Tema 3_2

LA ARQUITECTURA DE RED

SI | 23-24

Indices

1. La arquitectura de red.

¿Cómo comunicamos a los usuarios con las aplicaciones de otros computadores?

Cuando hablamos de arquitectura de red, puede que pensemos en como está construida la red, los cables, los equipos, etc. Pero no es así, el concepto de arquitectura de red es más amplio e incluye cuestiones relacionadas con el hardware y con el software de una red.

Antes de definir el concepto de arquitectura de red, es conveniente que entiendas que uno de los problemas más importantes a la hora de diseñar una red no es que los equipos se conecten entre sí, si no que estos equipos puedan comunicarse, entenderse, compartir recursos, que al fin y al cabo es lo que pretendemos. Para esto ya hemos mencionado que se necesitan unos protocolos de comunicaciones. Debido a la complejidad que acarrea considerar la red como un todo, se consideró oportuno organizar las redes como una serie de capas, donde cada capa se ocuparía de alguna función. De esta forma se reduciría la complejidad del diseño de la red y de las aplicaciones que en ella se utilicen.

Por tanto, podemos definir arquitectura de red como el conjunto de capas o niveles, junto con los protocolos definidos en cada una de estas capas, que hacen posible que un ordenador se comunique con otro ordenador independientemente de la red en la que se encuentre.

Esta definición implica, que la especificación de una arquitectura de red debe incluir información suficiente para que cuando se desarrolle un programa o se diseñe algún dispositivo, cada capa responda de forma adecuada al protocolo apropiado.

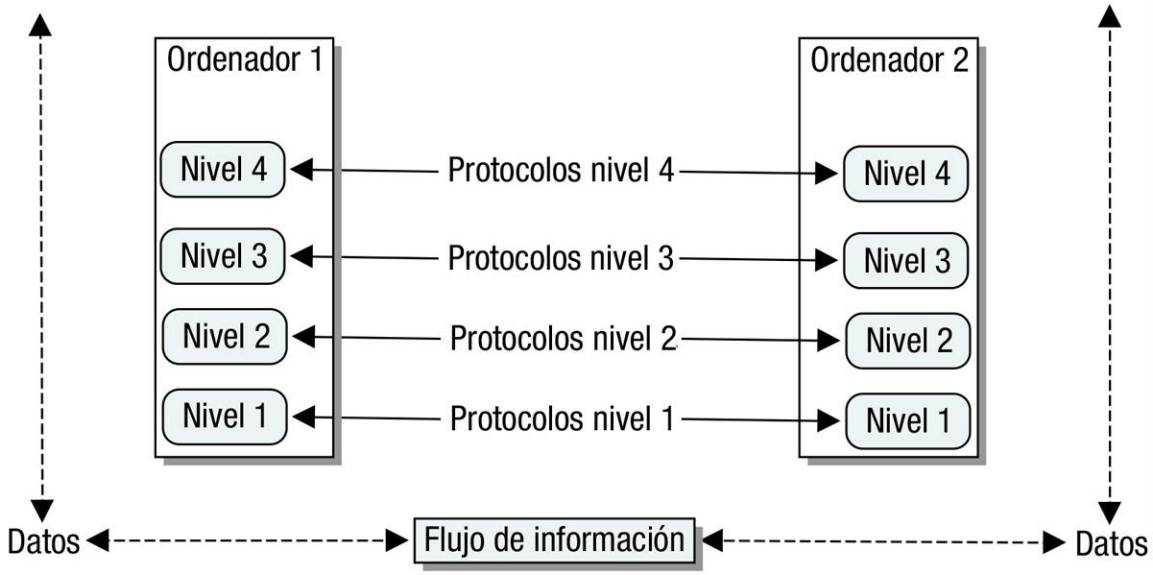
De todo esto podemos concluir que la arquitectura de red tendrá que tener en cuenta al menos tres factores importantes como son:

- La forma en la que se conectan los nodos de una red, que suele conocerse como **topología**, además de las características físicas de estas conexiones.
- La manera de cómo compartir información en la red, que en algunos casos obligará a elegir un **método de acceso a la red** y unas reglas para evitar perdida de información.
- Unas reglas generales que no sólo favorezcan la comunicación, si no que la establezcan, mantengan y permitan la utilización de la información, estás reglas serán los **protocolos de comunicación**.
- A continuación, estudiaremos con más detalle cómo funcionan las arquitecturas basadas en niveles, los protocolos y lo más importante, veremos los dos modelos más importantes en el desarrollo de las redes, el modelo de referencia OSI y la pila de protocolos TCP/IP, que podemos considerarla como la arquitectura base para las comunicaciones por Internet.

1.1. MODELO OSI Y PROTOCOLOS TCP/IP.

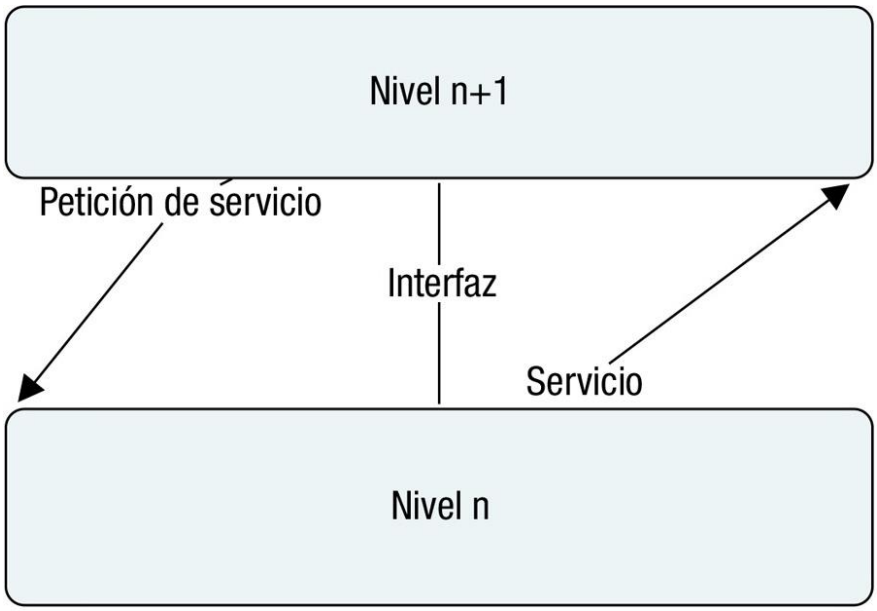
Ya hemos comentado anteriormente, que la arquitectura de red se dividía por niveles o capas para reducir la complejidad de su diseño. Esta división por niveles conlleva que cada uno de estos niveles tenga asociados, uno o varios protocolos que definirán las reglas de comunicación de la capa correspondiente. Por este motivo, también se utiliza el término **pila de protocolos** o **jerarquía de protocolos** para definir a la arquitectura de red que utiliza unos protocolos determinados, esto lo veremos más claramente cuando expliquemos el conjunto de protocolos TCP/IP.

Pero ¿cómo funciona una arquitectura basada en niveles? Para poder explicar esto utilizaremos diferentes gráficos que creemos que pueden ilustrar mejor la explicación.



En el gráfico anterior, podemos ver el esquema de una arquitectura de red de cuatro niveles. Podemos observar dos ordenadores que tendrán implementada la arquitectura, como tenemos cuatro niveles, cada nivel tendrá sus protocolos, por lo que podemos decir que las comunicaciones entre niveles iguales se hacen a través de los protocolos correspondientes. Pero el flujo real de información, con los datos que queremos transmitir irá de un ordenador a otro pasando por cada uno de los niveles. Esto implica que en la realidad los datos no se transfieren directamente de una capa a otra del mismo nivel, si no que cada capa pasa los datos e información de control a la capa adyacente. De esta manera la información pasará por todas las capas, se pasará al medio de transmisión adecuado y posteriormente sucederá lo mismo, pero en sentido contrario, en el otro ordenador. De esta manera la información llegará a su destino y cada nivel sólo se ocupará de los datos y la información de control que necesite, según el protocolo utilizado, sin preocuparse de lo que hagan o necesiten los otros niveles.

Cabe mencionar que con esta forma de trabajar cada capa tiene unos servicios asignados, además las capas están jerarquizadas y cada una tiene unas funciones, de esta forma los niveles son independientes entre sí, aunque se pasan los datos necesarios de una a otra.



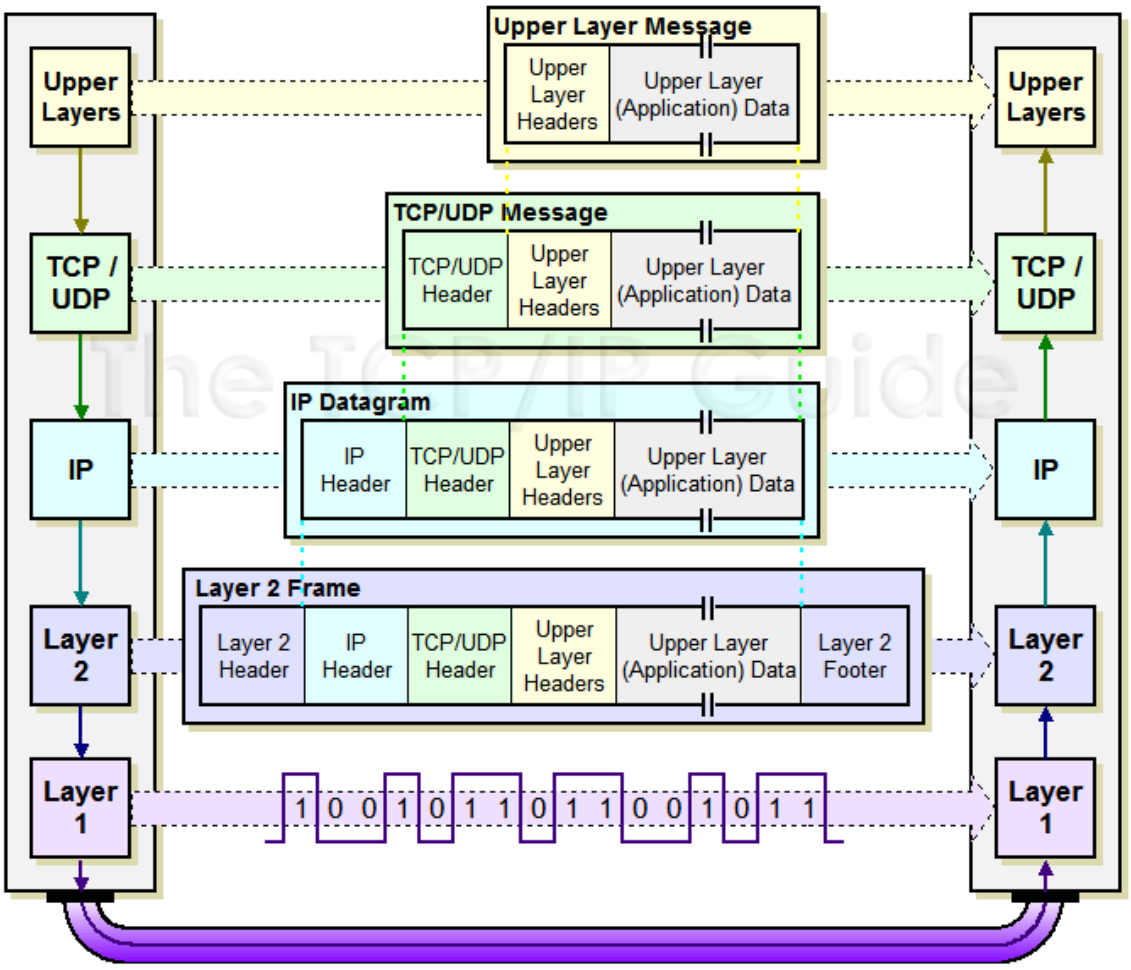
Para poder hacer esto, las capas adyacentes tienen lo que se llama una **interfaz**. En este contexto la interfaz definirá las operaciones y servicios que la capa inferior ofrece a la superior.

Cuando los diseñadores, diseñadoras, o fabricantes quieren fabricar productos compatibles, deben seguir los estándares de la arquitectura de red, para esto es importante definir interfaces claras entre niveles y que cada nivel tenga bien definidos sus servicios.

Todo esto implica que para un buen funcionamiento de la red se deben respetar ciertas reglas, como, por ejemplo: que los servicios se definan mediante protocolos estándares, que cada nivel sólo se comunique con el nivel superior o el inferior y que cada nivel inferior proporcione servicios a su nivel superior.

Hay que comentar que este tipo de arquitectura por niveles conlleva que cada nivel genera su propio conjunto de datos, ya que cada capa pasa los datos originales junto con la información que ella genera, para así poder controlar la comunicación por niveles. Esta información para los niveles inferiores se trata como si fueran datos, ya que sólo la utilizará el nivel correspondiente del ordenador de destino. Más adelante veremos los diferentes nombres que tienen estos datos según la arquitectura que se utilice.

Para terminar, destacar que las arquitecturas de red basadas en capas facilitan las compatibilidades, tanto de software como de hardware, así como las modificaciones futuras, ya que no es necesario cambiar todas las capas cuando queremos mejorar el sistema. Bastaría modificar los protocolos por niveles y podríamos conseguir mejoras en el sistema.



1.2. PROTOCOLO DE COMUNICACIÓN.

Un protocolo de comunicaciones es un **conjunto de reglas normalizadas** para la representación, señalización, autenticación y detección de errores necesario para enviar información a través de un canal de comunicación.

Entre los protocolos necesarios para poder establecer una comunicación necesitamos protocolos para:

- Identificar el emisor y el receptor.
- Definir el medio o canal que se puede utilizar en la comunicación.
- Definir el lenguaje común a utilizar.
- Definir la forma y estructura de los mensajes.
- Establecer la velocidad y temporización de los mensajes.
- Definir la codificación y encapsulación del mensaje.

Los protocolos usados en las redes están adaptados a las características del emisor, el receptor y el canal, además los protocolos deben definir los detalles de cómo transmitir y entregar un mensaje.

Si nos centramos en las redes de ordenadores, podemos definir algunas cuestiones que los protocolos de redes deben resolver, estas cuestiones serán:

- **El enrutamiento**: En las redes de ordenadores pueden tenerse diferentes rutas para llegar a un mismo destino, por tanto, debe elegirse una de ellas, siendo deseable que siempre se elija la mejor o más rápida. Por tanto, las arquitecturas de red deben tener protocolos que sirvan para este fin, ya veremos cuáles son y en qué nivel se resuelven.
- **El direccionamiento**: Dado que una red se compone de muchos nodos conectados entre sí, debe haber alguna forma de conocer cuál es cual. Para esto necesitamos definir direcciones de red que permitan determinar a qué ordenador me quiero conectar o por dónde debo conectarme para llegar a un destino. Para poder conseguir esto, las arquitecturas de red definen protocolos de direccionamiento, desde un punto de vista lógico y físico, que se definen en niveles adecuados para que la comunicación sea posible, y no se produzcan duplicidades.
- **La necesidad de compartir un medio de comunicaciones**: Puede darse el caso que se comparta un mismo medio para transmitir, por tanto, deben establecerse mecanismos que controlen el acceso al medio y el orden en el que se accede.
- **La saturación**: Los protocolos de cualquier nivel deben ser capaces de evitar que el receptor del mensaje, o los dispositivos intermedios que actúan en la transmisión del mensaje, se saturen. Esto suele ser un problema, y no siempre es fácil de resolver, pero un buen diseño y la adecuación de la red a las necesidades ayudan.
- **El control de errores**: Es deseable que los protocolos de red tengan mecanismos de control de errores. Como veremos cuando analicemos las arquitecturas de red este control se puede hacer desde diferentes puntos de vista y en diferentes niveles.

Hemos citado algunas cuestiones, pero está claro que los protocolos resuelven muchas más, lo importante a tener en cuenta es que gracias a unos protocolos estandarizados, y a un buen diseño de red, podemos conseguir que ordenadores de todo el mundo se comuniquen entre sí.

1.3. FUNCIONAMIENTO DE UNA ARQUITECTURA BASADA EN NIVELES.

El **modelo OSI**, siglas en inglés de Open System Interconnection o traducido, Interconexión de Sistemas Abiertos, es el modelo de red creado por la Organización Internacional para la Normalización (ISO) en el año 1984. Este modelo define un marco

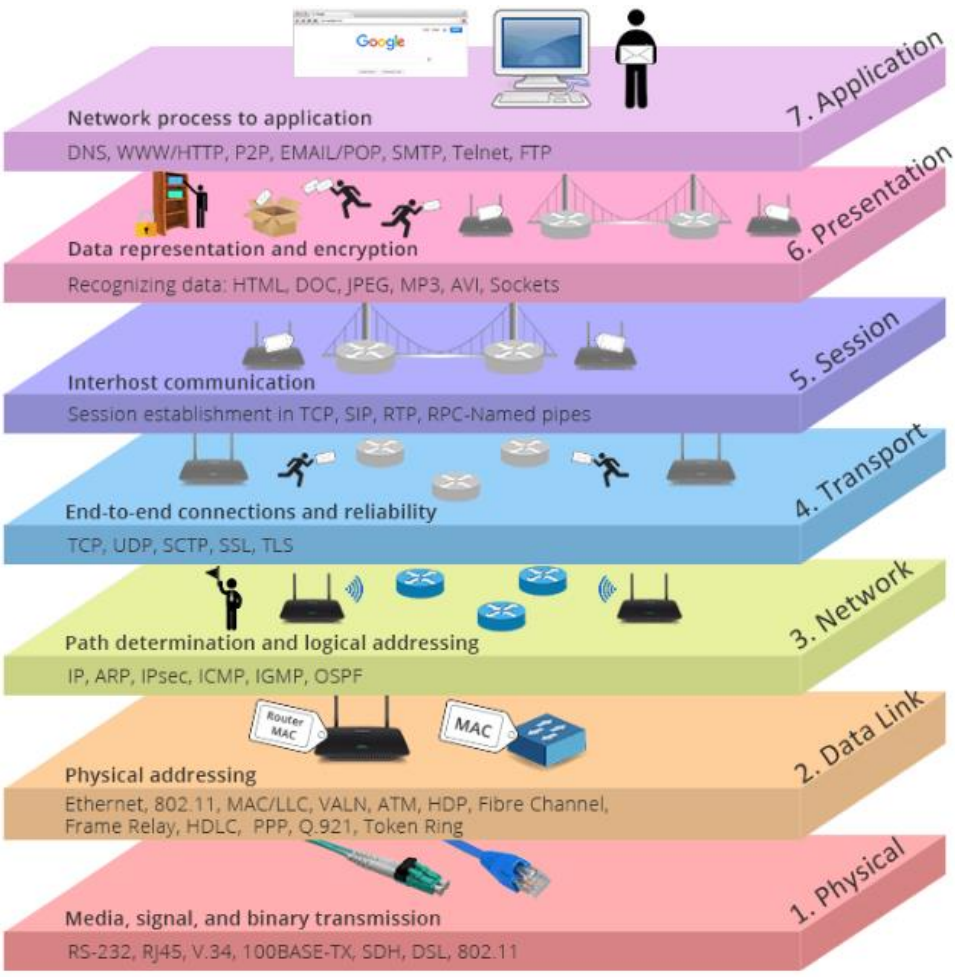
de referencia para la definición de arquitecturas de interconexión de sistemas de comunicaciones. Hay que destacar que el modelo OSI simplifica las actividades de red, ya que agrupa los procesos de comunicación en siete capas que realizan tareas diferentes. Es conveniente tener en cuenta que el modelo OSI, no es una arquitectura desarrollada en ningún sistema, sino una referencia para desarrollar arquitecturas de red, de forma que los protocolos que se desarrollen puedan ser conocidos por todos.

Aunque el modelo OSI no está realmente desarrollado en ningún sistema, sí es conveniente conocerlo y aplicarlo, ya que nos sirve para poder entender los procesos de comunicación que se producen en una red, y además puede usarse como referencia para realizar una detección de errores o un plan de mantenimiento.

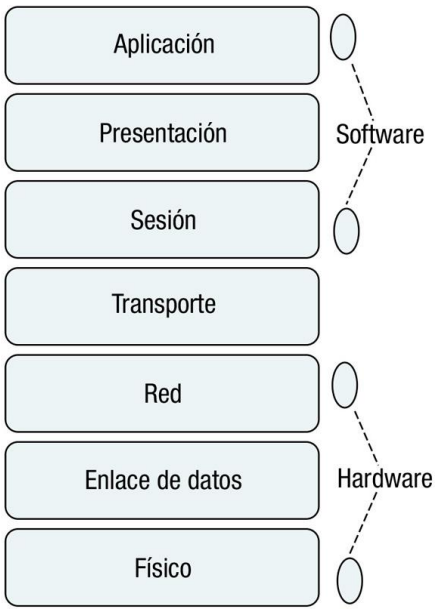
Los niveles OSI son:

Capa	Nombre	Funciones
1	Capa física o nivel físico.	Se encarga de las conexiones físicas, incluyendo el cableado y los componentes necesarios para transmitir la señal.
2	Capa o nivel de enlace de datos.	Empaqueta los datos para transmitirlos a través de la capa física. En esta capa se define el direccionamiento físico utilizando las conocidas direcciones MAC. Además se encarga del acceso al medio, el control de enlace lógico o LLC y de la detección de errores de transmisión, entre otras cosas.
3	Capa o nivel de red.	Separa los datos en paquetes, determina la ruta que tomaran los datos y define el direccionamiento.
4	Capa o nivel de transporte.	Se encarga de que los paquetes de datos tengan una secuencia adecuada y de controlar los errores.
5	Capa o nivel de sesión.	Mantiene y controla el enlace entre los dos extremos de la comunicación.
6	Capa o nivel de presentación.	Determina el formato de las comunicaciones así como adaptar la información al protocolo que se esté usando.
7	Capa o nivel de aplicación.	Define los protocolos que utilizan cada una de la aplicaciones para poder ser utilizadas en red.

La representación gráfica del modelo OSI, suele hacerse como una pila, donde en lo más alto estaría la capa 7 de aplicación y en lo más bajo la capa 1 o física.



Es conveniente mencionar que en ocasiones se hace referencia a que las capas 1, 2 y 3 del modelo están relacionadas con el hardware y las capas 5, 6 y 7 están relacionadas con el software, siendo la capa 4 una capa intermedia entre hardware y software. Esto suele ser así porque los dispositivos y componentes de red, suelen trabajar en los niveles 1 a 3, siendo los programas los que trabajan en los niveles superiores.



1.4. TCP/IP.

Cuando se habla de protocolos TCP/IP, realmente se suele estar haciendo referencia a la arquitectura de red que incluye varios protocolos de red, de entre los cuales dos de los más destacados son el protocolo TCP (Protocolo de Control de Transmisión) y el protocolo IP (Protocolo de Internet).

Por tanto, sería conveniente considerar este modelo como una arquitectura en sí, siendo la más utilizada, ya que es la base de las comunicaciones de Internet y de los sistemas operativos modernos.

Cuando nos referimos a la arquitectura TCP/IP o modelo TCP/IP, nos estamos refiriendo a un conjunto de reglas generales de diseño e implementación de protocolos de red, que permiten la comunicación de los ordenadores. Como veremos con más detalle durante esta unidad, existen protocolos para los diferentes tipos de servicios de red.

La arquitectura TCP/IP está compuesta de cuatro capas o niveles que son:

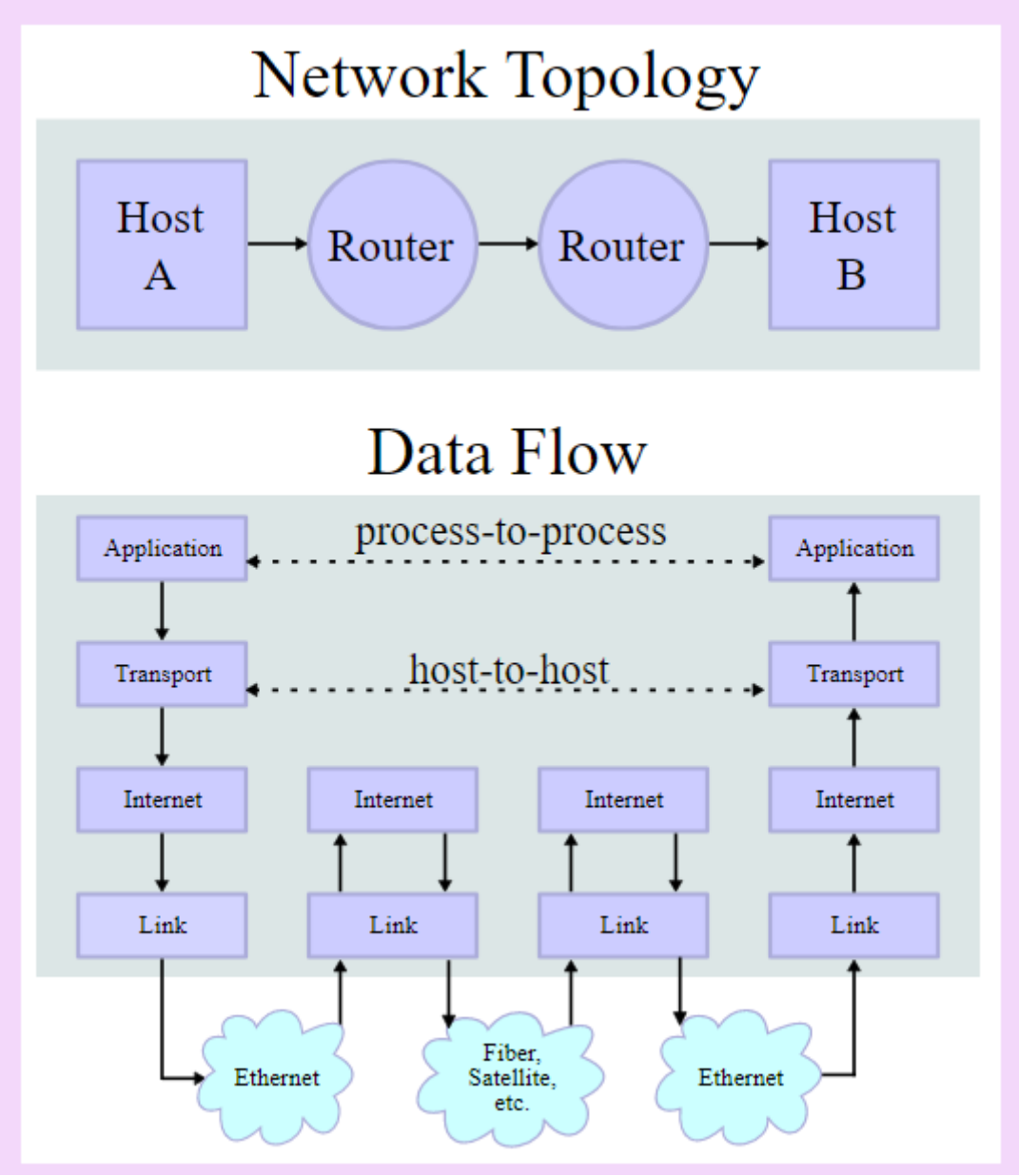
Capa	Nombre	Funciones
1	Capa o nivel de acceso a la red, de enlace o también llamado de subred.	Se encarga del acceso al medio de transmisión, es asimilable a los niveles 1 y 2 del modelo OSI, y sólo especifica que deben usarse protocolos que permitan la conexiones entre ordenadores de la red. Hay que tener en cuenta que esta arquitectura está pensada para conectar ordenadores diferentes en redes diferentes, por lo que las cuestiones de nivel físico no se tratan, y se dejan lo suficientemente abiertas para que se pueda utilizan cualquier estándar de conexión. Permite y define el uso de direcciones físicas utilizando las direcciones MAC.
2	Capa o nivel de red también llamada de Internet.	Al igual que la capa de red del modelo OSI, esta capa se encarga de estructurar la información en paquetes, determina la ruta que tomaran los paquetes y define el direccionamiento. En esta arquitectura los paquetes pueden viajar hasta el destino de forma independiente, pudiendo atravesar redes diferentes y llegar desordenados, sin que la ordenación de los paquetes sea responsabilidad de esta capa, por tanto tampoco se encarga de los errores. El protocolo más significativo de esta capa es el protocolo IP, y entre sus funciones está la de dar una dirección lógica a todos los nodos de la red.
3	Capa o nivel de transporte.	Es igual al nivel de transporte del modelo OSI. Se encarga de que los paquetes de datos tengan una secuencia adecuada y de controlar los errores. Los protocolos más importantes de esta capa son: TCP y UDP. El protocolo TCP es un protocolo orientado a conexión y fiable, y el protocolo UDP es un protocolo no orientado a conexión y no fiable.
4	Capa o nivel de aplicación.	Esta capa englobaría conceptos de las capas de sesión, presentación y aplicación del modelo OSI. Incluye todos los protocolos de alto nivel relacionados con las aplicaciones que se utilizan en Internet.

Una comparativa de esta arquitectura con el modelo OSI podemos verla en el siguiente gráfico.

Capas según el modelo OSI		Capas según el modelo TCP/IP	
7	Aplicación *Application*	4	Aplicación *Process*
6	Presentación *Presentation*		
5	Sesión *Session*		
4	Transporte *Transport*	3	Transporte *Host-to-Host*
3	Red *Network*	2	Internet *Network*
2	Enlace de datos *Data Link*	1	Acceso al medio *Media*
1	Física *Physical*		

La arquitectura TCP/IP se estructura en capas jerarquizadas y es el utilizado en Internet, por lo que en algunos casos oiréis hablar de Familia de Protocolos de Internet refiriéndose a está arquitectura cuando trabaja en Internet.

Es conveniente recordar que en algunos casos se divide la capa de acceso a la red, en capa de hardware o física y enlace de datos, con lo que la arquitectura tendría cinco niveles en vez de cuatro. Esto suele hacerse en referencia al modelo OSI. En realidad, esto se puede hacer y no cambiaría la estructura de la arquitectura.



1.5. EL NIVEL DE ACCESO A LA RED

La arquitectura TCP/IP en su estandarización original no se preocupaba demasiado del nivel físico en sí, de hecho, en un principio sólo se preocupó de estandarizar los protocolos relacionados con el enlace de datos, de ahí el nombre de este nivel.

Posteriormente con el auge de las redes de todo tipo, se vio que los estándares que ya existían desde un punto de vista físico, cada vez se tenían que tener más en cuenta, y por esto algunos autores, desarrolladores y diseñadores consideran que la arquitectura TCP/IP realmente consta de cinco capas, siendo la primera la capa física o de hardware y la segunda la de enlace de datos, tal y como recomienda el modelo OSI.

Para nosotros nos basta con considerarla como una sola, tal y como viene referido en el RFC 1122, documento que define el modelo TCP/IP.

La principal función de este nivel es convertir la información suministrada por el nivel de red, en señales que puedan ser transmitidas por el medio físico. La función inversa es convertir las señales que llegan por el medio físico en paquetes de información manejables por el nivel de red.

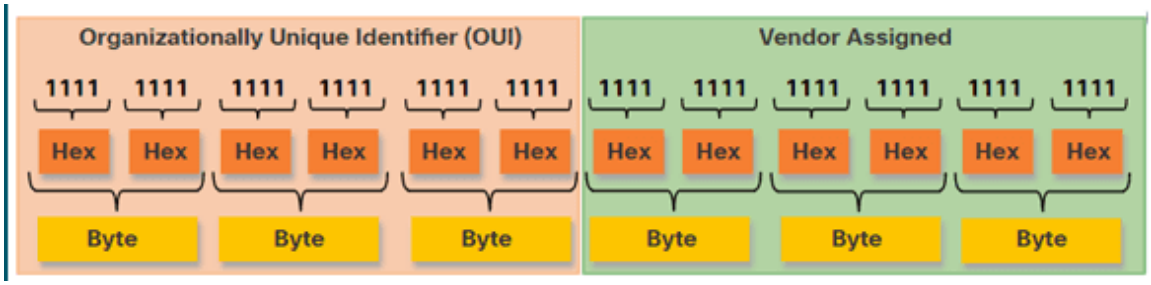
En este nivel se deben tener en cuenta las cuestiones relacionadas con las conexiones físicas, que en las redes locales vienen definidas por el estándar Ethernet. Este estándar define las características de cableado y señalización de nivel físico, y los formatos de las tramas de datos del nivel de enlace de datos. Ethernet es la base para el estándar **IEEE 802.3**, que es un estándar internacional que tiene posibilidades de uso tanto en redes locales como en redes de área amplia.

Otro aspecto importante de este nivel es lo relacionado con el direccionamiento físico. Este concepto viene de lo que se considera una subcapa del nivel de enlace de datos, y que se llama control de acceso al medio, cuyas siglas en inglés, MAC, se utilizan para definir lo que se conoce como direcciones MAC.

La dirección MAC es un identificador de 48 bits, que suele representarse en forma de números hexadecimales, en un formato de 6 bloques de dos números hexadecimales, divididos por dos puntos. El formato es el siguiente:

FF:FF:FF:FF:FF:FF

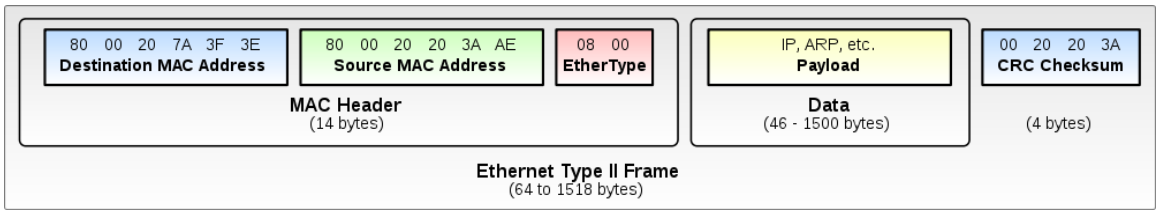
Los 24 bits más significativos (los de la izquierda) determinan el fabricante y se les conoce como Identificador Único de Organización y los 24 bits menos significativos (los de la derecha), identifican una interfaz concreta. De esta forma ninguna tarjeta de red tiene la misma dirección física.



En este nivel hay un protocolo relacionado con el direccionamiento físico. Este protocolo es el **ARP**.

ARP son las siglas en inglés del **protocolo de resolución de direcciones**, este protocolo trabaja a nivel de enlace de datos y se encarga de encontrar la dirección física o MAC que tiene relación con la correspondiente dirección lógica, que, como veremos en el siguiente apartado, se corresponde con la dirección IP. Lo que hace ARP es traducir direcciones lógicas (IP) a direcciones físicas (MAC). Existe su inverso el RARP que son las siglas en inglés del protocolo de resolución de direcciones inverso, hace la función inversa del protocolo ARP, pero no es tan utilizado.

Para terminar, mostramos el formato de la unidad de información de este nivel. Cada nivel tendrá una unidad de información, en este nivel se llama **TRAMA,** y tiene un formato determinado.



Sólo destacaremos que en la trama tenemos los datos que recibimos de las capas superiores, y que la capa de enlace le agrega una cabecera, con las direcciones MAC origen y destino, junto con el tipo de trama Ethernet que se utiliza, y una cola donde se agrega información para el control de errores.

1.5.1. Obtención de MAC en distintos sistemas operativos

Windows 2000/XP/Vista/7/8/10/11

En el entorno Windows la Dirección MAC se conoce como «dirección física». La manera más sencilla es abrir una terminal de línea de comandos («cmd» desde Inicio>Ejecutar) y allí usar la instrucción: `ipconfig /all`, o también se puede usar el comando `getmac`.

UNIX, GNU/Linux y Mac OS X

En el entorno de familia *nix (Mac Os X está basado en UNIX), habrá que abrir un terminal y ejecutar el comando: **ifconfig**. Esto nos muestra las interfaces seguidas de sus respectivas direcciones MAC en el epígrafe ether. (Nota: para ejecutar "ifconfig" algunas distribuciones requieren que se tengan privilegios de root: "sudo ifconfig").

Usando el paquete iproute2, es posible obtener las direcciones MAC de todas las tarjetas ethernet: "ip link list".

Tanto en Mac OS X 10.5, 10.7 o 10.9, para saber la dirección MAC basta con ir a Preferencias del Sistema > Red y dentro del apartado Wi-FI darle al botón Avanzado... En la ventana que saldrá, abajo del todo vendrá la dirección Wifi correspondiente a nuestro ordenador.

1.6. EL NIVEL DE INTERNET O DE RED.

El nivel de red del modelo TCP/IP se considera el nivel de la arquitectura más importante, ya que permite que las estaciones envíen información a la red en forma de paquetes. Estos paquetes viajan por la red de forma independiente, pudiendo atravesar diferentes redes y sin un orden establecido. Ésta es una de las principales ventajas de esta arquitectura y por eso es la base de Internet.

El objetivo principal del nivel de red será encaminar los paquetes desde el nodo origen hasta el nodo destino.

En la arquitectura TCP/IP la capa de red es casi totalmente asimilable a la capa de red del modelo OSI, pero en el caso de la arquitectura TCP/IP la capa de red no se preocupa de las tareas de ordenación de los paquetes cuando llegan a su destino. Esto es lo que se conoce como servicio no orientado a conexión. Cuando los paquetes se tratan de forma independiente, conteniendo cada uno la dirección de destino, se dice que se usa la técnica de datagrama, por tanto, Internet es una red de conmutación de paquetes basada en datagramas.

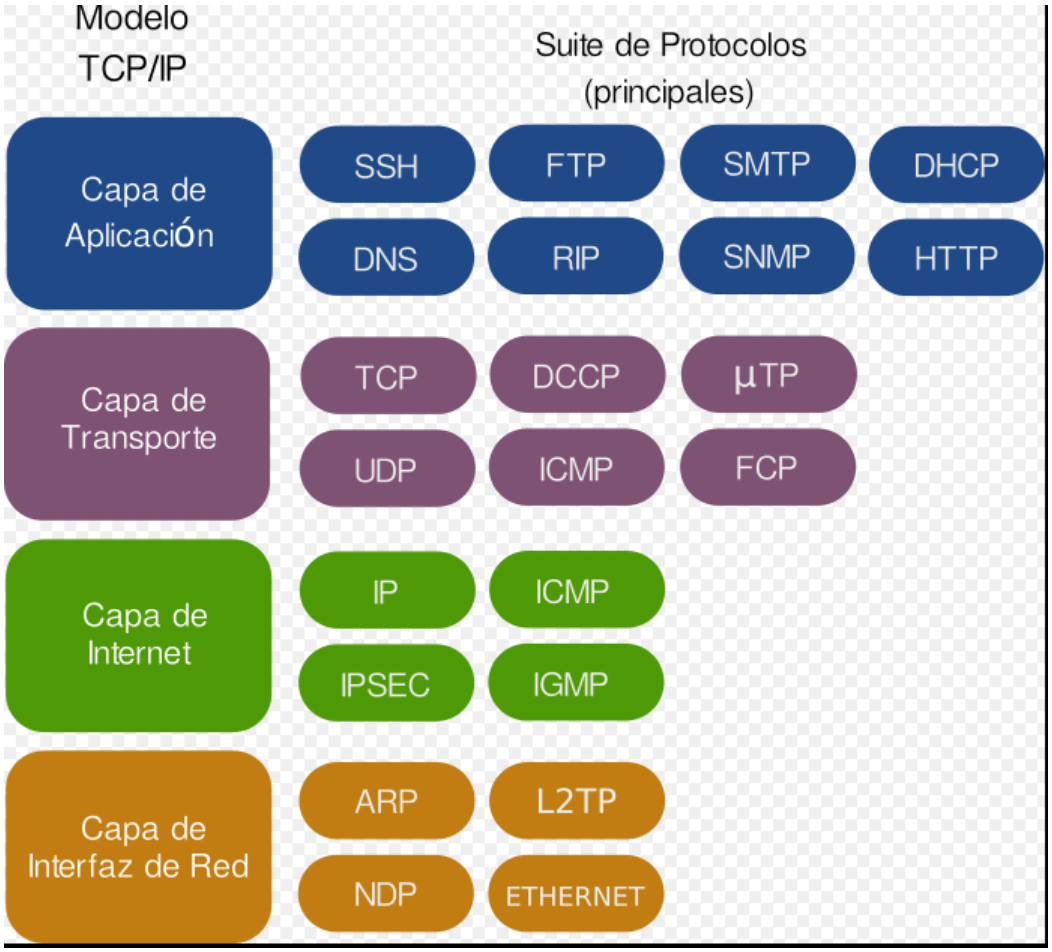
Entre las funciones de la capa de red se encuentra:

- **El direccionamiento:** Permite identificar de forma única cada nodo de la red. Cuando se habla de direccionamiento en este nivel, se está hablando de direccionamiento lógico, para distinguirlo del direccionamiento físico que ya hemos visto anteriormente.
- **La conectividad:** Conseguir que los nodos de una red se conecten, independientemente de la red a la que pertenezcan.
- **El enrutamiento:** También llamado encaminamiento, los protocolos de esta capa deben ser capaces de encontrar el mejor camino entre dos nodos.
- **El control de la congestión:** Es conveniente realizar un control del tráfico, ya que si un nodo recibe más información de la que puede procesar, se produce una saturación y este problema puede extenderse a toda la red.

Para realizar todas estas funciones el nivel de red utiliza diferentes protocolos, entre los protocolos más destacados de este nivel tenemos:

- **IP:** Internet Protocol, o Protocolo de Internet proporciona un enrutamiento de paquetes no orientado a conexión y es usado tanto por el origen como por el destino para la comunicación de datos.
- **ARP y RARP:** También se utilizan en la capa de enlace de datos y sirven para relacionar direcciones IP con direcciones MAC y viceversa.
- **ICMP:** Protocolo de mensajes de control en Internet, suministra capacidades de control y envío de mensajes. También se considera protocolo del nivel de transporte, y herramientas tales como ping y tracert lo utilizan para poder funcionar.

- **OSPF:** Es un protocolo de enrutamiento que busca el camino más corto entre dos nodos de la red.
- **RIP:** Protocolo de enrutamiento de información, al igual que OSPF, también busca el camino más corto, pero utilizando otras técnicas de enrutamiento.



1.6.1. IP (Internet Protocol)

Como se puede comprobar este nivel tiene varias funciones, y varios protocolos, pero podemos decir que el más importante de todos, de hecho, da nombre a la arquitectura, es el protocolo IP.

El protocolo IP, además de lo mencionado anteriormente, también proporciona las direcciones IP. Una dirección IP es un número que identifica de manera lógica y jerárquicamente a una interfaz dentro de una red que utilice el protocolo de Internet. Más adelante conocerás más sobre el direccionamiento IP, pero ahora es conveniente que conozcas que existen dos versiones IPv4 (IP versión 4) e IPv6 (IP versión 6). Se diferencian en el número de bits que utilizan, versión 4 utiliza direcciones de 32 bits y la versión 6 utiliza direcciones de 128 bits.

El Protocolo de Internet (IP) es un componente fundamental de las redes de comunicación, incluyendo Internet. IP es un conjunto de reglas y protocolos que rigen la forma en que los datos se envían, enrutados y recibidos en una red de computadoras.

En el corazón del Protocolo de Internet se encuentran las direcciones IP. Estas son etiquetas numéricas únicas que se asignan a cada dispositivo conectado a una red IP. Existen dos versiones principales de direcciones IP: IPv4 (Protocolo de Internet versión 4) y IPv6 (Protocolo de Internet versión 6).

- **IPv4:** Utiliza direcciones de 32 bits y se representa en formato decimal separado por puntos, como 192.168.1.1. IPv4 es ampliamente utilizado, pero el número limitado de direcciones disponibles ha llevado a problemas de agotamiento de direcciones.

- **IPv6:** Utiliza direcciones de 128 bits y se representa en un formato hexadecimal, como 2001:0db8:85a3:0000:0000:8a2e:0370:7334. IPv6 se ha desarrollado para abordar la escasez de direcciones IPv4 y proporcionar un espacio de direcciones mucho más amplio.

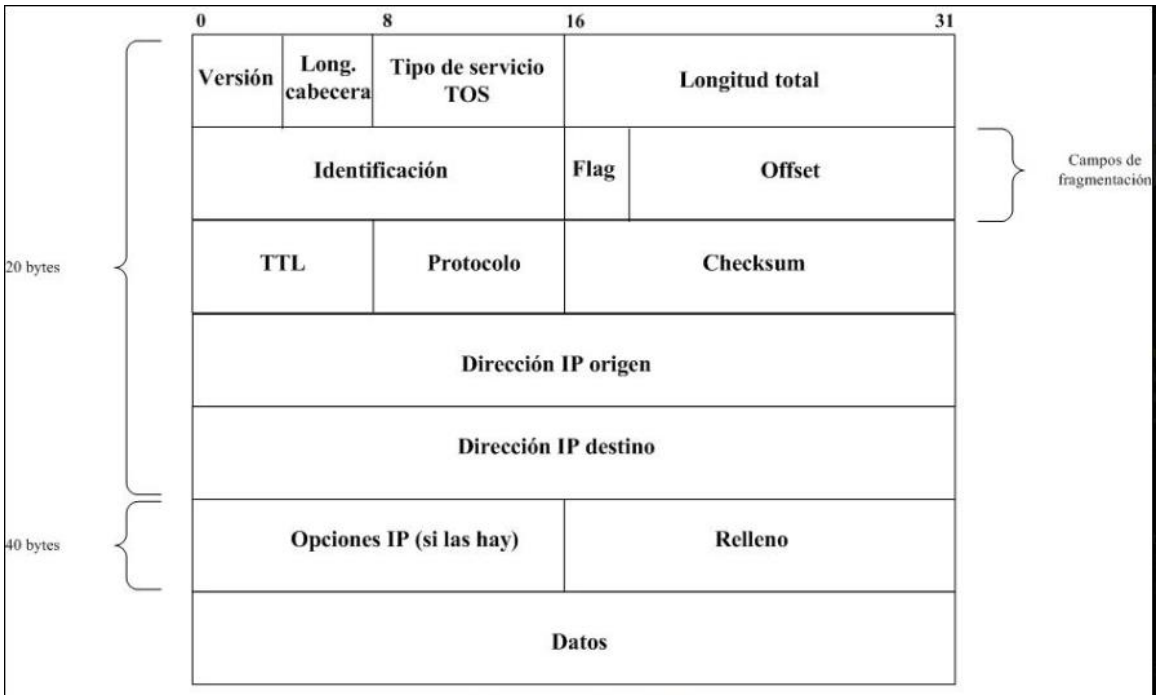
IP es esencial para el **enrutamiento de paquetes** en una red. Cada vez que un dispositivo envía datos a otro, esos datos se dividen en pequeños paquetes llamados "datagramas". Cada datagrama incluye la dirección IP del dispositivo de origen y del dispositivo de destino. Los enrutadores y conmutadores en la red utilizan esta información para transmitir los datagramas de un nodo a otro a través de la red, siguiendo la mejor ruta disponible.

El Protocolo de Internet es un protocolo **sin estado y sin conexión**, lo que significa que no mantiene un estado de conexión persistente entre los dispositivos. Cada datagrama se enruta de manera independiente, lo que permite una comunicación eficiente y escalable.

IP es un protocolo global que permite la comunicación a nivel mundial. Cada dispositivo en Internet tiene una dirección IP única, lo que facilita la identificación y la comunicación en la red global.

El Protocolo de Internet se utiliza en una amplia gama de aplicaciones, desde navegación web y correo electrónico hasta servicios en la nube, videoconferencias, juegos en línea, dispositivos de Internet de las cosas (IoT) y mucho más.

IPv4



IPv6

Los primeros 40 bytes (320 bits) son la cabecera del paquete y contiene los siguientes campos:

Offset del octeto	Bit offset	0								1								2								3							
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Versión				Clase de tráfico								Etiqueta de flujo																			
4	32	Longitud del campo de datos																Cabecera siguiente								Límite de saltos							
8	64	Dirección de origen																															
C	96																																
10	128																																
14	160																																
18	192	Dirección de destino																															
1C	224																																
20	256																																
24	288																																

- Direcciones de origen (128 bits)
- Direcciones de destino (128 bits)
- Versión del protocolo IP (4 bits)

- Clase de tráfico (8 bits, Prioridad del Paquete)
- Etiqueta de flujo (20 bits, manejo de la Calidad de Servicio)
- Longitud del campo de datos (16 bits)
- Cabecera siguiente (8 bits)
- Límite de saltos (8 bits, Tiempo de Vida).

1.7. EL NIVEL DE TRANSPORTE

Cumple la función de establecer las reglas necesarias para establecer una conexión entre dos dispositivos remotos. Al igual que las capas anteriores, la información que maneja esta capa tiene su propio nombre y se llama **segmento**.

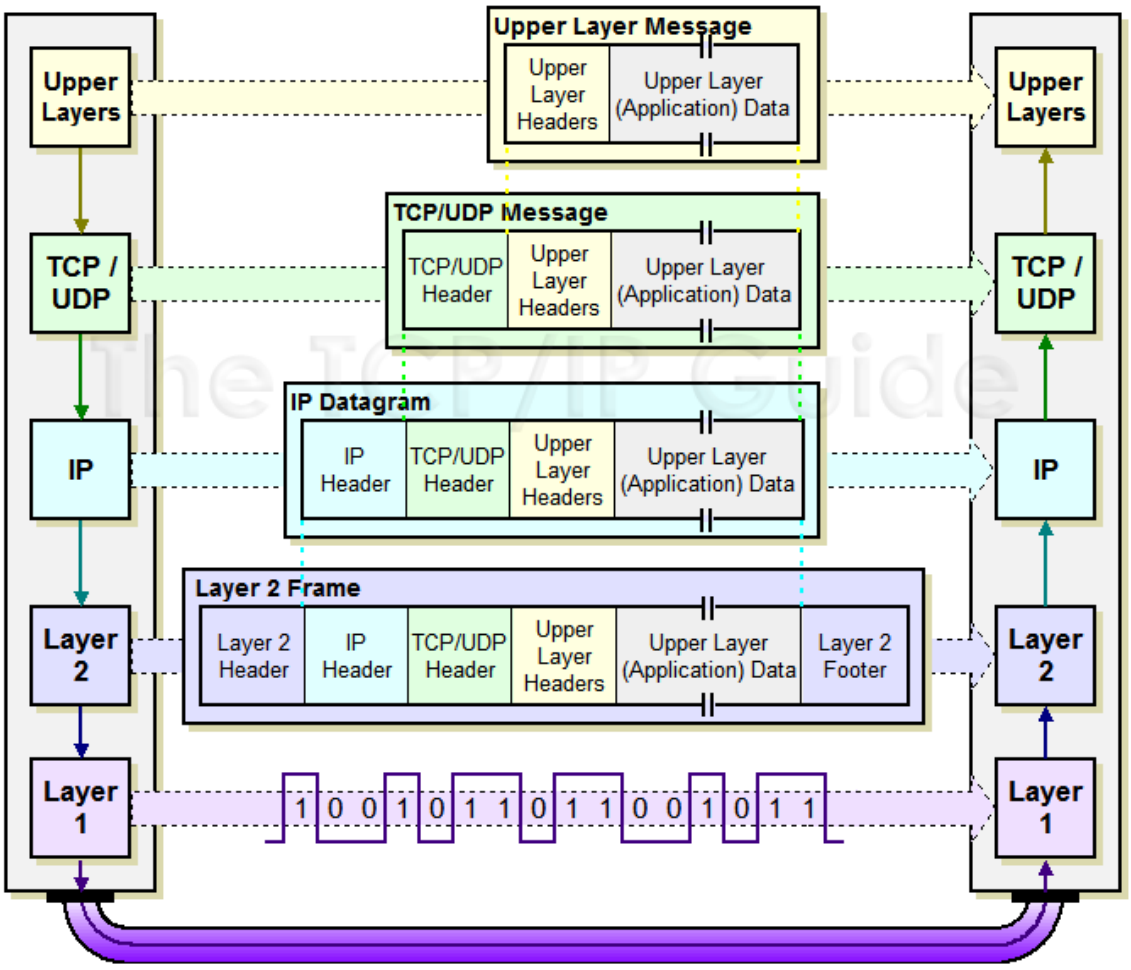
Por tanto, la capa de transporte se debe de encargar de unir múltiples segmentos del mismo flujo de datos. Como la capa de red en la arquitectura TCP/IP no se preocupa del orden de los paquetes ni de los errores, es en esta capa donde se deben cuidar estos detalles.

El nivel de transporte de la arquitectura de TCP/IP es totalmente asimilable al nivel de transporte del modelo OSI, por tanto, podemos decir que este nivel es el encargado de la transferencia libre de errores de los datos entre el emisor y el receptor, aunque no estén directamente conectados, así como de mantener el flujo de la red. La tarea de este nivel es proporcionar un transporte de datos confiable de la máquina de origen a la máquina destino, independientemente de la red física.

En este nivel trabajan varios protocolos, pero los dos más importantes son el **TCP** y el **UDP**. Como vimos anteriormente:

- **TCP es un protocolo orientado a conexión y fiable,** se diseñó específicamente para proporcionar un flujo de bytes confiable de extremo a extremo a través de redes no fiables. Por eso es tan útil en Internet, ya que a diferencia del tráfico en una sola red donde conoceremos sus características, las redes que configuran Internet podrían tener diferentes topologías, anchos de banda, retardos, tamaños de paquete, etc. Pero TCP tiene un diseño que se adapta de manera dinámica a las propiedades de estas redes y permite la conexión en muchos tipos de situaciones.

- **UDP es un protocolo no orientado a conexión y no fiable,** este protocolo proporciona todo lo necesario para que las aplicaciones envíen datagramas IP encapsulados sin tener una conexión establecida. Uno de sus usos es en la transmisión de audio y vídeo en tiempo real, donde no es posible realizar retransmisiones por los estrictos requisitos de retardo que se tiene en estos casos.



Cuando un proceso de aplicación quiere establecer comunicación con otro proceso de aplicación remoto, debe especificar a cuál se conectará. El método que normalmente se emplea es el de definir direcciones de transporte en las que los procesos pueden estar a la escucha de solicitudes de conexión. Estos puntos terminales se llaman **puertos**.

Por tanto, un puerto serán las direcciones de transporte en las que los procesos pueden estar a la escucha de solicitudes de conexión. El termino puerto se utiliza en Internet, el termino genérico es el de **Punto de Acceso al Servicio de Transporte**, cuyas siglas en inglés son TSAP.

Los números de puertos son utilizados por TCP y UDP para identificar las sesiones que establecen las distintas aplicaciones. Algunos puertos son:

- 20: datos de FTP (Protocolo de transferencia de ficheros).
- 21: control de FTP.
- 53: DNS (Servicio de nombres de dominio).
- 80: http (Protocolo utilizado para servir y descargar páginas web).

Puertos: Service Name and Transport Protocol Port Number Registry (iana.org)

1.8. EL NIVEL DE APLICACIÓN

El nivel aplicación contiene los programas de usuario (aplicaciones) que hace que nuestro ordenador pueda crear textos, chatear, leer correo, visitar páginas web, etc.

En este nivel se incluyen todos los protocolos de alto nivel que utilizan los programas para comunicarse.

En la arquitectura TCP/IP este nivel incluye a los niveles de sesión, presentación y aplicación del modelo OSI.

Algunos de los protocolos de la capa de aplicación son:

- **FTP**: Protocolo utilizado en la transferencia de ficheros entre un ordenador y otro.
- **DNS:** Servicio de nombres de dominio, es el sistema utilizado en Internet para convertir los nombres de los nodos de red en direcciones de red.
- **SMTP**: Protocolo simple de transferencia de correo, basado en texto y utilizado para el intercambio de mensajes de correo. Está basado en el concepto cliente-servidor, donde un cliente envía un mensaje a uno o varios servidores.
- **POP**: Protocolo de oficina de correo, se utiliza en los clientes de correo para obtener los mensajes de correo almacenados en un servidor.
- **SNMP**: Protocolo de administración de redes, permite monitorizar y controlar los dispositivos de red y de administrar configuraciones y seguridad.
- **HTTP**: Protocolo de transferencia de hipertexto, es el protocolo utilizado en las transacciones de páginas web. Define la sintaxis y la semántica que utilizan los elementos de software de la arquitectura web (clientes, servidores, proxies) para comunicarse. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor. Tiene una versión segura que es el HTTPS.

1.8.1. Socket

Una vez que conocemos los diferentes niveles de la arquitectura podemos definir el concepto de socket. Un socket, es una conexión que está formada por la unión de la dirección IP más el puerto que se utiliza para la conexión.

Un socket se utiliza para establecer conexiones de red, enviar y recibir datos, y permite que las aplicaciones se comuniquen entre sí, ya sea en la misma máquina o a través de una red, como Internet.

Un socket actúa como un punto de comunicación que permite que dos dispositivos, como computadoras, se conecten y se comuniquen entre sí a través de una red. Cada socket tiene una dirección única que consiste en una dirección IP y un número de

puerto. Ejemplo: 192.168.1.11:80, esto significa que el ordenador cuya dirección es 192.168.1.11 está utilizando el puerto 80, que está asociado al protocolo http del nivel de aplicación, por tanto, esto puede significar que el ordenador está visitando una página web o sirviendo una página web

Los sockets se utilizan en aplicaciones de red que utilizan protocolos de comunicación como TCP (Protocolo de Control de Transmisión) o UDP (Protocolo de Datagramas de Usuario). Estos protocolos definen cómo se establece, se mantiene y se cierra una conexión, así como cómo se envían y reciben los datos.

Los sockets pueden ser de varios tipos, incluyendo sockets de escucha (listening sockets) y sockets de conexión (connected sockets). Los sockets de escucha se utilizan para esperar nuevas conexiones entrantes, mientras que los sockets de conexión se utilizan para transmitir datos una vez que se ha establecido una conexión.

Cada socket está asociado con una dirección IP y un número de puerto. La dirección IP identifica el dispositivo o servidor en la red, mientras que el número de puerto identifica una aplicación o servicio específico en ese dispositivo. Esto permite que **múltiples aplicaciones se comuniquen en un solo dispositivo utilizando diferentes números de puerto.**

Los sockets permiten la comunicación bidireccional, lo que significa que tanto el emisor como el receptor pueden enviar y recibir datos a través del socket. Esto permite la interacción y el intercambio de información entre las aplicaciones en dos extremos de la conexión.

Los sockets se utilizan en una amplia variedad de aplicaciones, desde navegadores web que se conectan a servidores para cargar páginas web (utilizando sockets TCP) hasta aplicaciones de mensajería instantánea y juegos en línea que utilizan sockets UDP para transmitir datos en tiempo real.

En resumen, un socket es un mecanismo esencial para permitir la comunicación entre dispositivos a través de una red utilizando protocolos de comunicación como TCP o UDP. Permite que las aplicaciones se conecten, envíen y reciban datos de manera eficiente, lo que es fundamental para la programación de redes y la comunicación en aplicaciones distribuidas.

Este concepto seguro que te será de utilidad más adelante cuando programes servicios web o aplicaciones que utilicen Internet.