría que llamar otra vez. Esta limitación es heredada en la conmutación a través de circuitos virtuales.

Ejercicio:

4. Estime el tiempo involucrado en la transmisión de un mensaje de datos para la técnica de conmutación de paquetes mediante datagramas (CDP) considerando los siguientes parámetros:

M: longitud en bits del mensaje a enviar.

V: velocidad de transmisión de las líneas en bps.

P: longitud en bits de los paquetes.

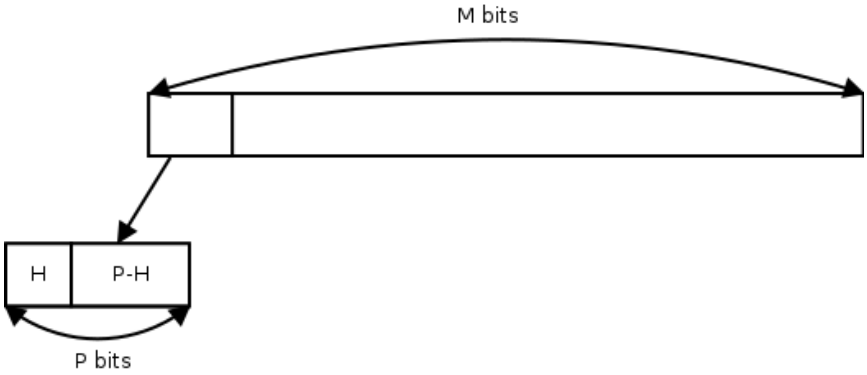
H: bits de cabecera de los paquetes.

N: número de nodos intermedios entre las estaciones finales.

D: tiempo de procesamiento en segundos en cada nodo.

R: retardo de propagación, en segundos, asociado a cada enlace.

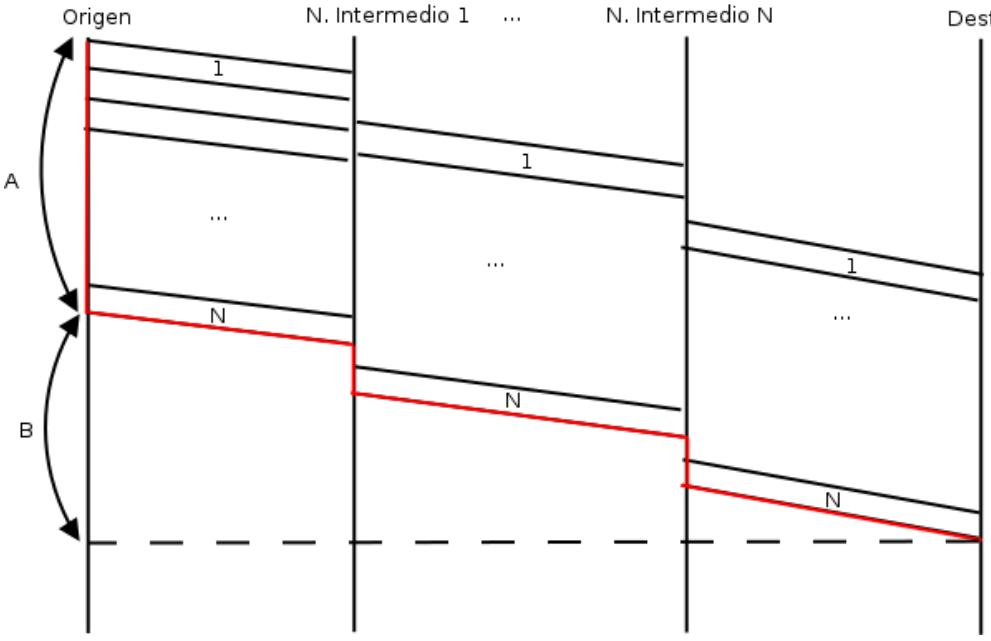
En el ejercicio se asume que los paquetes tienen la misma longitud. Por tanto, en el enunciado nos dicen que tenemos la estructura de paquetes, tendremos los M bits que se deben enviar troceados en trozos de P-H bits (cada paquete con su cabecera).



Para saber el tiempo que vamos a tardar en enviar los paquetes, necesitamos primero saber el número de paquetes:

Np = M / P-H (Redondeamos al entero siguiente).

En un momento dado, la capa de transporte nos ha enviado los M bits del mensaje para que los enviemos. Haciendo esta suposición nos olvidamos del resto de capas y nos centramos únicamente en la capa de red. Así, cada paquete será transmitido inmediatamente después que el anterior y además, cada uno tendrá su tiempo de transmisión, su tiempo de propagación y su tiempo de procesamiento en cada nodo.



El ejercicio nos pide el tiempo marcado en rojo. Es decir, el tiempo desde que comenzamos a enviar el primer bit hasta que se recibe por completo en el receptor.

Por tarto tenemos por un lado el tiempo que se tarda en trasmitir todos los paquetes A y el tiempo que tarda el último bit en propagarse por toda la red B.

A = Np * (P / V)

B = N * (R + D + (P / V)) + R

Por último el tiempo total de transmisión del mensaje sería la suma de ambas partes:

Tt = A + B

3. El protocolo IP

El protocolo IP hace que sea posible la idea de internet como una intercomunicación de redes. IP nos permite abstraernos de las distintas tecnologías que usan las redes y por tanto, que todas las redes puedan comunicarse entre sí y que exista el concepto de InternerWorking.

-IP v4, es la versión más extendida de IP. Se ha hecho un gran esfuerzo por extender IP v6, ya que soluciona bastantes problemas que IP v4 no.

-IP permite la interconexión de redes y el direccionamiento en internet, es decir, con las direcciones IP determinamos a los distintos dispositivos en internet a los que les puede llegar un paquete.

-Para poder enviar un paquete a través de internet, se utiliza el enrutamiento que utiliza a su vez una retransmisión salto a salto.

-IP es parecido a UDP, de hecho los paquetes de IP se denominan datagramas al igual que UDP. No es orientado a conexión, ni es fiable, sino que es best-effort, es decir, hace el máximo esfuerzo por que el paquete llegue a su destino pero si no llega "le da igual". En IP no hay "handshake" ni existe el control de errores ni de flujo.

-También gestiona la fragmentación, la cual es bastante importante en internet ya que coexisten distintos tipos de tecnologías y cada una de ellas está preparada para un tamaño máximo de paquete MTU (Maximum Transmission Unit), al ser este último distinto en cada red, puede ser que al pasar un datagrama de una red a otra el MTU sea distinto y se tenga que fragmentar el paquete.

Direcciones IP

Aunque el nombre de dominio es lo que usamos nosotros y lo que usan muchas de las aplicaciones que tenemos instaladas, a bajo nivel lo que realmente se usan son las direcciones IP. El protocolo DNS es el encargado de traducir un nombre de dominio a su dirección IP.

Las direciones IP deben ser unívocas, es decir, no se pueden compartir entre servicios y además, debe existir algún mecanismo para saber el camino que debe tomar un paquete para llegar a su destino.

El protocolo IP es un direccionamiento jerárquico, compuesto por dos la subred y el dispositivo. El prefijo de subred indica la dirección de un grupo de ordenadores, que es lo que denominamos subred, que están localizados en el mismo sitio y la máscara de subred determina qué parte de una dirección IP pertenece a la subred y qué parte de la dirección IP identifica al ordenador de una subred. No puede haber dos direcciones IP^iguales, por tanto, si una subred está compuesta por 20 ordenadores cada uno debe tener la parte de host de la dirección IP diferente.

La máscara de red diferencia ambas partes, si la ponemos en binario, todos los unos serán la parte de la dirección de subred y todos los ceros serán de la parte del host.

Por ejemplo, en este caso tenemos los octetos divididos, la porción que se usará para identificar a la subred será 200.27.4 y la porción para identificar el host será 112. Todos los host cuya dirección comience por 200.27.4 estarán localizados en el mismo lugar.

> Dirección IP -> 200.27.4.112 = 11001000.00011011.00000100.01110000
>
> Máscara -> 255.255.255.0 = 11111111.11111111.11111111.00000000

La dirección anterior junto a su máscara también la podemos encontrar de la siguiente forma 200.27.4.112/24, el último 24 representa el número de unos a la izquierda en la máscara.

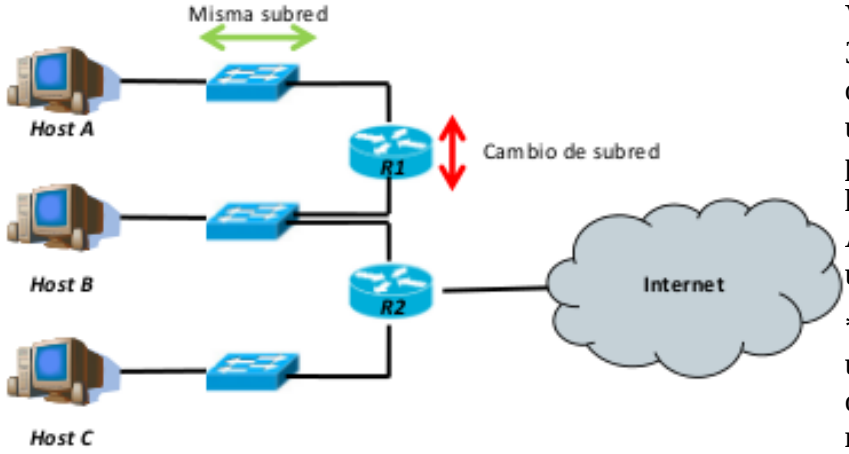
Para poder determinar de una forma rápida cuál es la subred utilizamos la subred, se realiza una operación bit a bit de la diracción IP y la máscara de subred:

	200.27.4.112	= 11001000.00011011.00000100.01110000
	&	&
	255.255.255.0	= 11111111.11111111.11111111.00000000
Subred ➔	200.27.4.0	= 11001000.00011011.00000100.00000000

A efectos prácticos lo que se hace es poner ceros en la dirección IP donde haya ceros en la máscara. En este ejemplo, al pertenecer los tres primeros octetos a la subred se dejan intactos y eliminamos únicamente el último octeto que pertenecía al host. Cuando un router tenga que decidir por dónde encaminar un paquete, tomará la misma decisión para todos los ordenadores pertenecientes a una misma subred, de la misma forma que se haría en la red telefónica.

¿Qué es una subred?

Es un conjunto de ordenadores que están localizados en el mismo lugar, que comparten una parte de su dirección IP y que se accede a ellos a través del mismo camino.

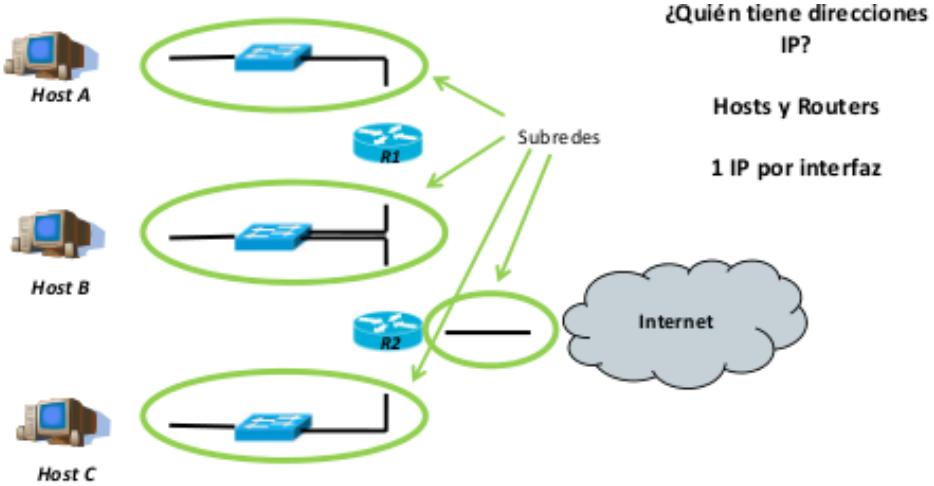


Vemos un ejemplo de red, en ella 3 grupos de host, todos los ordenadores están conectados a unos switches (conmutadores), para comunicar cada grupo de host entre sí, se utilizan routers. Además uno de esos routers se utiliza para acceder a la internet.

*La diferencia entre un switch y un router, es que el switch sólo opera en la capa de enlace y el router en la capa de red.

"Para determinar las subredes, separe cada interfaz de los hosts y routers, creando redes aisladas. Dichas redes aisladas se corresponden con las subredes."

Siguiento la regla anterior y aplicándola a la siguiente imagen, obtenemos el siguiente resultado.



Como se puede apreciar se ha separado tanto los routers como los host, esto se debe a que los hosts también operan en capa de red, es decir, los hosts tienen incluidas todas y cada una de las capas. Los routers sólo tienen la de capa de red hacia abajo y los switches, sólo tienen de la capa de enlace hacia abajo. Debido a que sólo tenemos en cuenta para hacer la separación a los dispositivos que tienen una entidad IP (hosts y routers) sólo hacemos la separación con ellos. Todos los dispositivos que operan en capa de red necesitan una dirección IP. Un router tendrá tantas direcciones IP como interfaces tenga. Por ejemplo, el router 2 tendrá tres direcciones IP.

¿Cómo se elige la máscara?

La máscara de red depende del número de hosts que haya en dicha red, ya que dependiendo del número de ceros que dejemos podremos tener un número de hosts concreto. El número de dispositivos máximo será:

Num_dispositivos = 2^num_ceros – 2

Por tanto, en el ejemplo que vimos antes cuya máscara era 255.255.255.0, podríamos tener:

2^8 - 2 = 254 dispositivos como máximo.

Debemos restar dos debido a que hay dos direcciones IP que están reservadas en todas las subredes, la primera y la última.

- 200.27.4.0 = 11001000.00011011.00000100.00000000 → Reservada (subred)
- 200.27.4.1 = 11001000.00011011.00000100.00000001 → Dispositivo #1
- ...
- 200.27.4.254 = 11001000.00011011.00000100.11111110 → Dispositivo #254
- 200.27.4.255 = 11001000.00011011.00000100.11111111 → Reservada (difusión)

- Dirección de subred: es la primera dirección IP en la subred. Se utiliza para el enrutamiento y no puede utilizarse en ningún dispositivo, es decir, nunca veremos ningún dispositivo cuya dirección IP termine en 0.
- Dirección de broadcast (difucsión): es la última dirección IP en la subred y se utiliza para enviar mensajes que puedan ser escuchados por todos los dispositivos de la subred. En vez de enviar un mensaje a cada dispositivo, se envía uno a la dirección broadcast. Si enviamos un mensaje broadcast a una subred con 254 ordenadores, no se envían 254 mensajes iguales sino uno sólo que se encontrará en un sitio donde pueda ser leído por todos (por ejemplo, si los 254 ordenadores están conectados al mismo switch, ahí se encontrará el mensaje de broadcast).

Direcciones públicas y privadas

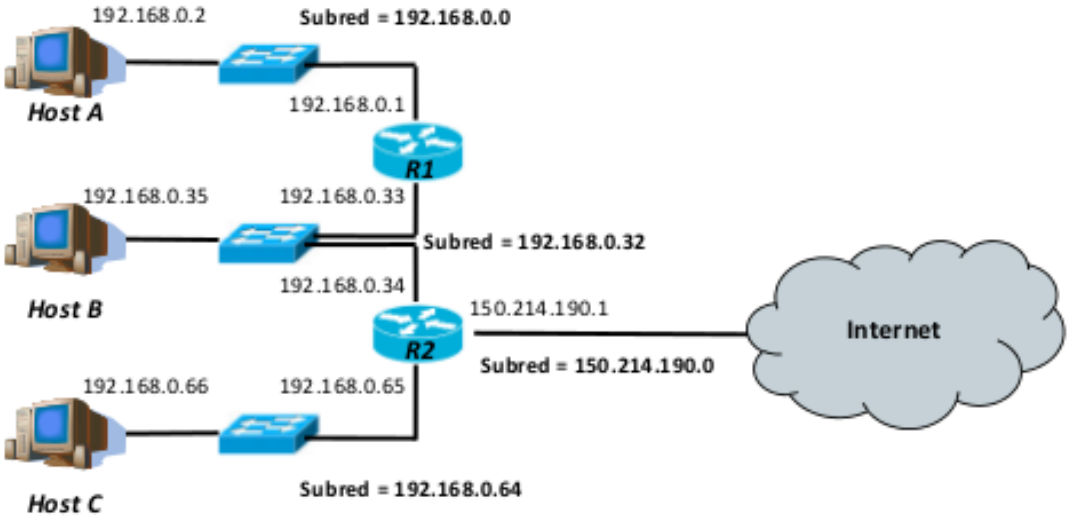
Para que un ordenador con una IP privada pudiera comunicarse en internet necesita una IP pública, ya que si no no podría comunicarse en internet, no se sabría a dónde enviar los paquetes respuesta para dicho ordenador. Dicha dirección IP pública es la del router. El router es el que actúa como separación entre la red pública y la privada. Los router hacen operaciones NAT (Network Address Translation), que básicamente consisten en sustituir la dirección IP privada del host que se quiere comunicar con internet por la suya propia pública, de forma que el receptor cuando envíe la respuesta la mandará a la dirección IP del router y éste ya la volverá a sustituir en el mensaje respuesta por la dirección IP del host inicial.

- Clase A: 1.0.0.1 a 126.255.255.254 (8 bits red, 24 bits hosts) Máscara de subred: 255.0.0.0. 2^24 – 2 = 16777214 dispositivos.
- Clase B: 128.0.0.1 a 190.255.255.254 (16 bits red, 16 bits hosts) Máscara de subred: 255.255.0.0. 2^16 – 2 = 65534 dispositivos.
- Clase C: 192.0.0.1 a 222.255.255.254 (24 bits red, 8 bits hosts) Máscara de subred: 255.255.255.0. 2^8 – 2 = 254 dispositivos.

Ejercicio:

Dada una red como la de la imagen, asignar direcciones IP a cada subred corporativa teniendo en cuenta que en cada subred hay 30 dispositivos y que la primera dirección IP por la que empezaremos a enumerar será 192.168.0.0. Además, la subred de acceso está directamente conectada a internet a través del router del ISP por lo que tendrá una dirección IP pública.

- Subredes corporativas: 30 dispositivos, direcciones privadas 192.168.0.0 ➔ 5 ceros, /27
- Subred de acceso: dirección pública (ISP) ➔ 2 ceros, /30, 150.214.190.0 (UGR)

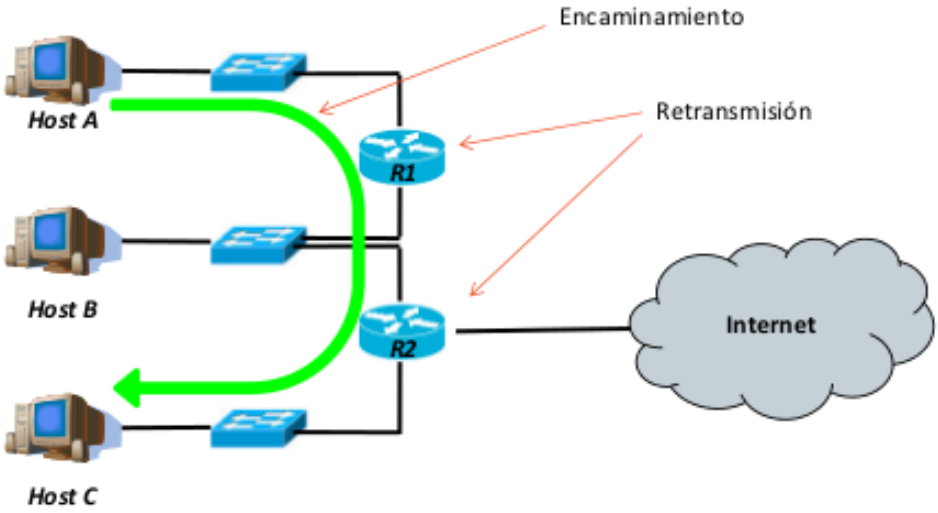


En primer lugar, como tenemos 30 dispositivos en cada subred debemos reservar como mínimo 5 ceros (una máscara /27) en la dirección IP para identificar a cada host (2^5 - 2 = 30). En el caso de la subred de acceso, sólo tenemos dos dispositivos (nuestro router y el router del ISP) por lo que sólo necesitamos 2 ceros (una máscara /30) en la dirección IP para identificar a cada host.

<u>El encaminamiento:</u>

Tiene como objetivo llevar cada paquete a su destino. Es la suma de dos operaciones:

- Encaminamiento per se (routing): que consiste en decidir la ruta del paquete. Es decir, cuando llegue un paquete a un nodo, teniendo en cuenta su destino, decidir si lo enviamos por una interfaz o por otra.
- Retransmisión (forwarding): es la implementación de dicho encaminamiento. Consiste en reenviar el paquete por una interfaz o por otra dependiendo de la ruta que se haya decidido.



Ejemplo: Si queremos enviar un paquete desde el host A hasta el host C, se debe decidir en primer lugar que dicho paquete debe seguir la línea verde. En cada router intermedio, se realizará la retransmisión de acuerdo con la ruta previamente decidida.

Retransmisión salto-a-salto:

Según la imagen anterior, en cada nodo se debe decidir por qué camino enviar el paquete o cuál es el siguiente nodo al que hay que enviarselo.

Las tres retransmisiones serían:

- Desde el host A, para llegar a C, lo envío a R1.
- Desde R1, para llegar a C, lo envío a R2.
- Desde R2, para llegar a C, se envía directamente a C.

Hay dos tipos de encaminamiento:

- Directo: para redes conectadas localmente.
- No directo: para redes accesibles a través de uno o más routers.

Tabla de encaminamiento

Es una tabla que pertenece al host, es decir, cada host tiene su propia tabla de encaminamiento, y en la cual hay mínimo tres columnas:

- Dirección IP destino: siempre es una dirección de subred. Dado que el objetivo es hacer la tabla lo más pequeña posible, en vez de tener una fila por cada host al que podemos enviar un paquete, sólo tenemos una fila por cada subred que represente a todos y cada uno de los host pertenecientes a dicha subred.
- Máscara: asociada a dicha subred.
- Siguiente nodo: router más cercano.

Dirección IP destino	Máscara	Siguiente nodo
192.168.0.0	/27	-
192.168.0.32	/27	192.168.0.1
192.168.0.64	/27	192.168.0.1
150.214.190.0	/30	192.168.0.1

Para enviar un paquete al nodo C, con dirección IP 192.168.0.66, debemos consultar la tabla de encaminamiento para saber que pertenece a la subred 192.168.0.64 y enviarlo al siguiente nodo correspondiente que en este caso sería 192.168.0.1.

*Cuando no hay un siguiente nodo, es decir, el host destino está dentro de nuestra subred, el

paquete se envía directamente a dicho host sin pasar por ningún intermedio utilizando la dirección MAC (dirección de la tarjeta de red) de dicho host destino.

Para determinar numéricamente la subred a la que pertecene una dirección IP destino, se realiza una operación & con todas y cada una de las máscaras de la tabla y compara el resultado con la dirección de destino de la tabla. La entrada seleccionada será la que coincida, es decir, haga matching, con la dirección IP destino de la tabla.

- 192.168.0.66 & /27 = 11000000.10101000.00000000.010**00010** & /27 = 192.168.0.64
  - ¿192.168.0.64 = 192.168.0.0? NO
- 192.168.0.66 & /27 = 11000000.10101000.00000000.010**00010** & /27 = 192.168.0.64
  - ¿192.168.0.64 = 192.168.0.32? NO
- 192.168.0.66 & /27 = 11000000.10101000.00000000.010**00010** & /27 = 192.168.0.64
  - ¿192.168.0.64 = 192.168.0.64? SÍ ➔ Siguiente Nodo = 192.168.0.1
- 192.168.0.66 & /30 = 11000000.10101000.00000000.01000**10** & /30 = 192.168.0.64
  - ¿192.168.0.64 = 150.214.190.0? NO
- ¿Colisión? La de máscara más restrictiva (+ 1s)

*Cuando ocurre una colisión, se elige la máscara más restrictiva, es decir, la máscara que tiene más unos. En el ejemplo anterior si estuvieran en conflicto las dos últimas, se elegiría la /30.

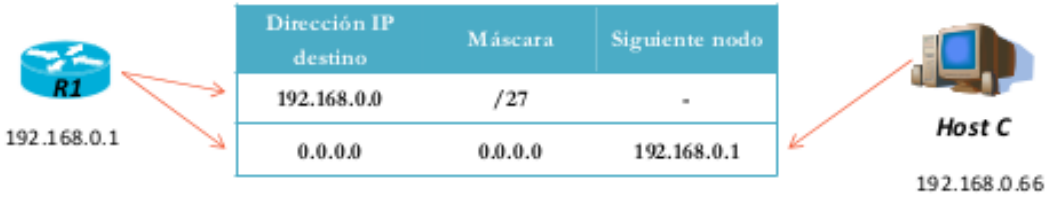
<u>Problemas con la tabla de encaminamiento</u>

- No direcciona internet: Por ejemplo, si entramos desde el navegador del host A en www.google.com y el DNS del host A sabe que la IP asociada a dicho nombre de dominio es 172.194.34.209, al intentar hacer matching de dicha IP con la tabla de encaminamiento no haremos matching con ninguna entrada. La capa de red ignora todos los paquetes que no hagan matching en la tabla de encaminamiento y por tanto, no podemos direccionar internet.
- Sólo un camino de salida: El host A sólo está conectado con el resto de subredes mediante el router 1, pero ¿si necesitamos 4 entradas?.

Dirección IP destino	Máscara	Siguiente nodo
192.168.0.0	/27	-
192.168.0.32	/27	192.168.0.1
192.168.0.64	/27	192.168.0.1
150.214.190.0	/30	192.168.0.1

¡¡Usar la entrada por defecto!! ➔ /0

Para resolver los problemas de esta tabla, debemos usar la orden por defecto, que nos dice el camino a llegar al resto de dispositivos de internet. En el caso del host A, todos los demás dispositivos que no pertenecen a su subred estarán detrás del router 1 y por tanto, es ahí donde ponemos la orden por defecto. Así, la tabla de encaminamiento del host A sólo tendría dos entradas: una para poder direccionar paquetes en la subred del dispositivo A y otra para poder comunicarse con el resto de dispositivos.



Ejercicio:

Diseñar la tabla de encminamiento en el router 2.

Para ello hay que seguir los siguientes pasos:

- Incorporar todas las redes directamente conectadas.
- Incorporar la entrada por defecto.
- Añadir todas las entradas adicionales necesarias.

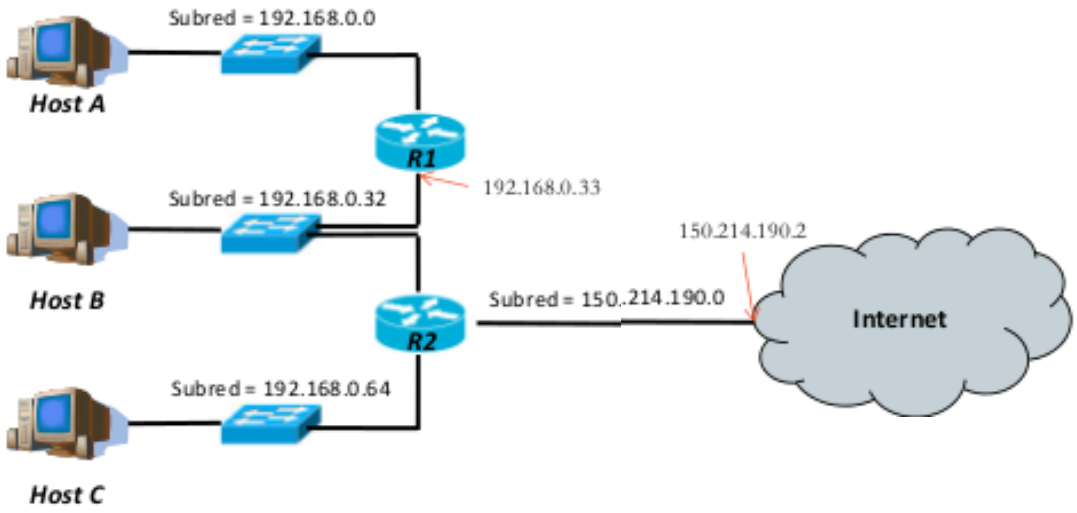
Dirección IP destino	Máscara	Siguiente nodo
192.168.0.32	/27	-
192.168.0.64	/27	-
150.214.190.0	/30	-
0.0.0.0	/0	150.214.190.2
192.168.0.0	/27	192.168.0.33

Las tres primeras direcciones son a las que está directamente conectado.

Después tenemos la entrada por defecto con el siguiente nodo a la red conectada con el router ISP.

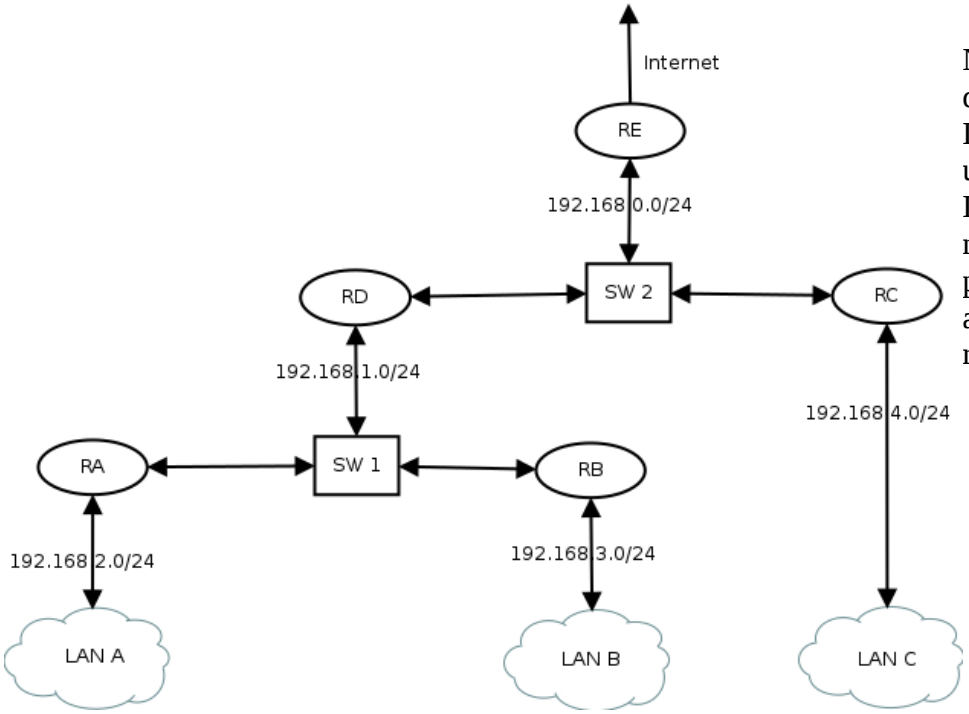
Por último, añadimos todas aquellas redes que no han quedado especificadas.

Debemos comprobar si con las entradas actuales en la tabla podemos llegar a todas las entradas.



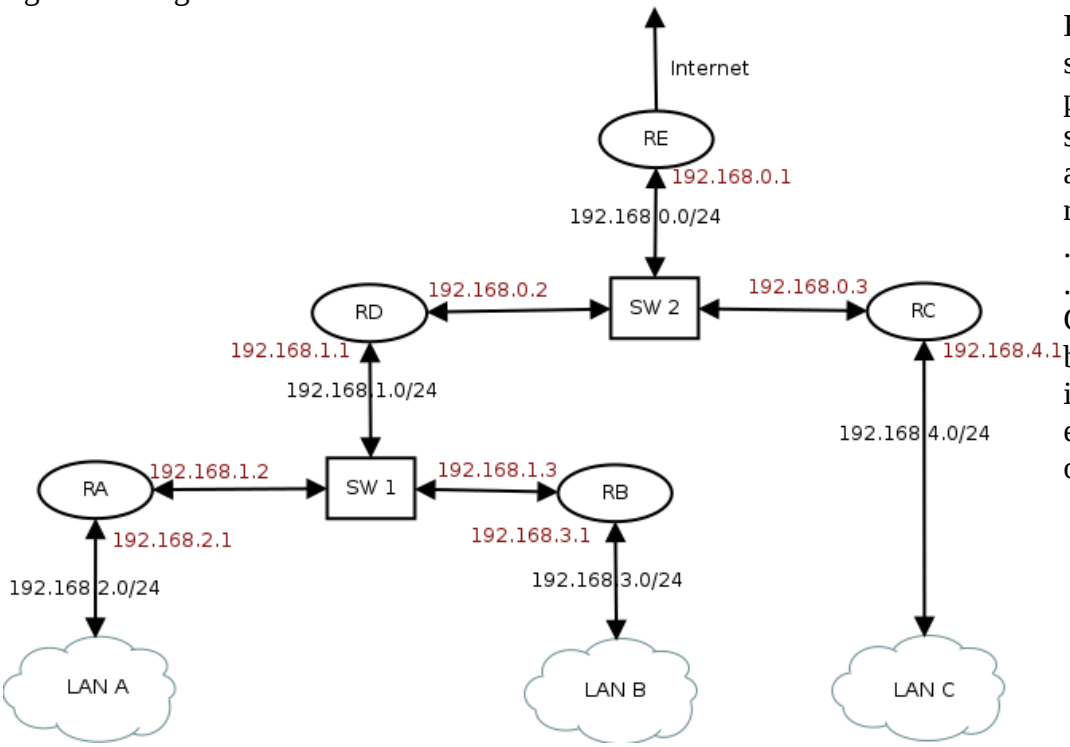
Ejercicio:

7. Imagine una situación donde hay cinco routers RA-RE. RA, RB y RC se conectan cada uno a una red local A, B y C, siendo cada router única puerta de enlace de cada red. RA, RB y RD están conectados entre sí a través de un switch. RC, RD y RE están conectados entre sí a través de un switch. RE conecta a Internet a través de la puerta de acceso especificada por el ISP. Especifique tablas de encaminamiento en los routers. Asigne a voluntad las direcciones IP e interfaces necesarias.



Nos dan las direcciones de cada subred. En cada LAN se encuentran los usuarios finales de la red. Los switch normalmento no tienen dirección IP, pero pueden tenerla para así poder configurarlos remotamente.

La asignación de direcciones IP a cada una de las interfaces de los routes se observa en rojo en la siguiente imagen:



El procedimiento seguido es, paritendo de un switch central, asignar la .1 al router de arriba, .2 al de la izq y .3 al de la dcha. Cada vez que bajamos un nivel incrementamos el penúltimo octeto.

Empezamos con la tabla de encaminamiento de RD. En primer lugar, añadimos las entradas de las subredes a los que estamos directamente conectados, es decir, la subred 192.168.0.0 y la subred 192.168.1.0. Es importante recordar que siempre se añadirán direcciones de subred con la excepción de un dispositivo que tenga un camino unívoco para llegar hasta él, una subred con un único dispositivo. En segundo lugar, añadimos la entrada por defecto, que conducirá hasta RE ya que es el router que está conectado a internet. Por último, añadimos las subredes a las cuales no podemos llegar con las entradas actuales, que en este caso son las subredes que se encuentran detrás de RA, RB y RC.

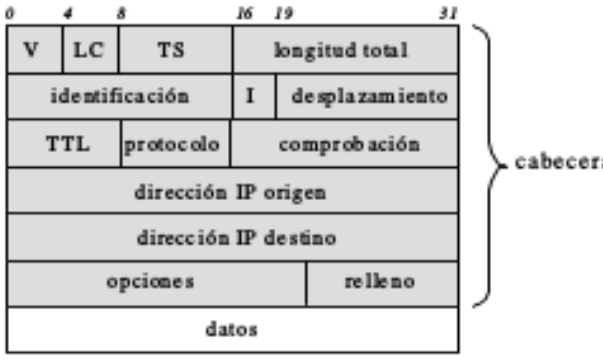
Dirección IP destino	Máscara	Siguiente nodo
192.168.0.0	/24	-
192.168.1.0	/24	-
0.0.0.0	/0	192.168.0.1
192.168.2.0	/24	192.168.1.2
192.168.3.0	/24	192.168.1.3
192.168.4.0	/24	192.168.0.3

A continuación realizamos la tabla de encaminamiento de RC. En primer lugar, añadimos las subredes a las que RC está directamente conectado, es decir, las subredes 192.168.0.0 y 192.168.4.0. Tras esto, añadimos la entrada por defecto para acceder a internet, por tanto será a través de RE. Por último, añadimos el resto de subredes a las que no llegamos directamente, es decir, las subredes que están detrás de RA, RB y RD.

Dirección IP destino	Máscara	Siguiente nodo
192.168.0.0	/24	-
192.168.4.0	/24	-
0.0.0.0	/0	192.168.0.1
192.168.1.0	/24	192.168.0.2
192.168.2.0	/24	192.168.0.2
192.168.3.0	/24	192.168.0.2

Las demás tablas de encaminamiento serían siguiendo el mismo procedimiento anterior.

Formato de un Datagrama IP



-V: campo en el que especificamos la versión de IP.
-LC (longitud cabecera): debido a que la longitud puede ser variable, hay que especificarla, se mide en palabras de 32b y puede tener hasta 5 valores.
-TS (tipo servicio): se utiliza para informar de un contenido de mayor o menor prioridad.
-Longitud total: se refiere a la longitud de la cabecera más la sección de datos.
-identificación: numero de secuencia.
-I: flags, el que más nos itneresa es MF que se pone a 0 si es el último fragmento de una segmentación, en caso contrario estará a 1.
-desplazamiento: posición de los datos del datagrama segmentado en el original.
-TTL: limita el tiempo que un datagrama puede pasar en la red, se decrementa cada vez que pasa por un router si todo va bien.
-protocolo: especifica que protocolo está por encima de IP: TCP, UDP o ICMP.
-comprobación: comprobación de que la transmisión ha sido correcta.
-opciones: se especifican algunas opciones de las que se puede hacer uso.
-relleno: debido a que el paquete debe tener tamaño múltiplo de 32b, si se añaden opciones deben añadirse bits de relleno hasta completar la última palabra de 32b de la cabecera.

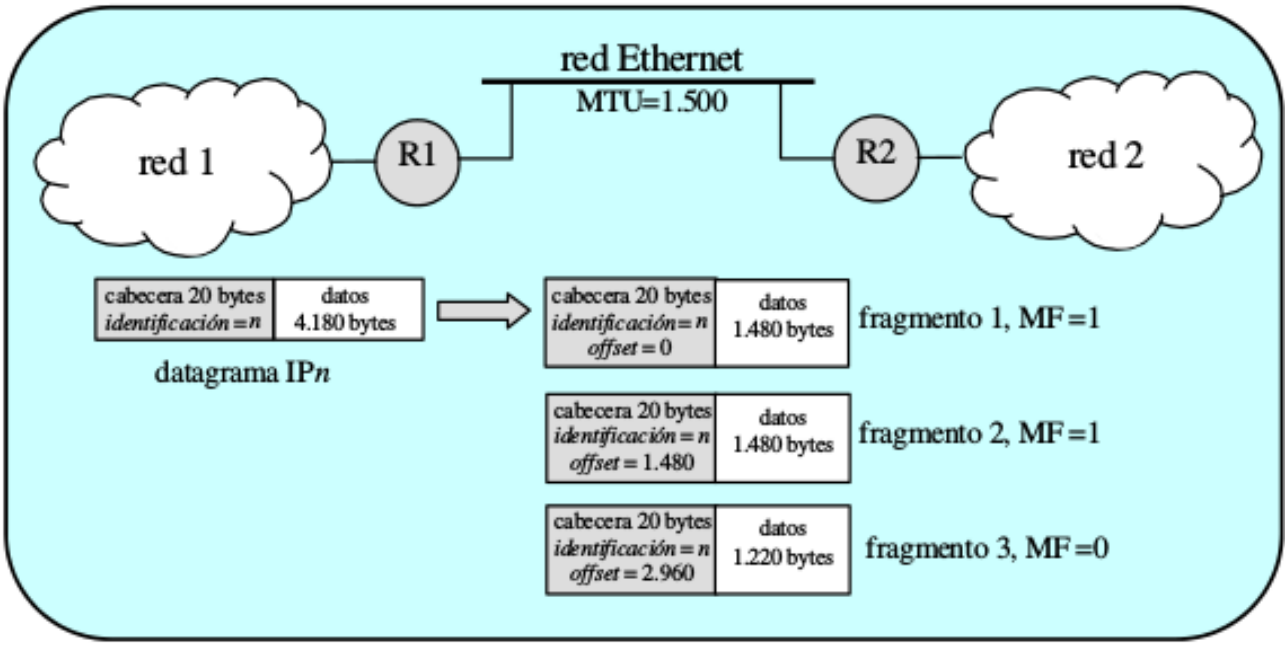
Fragmentación Ipv4

El tamaño máximo de un datagrama IP es de 65535 bytes, esto se debe a que el campo de la cabecera longitud total sólo puede almacenar números de hasta 16 bits, es decir 2^16 - 1 = 35535 bytes. A pesar de que este es el tamaño máximo que podemos direccionar, el MTU de la mayoría de redes es mucho más inferior por la tecnología subyacente, tanto física como capa de enlace.

Nivel de enlace	MTU (bytes)
PPP normal	1500
PPP bajo retardo	296
X.25	1600 (RFC 1356)
Frame Relay	1600 (normalmente)
Ethernet DIX	1500
Ethernet LLC-SNAP	1492
Token Ring 4 Mb/s	4440 (THT 8ms)
Classical IP over ATM	9180

En la imagen tenemos varios ejemplos de tecnologías de red con el máximo de MTU que aceptan.

Al haber diferentes MTU se necesita un procedimiento para poder fragmentar los paquetes.



-identificación: identificación unívoca para el datagrama.
-offset (desplazamiento): este flag se utiliza para saber qué hueco del datagrama completo pertenece cada fragmento. El offset del primer fragmento será 0 y el seguiente se hará contando el número que tenga el primer datagrama.
-MF: permite saber al destino cuando le han llegado todos los fragmentos de un paquete. Todos los fragmentos tendrán el flag activado, menos el último que lo tendrá a 0.

4. Asociación con Capa de Enlace: Protocolo ARP

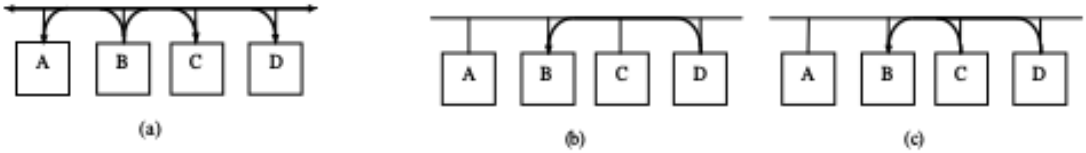
IP se ocupa de decidir quién será el próximo nodo al que le enviaremos el paquete pero la capa de enlace es la que realmente envía el paquete a dicho nodo y para ello, en vez de utilizar la dirección IP utiliza la dirección MAC del dispositivo. La dirección MAC es la dirección unívoca de la tarjeta de red que identifica a un dispositivo.

En el ejemplo del encaminamiento del host A al C, el host A envía el paquete al router 1, para ello

debe conocer su dirección MAC. Tras ello, el router 1 deberá enviar el paquete a la dirección MAC del router 2 y por último, el router 2 la enviará a la dirección MAC del host C.
Por tanto, aunque IP nos indique el camino a seguir, la información no se envía a una dirección IP destino sino a una dirección MAC destino. Para ello, necesitamos algún protocolo para traducir una dirección IP a una dirección MAC y a la inversa. Dicho protocolo es ARP. El protocolo no se encuentra ni en capa de red ni en capa de enlace, sino que actúa de interfaz entre las dos.

Formato de una dirección MAC
El formato es el siguiente:
HH-HH-HH-HH-HH-HH donde H es un número hexadecimal.

Protocolo ARP (directo): Obtener MAC a partir de IP(a) y (b).
Esta es la operación más típica: una vez sabido cuál es el siguiente nodo en la ruta, obtenemos su dirección MAC a partir de la dirección IP que hay en la tabla de encaminamiento.
Protocolo RARP(inverso): Obtener MAC a partir de IP(a) y (c).
Esto se suele usar cuando no hay información sobre IP inicialmente, por ejemplo, cuando se inician sistemas operativos en máquinas virtuales en redes virtuales y así poder conocer las IPs de todos los elementos de la red.



Formato ARP

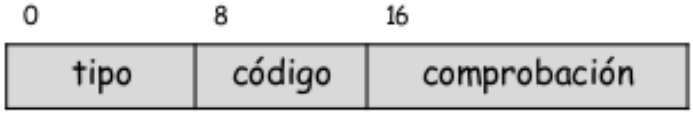
0	8	16	31
Htipo		Ptipo	
Hlen	Plen	Operación	
Hemisor (bytes 0-3)			
Hemisor (bytes 4-5)		Pemisor (bytes 0-1)	
Pemisor (bytes 2-3)		Hsol (bytes 0-1)	
Hsol (bytes 2-5)			
Psol (bytes 0-3)			

Lo primero que se indica en un paquete ARP es el tipo de protocolo de capa de enlace y a continuación el tipo de protocolo de capa de red, por ejemplo el protocolo Ethernet se identifica con un 1 y el protocolo IP con un 8 (H significa hardware y P, Protocol), tras eso, se indica la longitud de cada una de las direcciones en bytes (primero la de capa de enlace y luego la de capa de red), en el caso de Ethernet serían 6 bytes y en el de IP, 4. Esto sirve para que se sepa el formato de dirección que le indicaremos a continuación, al igual que en IP se usaba el campo de versión para saber la sintaxis del resto del paquete. En primer lugar indicamos la dirección hardware del emisor, tras eso, indicamos la dirección IP del emisor y por último, indicamos las direcciones hardware e IP del receptor.
El paquete conoce las longitudes de todas estas direcciones gracias a los campos de longitud de dirección (aunque en la imagen la parte de direcciones parezca divida en varios campos es uno sólo, Hemisor serían 6 bytes y Pemisor serían 4). Este formato lo usaremos tanto para solicitudes como para respuestas, el tipo de operación se indica también en la cabecera.

5. El protocolo ICMP

Es un protocolo de capa de enlace, que está encapsulado en IP, simplemente significa que la cabecera del protocolo que encapsula va antes que la del otro, es decir, un protocolo está en una capa que da servicio a otro protocolo que está encapsulado y es un protocolo de gestión que informa de problemas que pueden surgir en una comunicación por internet: paquetes que no han llegado, destinos que no se conocen, etc.

Los errores siempre se envían al origen, por ejemplo, cuando intentamos hacer ping a un ordenador que no existe y vemos un mensaje en terminal indicándonos que no existe dicha dirección.



-tipo: tipo del mensaje, con 8b de tamaño.
-código: subtipo del mensaje, con 8b.
-comprobación: este contenido se refiere al paquete o a una parte del paquete que ha causado el error.

Campo *tipo*	Mensaje ICMP
8/0	Solicitud/respuesta de eco
3	Destino inalcanzable
4	Ralentización del origen
5	Redireccionamiento
11	Tiempo de vida excedido
12	Problema de parámetros
13/14	Solicitud/respuesta de sello de tiempo
17/18	Solicitud/respuesta de máscara de red