Sumario

UT 01: Conceptos de redes e interconexión a WAN

Este capítulo presenta de forma muy resumida algunos conceptos de red, necesarios para abordar el resto de las unidades de trabajo con una base. Los conocimientos de Redes se adquieren en el primer curso del ciclo formativo de SMR, por lo que se supone que partimos de un nivel de conocimientos básico, suficiente para configurar y administrar pequeñas redes.

.1 Introducción

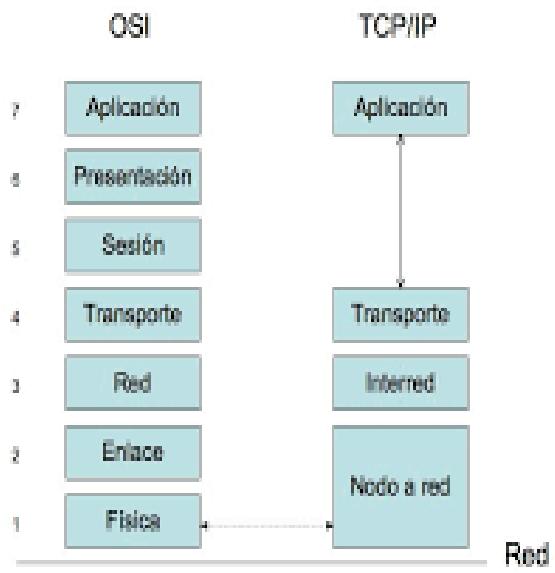
Este capítulo presenta de forma muy resumida algunos conceptos de red, necesarios para abordar el resto de las unidades de trabajo con una base. Los conocimientos de Redes se adquieren en el primer curso del ciclo formativo de SMR, por lo que se supone que partimos de un nivel de conocimientos básico, suficiente para configurar y administrar pequeñas redes.

.2 Repaso de los modelos OSI y TCP/IP

El modelo OSI

OSI (Open Systems Interconnection) es el nombre que se le dió a los modelos teóricos de arquitectura de red cuando se empezó a pensar en sistematizar su diseño, en los primeros tiempos de la Informática.

Consiste en un modelo de 7 capas con sucesivos niveles de abstracción, en los que cada una de ellas se encarga de realizar unas funciones determinadas y transferir la información a su nivel homólogo en otro sistema informático, o a los niveles adyacentes en el mismo sistema. Estas funciones se realizan mediante **protocolos** que siguen unas normas.



En la práctica, el modelo se simplificó al implementar internet con la arquitectura TCP/IP, en la que la que varias capas se integran en una sola, la de Aplicación.

El modelo OSI se representa con las siguientes capas:

Capa física

Es la primera capa del Modelo OSI. Es la que se encarga de la topología de red y de las conexiones globales del ordenador hacia la red, referida tanto al medio físico como a la forma en la que se transmite la información.

Sus principales funciones se pueden resumir como:

- Definir el medio o medios físicos por los que va a viajar la comunicación: cable de pares trenzados, cable coaxial, guías de onda, aire, fibra óptica.
- Definir las características materiales (componentes y conectores mecánicos) y eléctricas (niveles de tensión) que se van a usar en la transmisión de los datos por los medios físicos.
- Definir las características funcionales de la interfaz (establecimiento, mantenimiento y liberación del enlace físico).
- Transmitir el flujo de bits a través del medio.
- Manejar las señales eléctricas del medio de transmisión, polos en un enchufe, etc.
- Garantizar la conexión (aunque no la fiabilidad de dicha conexión).

Capa de enlace de datos

Esta capa se ocupa del direccionamiento físico, del acceso al medio, de la detección de errores, de la distribución ordenada de tramas y del control del flujo. El dispositivo que usa la capa de enlace es el **Switch**, encargado de recibir las tramas de datos y enviarlas a sus respectivos destinatarios

Capa de red

Se encarga de identificar el enrutamiento existente entre una o más redes. Las unidades de datos se denominan paquetes. El objetivo de la capa de red es hacer que los datos lleguen desde el origen al destino, aun cuando ambos no estén conectados directamente. Los dispositivos que facilitan tal tarea se denominan encaminadores o **enrutadores**, aunque es más frecuente encontrarlo con el nombre en inglés **routers**. Los routers trabajan en esta capa, aunque pueden actuar como switch de nivel 2 en determinados casos, dependiendo de la función que se le asigne. Los firewalls actúan sobre esta capa para descartar direcciones de máquinas. En este nivel se realiza el direccionamiento lógico y la determinación de la ruta de los datos hasta su receptor final.

Capa de transporte

Capa encargada de efectuar el transporte de los datos (que se encuentran dentro del paquete) de la máquina origen a la de destino, independizándolo del tipo de red física que esté utilizando. La PDU de la capa 4 se llama Segmento o Datagrama, dependiendo de si corresponde a los protocolos TCP o UDP.

Capa de sesión

Esta capa es la que se encarga de mantener y controlar el enlace establecido entre dos computadores que están transmitiendo datos de cualquier índole. Por lo tanto, el servicio provisto por esta capa es la capacidad de asegurar que, dada una sesión establecida entre dos máquinas, la misma se pueda efectuar para las operaciones definidas de principio a fin, reanudándolas en caso de interrupción.

Capa de presentación

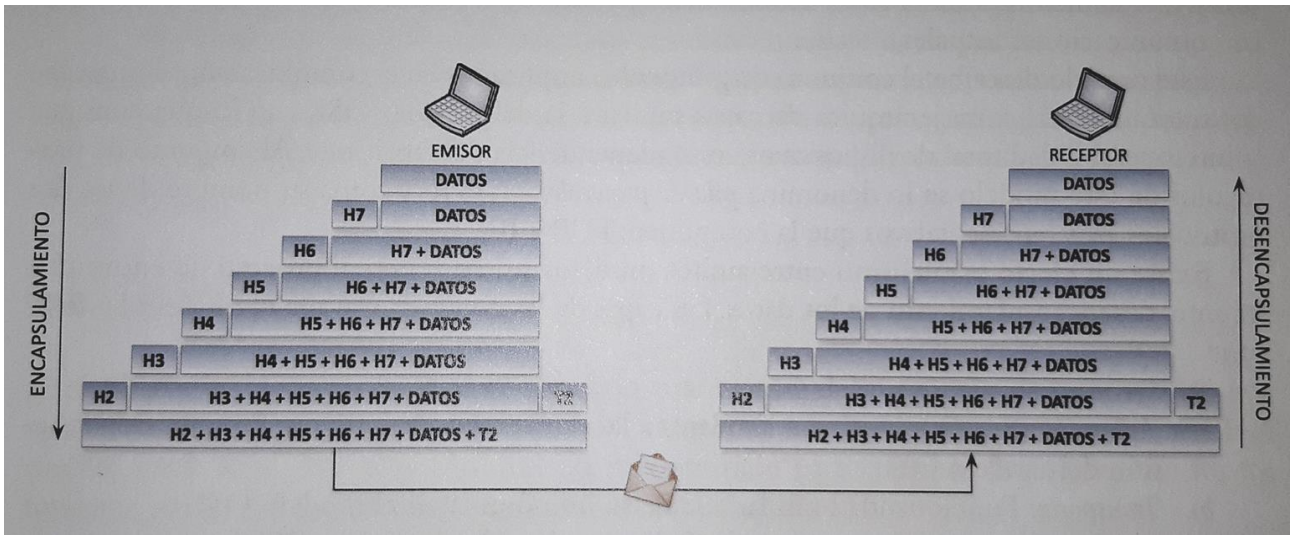
El objetivo es encargarse de la representación de la información, de manera que aunque distintos equipos puedan tener diferentes representaciones internas de caracteres, los datos lleguen de manera reconocible. Esta capa es la primera en trabajar más el contenido de la comunicación que el cómo se establece la misma. En ella se tratan aspectos tales

como la semántica y la sintaxis de los datos transmitidos, ya que distintas computadoras pueden tener diferentes formas de manejarlas. Esta capa también permite cifrar los datos y comprimirlos. Por lo tanto, podría decirse que esta capa actúa como un traductor.

Capa de aplicación

Ofrece a las aplicaciones la posibilidad de acceder a los servicios de las demás capas y define los protocolos que utilizan las aplicaciones para intercambiar datos, como correo electrónico (Post Office Protocol y SMTP), gestores de bases de datos y servidor de ficheros (FTP). Hay tantos protocolos como aplicaciones distintas y puesto que continuamente se desarrollan nuevas aplicaciones el número de protocolos crece sin parar. El módulo "Servicios en Red" consiste en el diseño, implementación y configuración de los diferentes servicios o aplicaciones que se apoyan en una red de ordenadores. El usuario normalmente no interactúa directamente con el nivel de aplicación. Suele interactuar con **programas,** que a su vez interactúan con el nivel de aplicación, ocultando la complejidad subyacente.

La información se empaqueta de capa a capa en bloques denominados "datagramas" o unidades de datos a los que se añade información (cabeceras) en un proceso denominado **encapsulamiento** (que requiere un **desencapsulamiento** en destino), como vemos en la siguiente gráfica:



El modelo Cliente/Servidor

El modelo "cliente/servidor" es el paradigma más empleado para diseñar y representar la

comunicación entre aplicaciones, y será en el que nos centremos en este módulo de "Servicios en Red".

Este modelo define la estructura de las aplicaciones que se comunican entre sí, así como su sincronización. Se representa mediante dos procesos que interactúan entre sí:

- El **proceso cliente**: inicia la comunicación, solicitando que se realice una determinada operación. Es el elemento activo, realiza la petición al proceso servidor y queda a la espera de una respuesta.
- El **proceso servidor** se encuentra inicialmente en modo pasivo, a la espera de que se realice una conexión de un cliente potencial. Debe ser un servicio robusto, dimensionado para ofrecer respuesta a peticiones en determinadas condiciones de capacidad de servicio, incluyendo aspectos como la seguridad y privacidad de la información.

Los servicios de red

Un servicio de red es una función o prestación ofrecida por una aplicación hacia los usuarios o hacia otras aplicaciones, a través de ciertos protocolos. Es importante no confundir los protocolos del nivel de aplicación con las apliaciones que los utilizan.

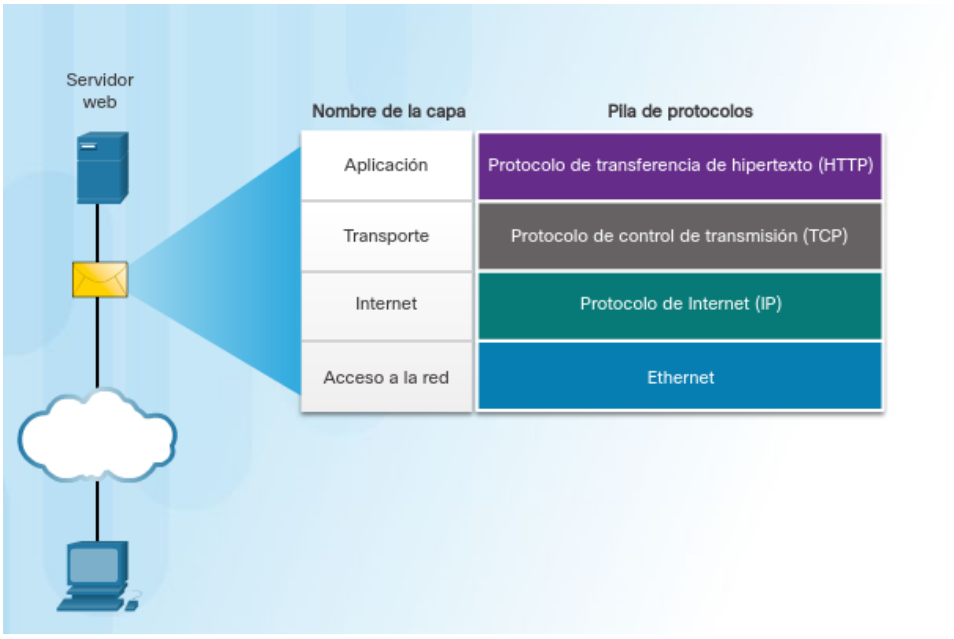
- Una **aplicación** es un programa instalado por el usuario o integrado en el sistema operativo, que se sirve de los protocolos de la arquitectura TCP/IP. Ejemplos: Opera, Internet Explorer, Outlook, Mozilla Firefox, etc...
- Un **protocolo** es un conjunto de normas formales, que detallan de qué manera se comunican los sistemas para ofrecer servicios en red. Algunos protocolos de nivel de aplicación son IMAP, HTTP, etc...

Ejemplos:

Servicio	Aplicación servidor	Aplicación cliente	Protocolos
Web	• Apache • IIS	• Firefox • Internet Explorer • Opera	• HTTP • HTTPS

		• Safari • Chrome	
Correo electrónico	• Exchange • Postfix • Sendmail	• Evolution • Outlook • Thunderbird	• POP • IMAP • SMTP

.3 El modelo TCP/IP. Diseño y configuración de Redes.



Muchos servicios de comunicaciones pueden representarse como una simplificación del modelo OSI, reduciendo el número de capas a 4: **Acceso (Ethernet), Red (Internet), Transporte y Aplicación.** Por ejemplo, el servicio web:

Los protocolos que se muestran en la figura son:

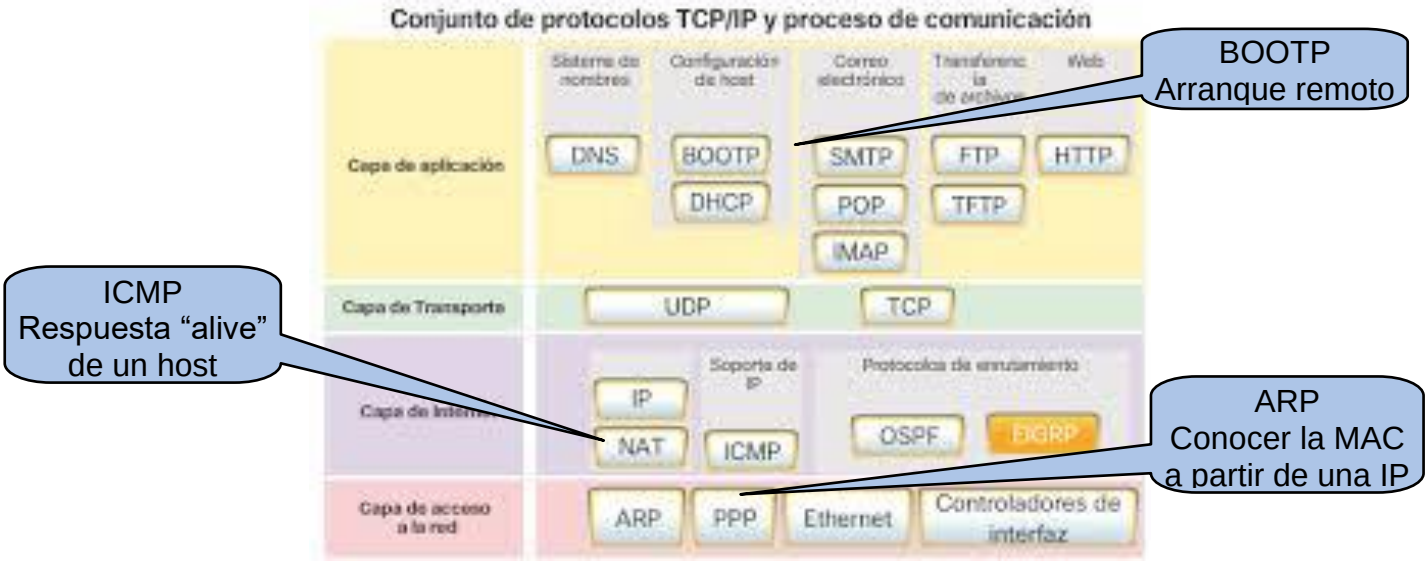
HTTP: es un protocolo de aplicación que rige la forma en que interactúan un servidor web y un cliente web. HTTP define el contenido y el formato de las solicitudes y respuestas intercambiadas entre el cliente y el servidor. Tanto el cliente como el software del servidor web implementan el HTTP como parte de la aplicación. HTTP se basa en otros protocolos para regular la forma en que se transportan los mensajes entre el cliente y el servidor.

TCP: es el protocolo de transporte que administra las conversaciones individuales. TCP divide los mensajes HTTP en partes más pequeñas, llamadas “segmentos”. Estos segmentos se envían entre los procesos del servidor y el cliente web que se ejecutan en el

host de destino. También es responsable de controlar el tamaño y los intervalos a los que se intercambian los mensajes entre el servidor y el cliente.

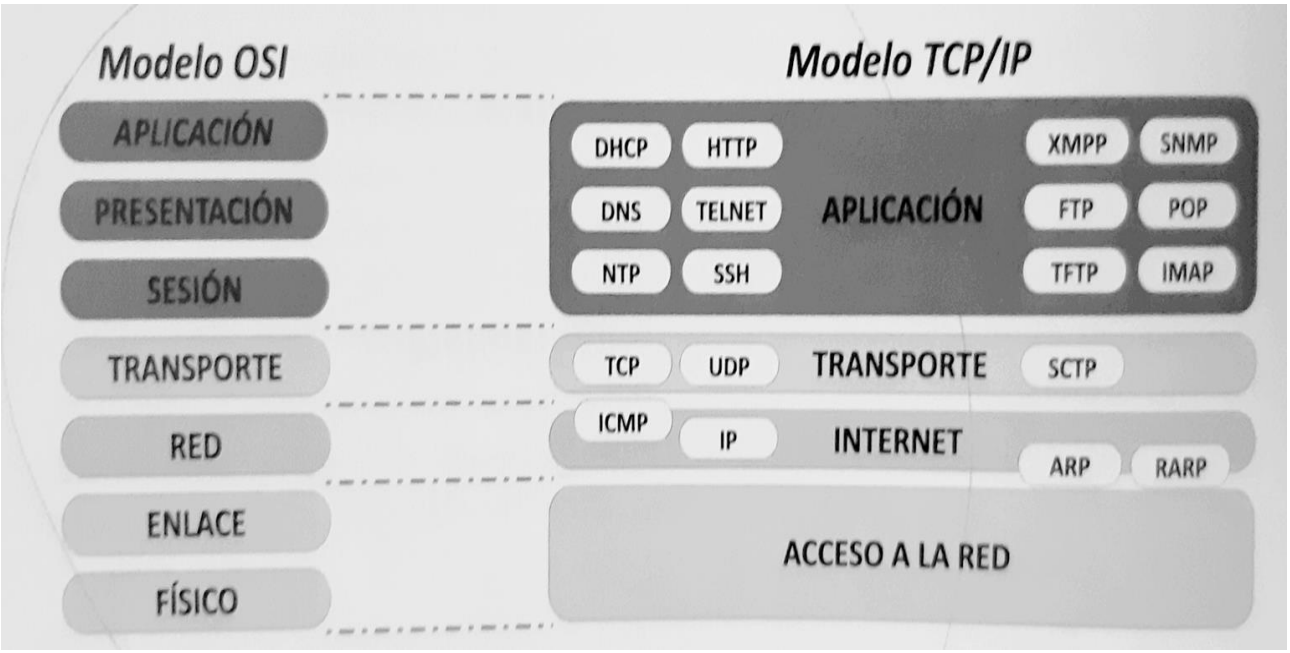
IP: responsable de tomar los segmentos formateados de TCP, encapsularlos en paquetes, asignar las direcciones apropiadas y seleccionar la mejor ruta al host de destino.

Ethernet: es un protocolo de acceso a la red que describe dos funciones principales: la comunicación a través de un enlace de datos y la transmisión física de datos en los medios de red. Los protocolos de acceso a la red son responsables de tomar los paquetes de IP y los formatean para transmitirlos por los medios.



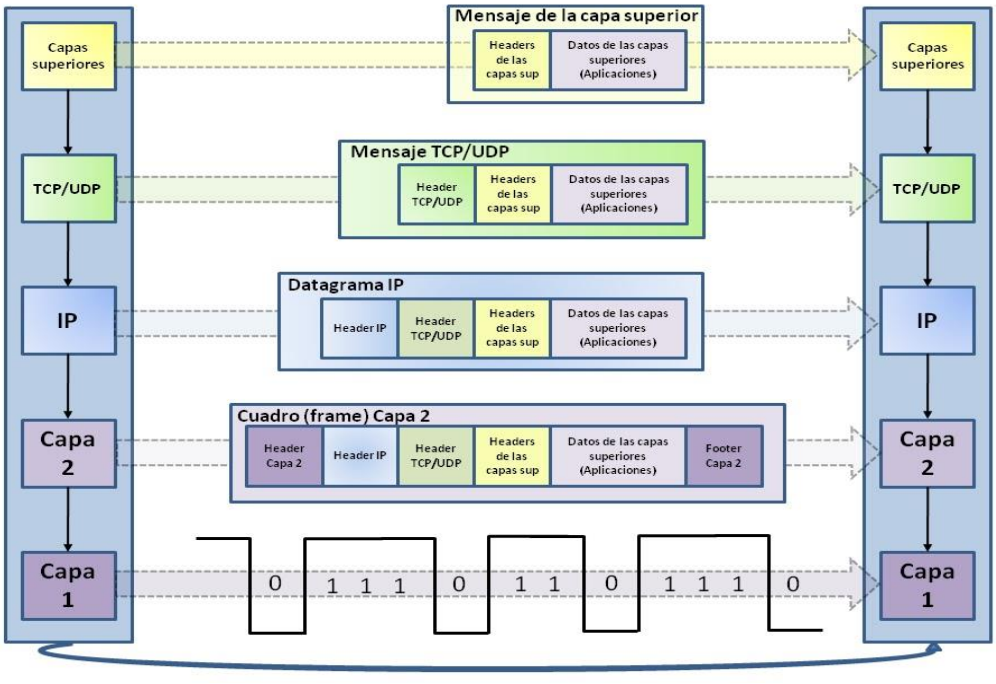
.3.1

La siguiente gráfica representa la equivalencia de protocolos TCP/IP con respecto a OSI:

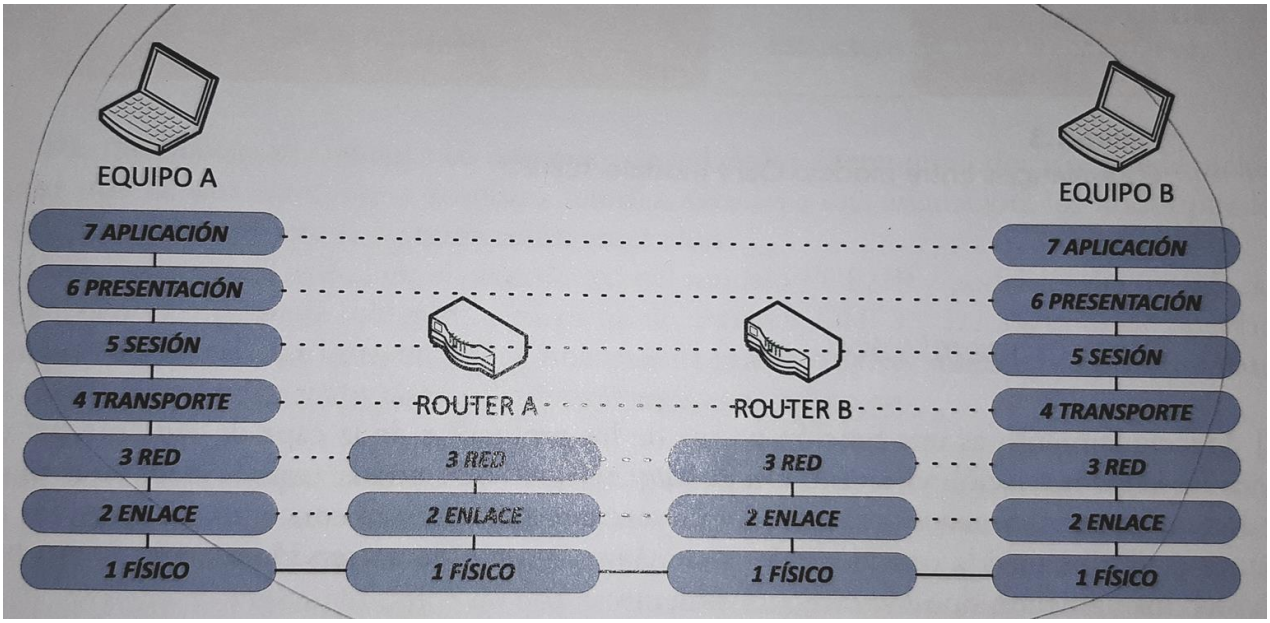


La siguiente figura representa el encapsulamiento de PDUs aplicado a la arquitectura TCP/IP de cuatro capas:

Encapsulamiento / Desencapsulamiento en un router (cuando la información se envía a una



red diferente):



Ethernet / Direccionamiento MAC (Nivel de Acceso)

Las tarjetas de red se conectan al medio físico (cableado de comunicaciones) a través de interfaces con una **dirección física**, llamada "**dirección MAC**" (Media Access Control). Esta dirección es permanente e identifica a cada una de las conexiones de red. A diferencia de la dirección de red (o dirección IP, que veremos más adelante), la dirección MAC presente desde el momento en que el dispositivo físico (ordenador, teléfono o cualquier periférico) comienza a arrancar y permite identificar a cada dispositivo, porque las tramas emitidas por él llevarán su dirección MAC en la cabecera de la trama.

Una dirección MAC está formada por 48 bits divididos en 6 bloques de 8 bits. Se suele representar en modo hexadecimal, por ejemplo:

```
$ ifconfig
wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.29  netmask 255.255.255.0  broadcast 192.168.0.255
        ether d4:25:8b:f5:13:ce  txqueuelen 1000  (Ethernet)
        (...)
```

Dirección MAC: **d4:25:8b:f5:13:ce**

(1101 0100 : 0010 0101 : 1000 1011: 1111 0101 : 0001 0011 : 1100 1110)

Los dispositivos de conexión de nivel 2 switches y hubs. Estos dispositivos se pueden conectar a diferentes ordenadores u otra clase de equipos que dispongan de tarjetas de red, permitiendo la comunicación entre los equipos conectados por diferentes bocas

(generalmente conexiones cableadas de tipo RJ45) bajo ciertos criterios.

Por defecto, un switch permite que las tramas (frames) enviadas a cualquiera de sus bocas se retransmitan a través de todas las demás. Los switches aportan cierta "inteligencia" para evitar que el tráfico de red se dispare. Cuando reciben tramas desde un equipo e identifican que este se encuentra en una boca determinada, toman nota de su dirección física (MAC) y la boca en la que se encuentra. De esa manera, cuando reciban alguna trama dirigida a ese equipo en el futuro, la reenviarán únicamente por la boca correspondiente, evitando retransmitir todas las tramas hacia toda la red.

Los hubs, en cambio, no tienen un comportamiento inteligente. Se limitan a conectar una serie de equipos entre sí, retransmitiendo todas las tramas que les llegan hacia todas las bocas de red disponibles. Todos los equipos conectados a la red escucharán las señales.

.3.2

.3.3

.3.4

.3.5

.3.6

El protocolo IP (Nivel de Red)

El protocolo IP (Internet Protocol) ofrece las siguientes funciones:

- Enrutamiento de paquetes de datos.
- Asignar direcciones a los paquetes de datos.
- Identificar el tipo de contenido y tipo de servicio.
- Fragmentar paquetes demasiado grandes.

Direccionamiento IPv4

Para poder comunicarse en una red, cada equipo debe tener una dirección IP exclusiva. En el direccionamiento IP en clases, existen tres clases de dirección que se utilizan para asignar direcciones IP a los equipos. El tamaño y tipo de la red determinará la clase de dirección IP que aplicaremos cuando proporcionemos direcciones IP a los equipos y otros

hosts de nuestra red.

Componentes de una dirección IP

La dirección IP se expresa mediante un número binario de 32 bits, dividido en 4 octetos, es decir, comprendido entre estos valores:

`00000000.00000000.00000000.00000000`

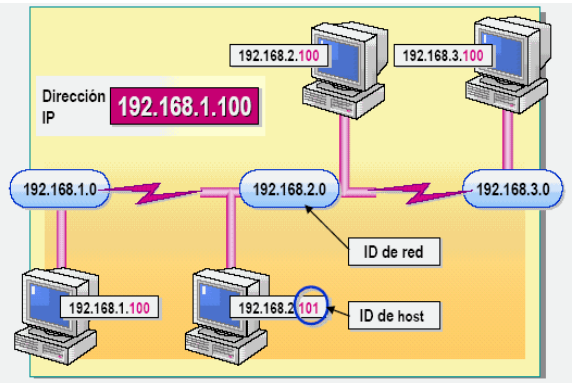
`11111111.11111111.11111111.11111111`

`(en decimal, entre 0.0.0.0 y 255.255.255.255)`

Igual que una dirección postal tiene dos partes (calle y número), una dirección IP está formada por dos partes: **el ID de host y el ID de red.** La primera parte de una dirección IP es el ID de red, que identifica el segmento de red en el que está ubicado el equipo. Todos los equipos del mismo segmento deben tener el mismo ID de red. La segunda parte de una dirección IP es el ID de host.

El número de bits que representa la dirección de red es variable, según veremos a continuación. Por ejemplo, si la dirección de red tiene 16 bits, el formato de una dirección IP sería:

```
RRRRRRRR.RRRRRRRR.HHHHHHHH.HHHHHHHH
R: bits del ID de red
H: bits del ID de host
```



Máscara de red y dirección de red

La máscara de red es la manera diferenciar el prefijo de la dirección IP correspondiente a la ID de red, de la parte correspondiente al identificador de host. Es un número de 32 bits que define un "1" en las posiciones del identificador de red y un "0" en las posiciones del identificador de host.

Al aplicar la operación binaria AND a la IP de un determinado equipo y la máscara de red, se obtiene un resultado (un número binario de 32 bits) que representa la dirección de red en la que se encuentra el equipo. Mediante la aplicación de máscaras de red se puede segmentar una red en diferentes subredes.

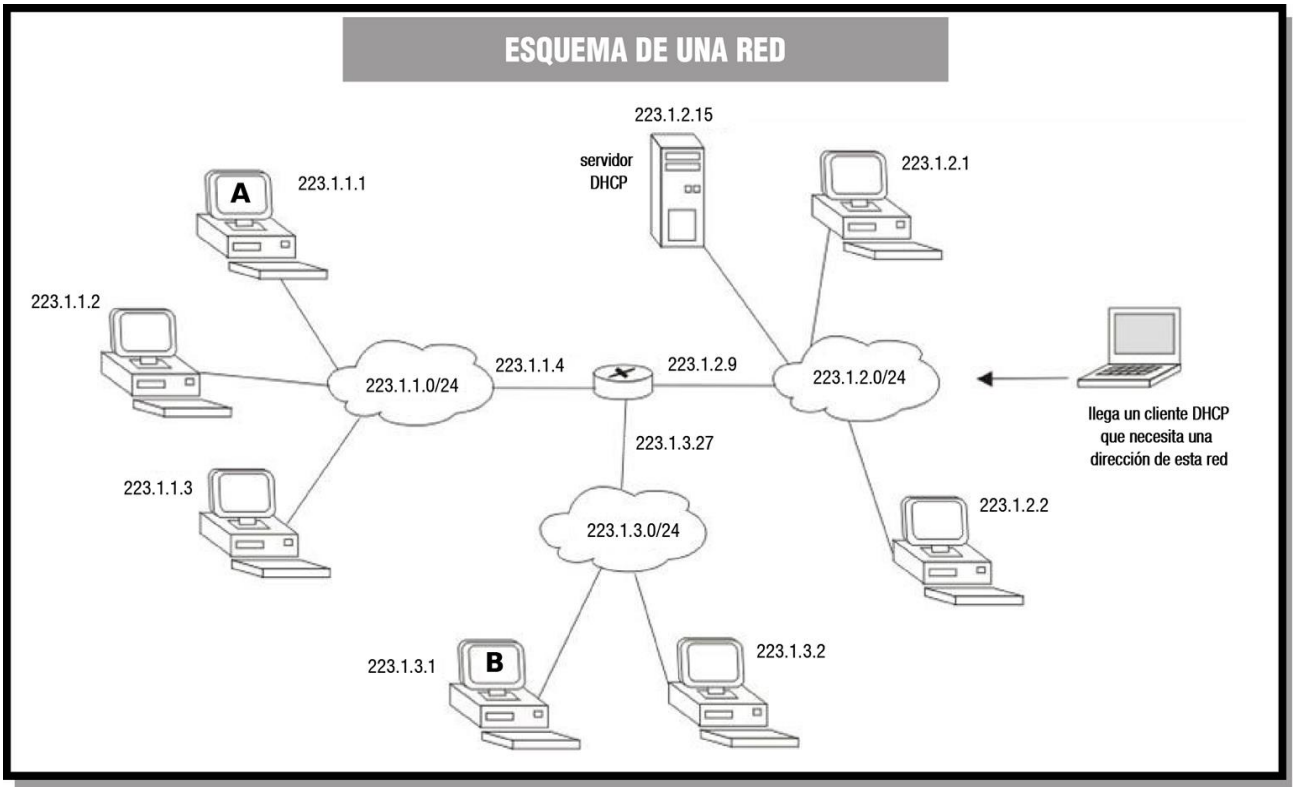
Ejemplo:

Dirección `IP 204.51.170.85` con máscara de red `255.255.255.0` (esto se suele expresar como dirección `204.51.170.85/24`, es decir, **máscara de red de 24 bits.**

```
IP          204.51.170.85          11001100.00110011.10101010.01010101
Máscara     255.255.255.0          11111111.11111111.11111111.00000000

Operación AND, Resultado =>        11001100.00110011.10101010.00000000

Dirección de red =>                204     .51      .170     .0
```



Clases de direcciones IP

El identificador de red puede tener una longitud variable en bits. En el diseño de internet se definieron unos grupos de direccionamiento por defecto, llamados "clases":

- **Clase A:** Direcciones cuyo primer bit es 0, con un byte para el ID de red y 3 bytes para el ID de Host.

`Rango: (0-127).(0-255).(0-255).(1-254)`

`Máscara por defecto: 255.0.0.0`

Son 128 subredes posibles, con 16M de equipos por cada red

- **Clase B:** Direcciones cuyos dos primeros bits son "10", con 2 bytes para el ID de red y 2 bytes para el ID de host.

`Rango: (128-191).(0-255).(0-255).(1-254)`

`Máscara por defecto: 255.255.0.0`

Son 16384 subredes, con 65536 direcciones cada una

- **Clase C:** Direcciones cuyos tres primeros bits son "110", con 3 bytes para el ID de red y 1 byte para el ID de host.

`Rango: (192-224).(0-255).(0-255).(1-254)`

`Máscara por defecto: 255.255.255.0`

Son 2M subredes, con 256 direcciones cada una

Clases de direcciones IP.

CLASE IP	VALOR DE CAMPO	IDENTIFICADOR DE RED	IDENTIFICADOR DE ESTACIÓN
A	0	7 bits	24 bits
B	10	14 bits	16 bits
C	110	21 bits	8 bits
D	1110	28 bits	x
E	11110	27 bits	x

Rango de direcciones para las clases IP.

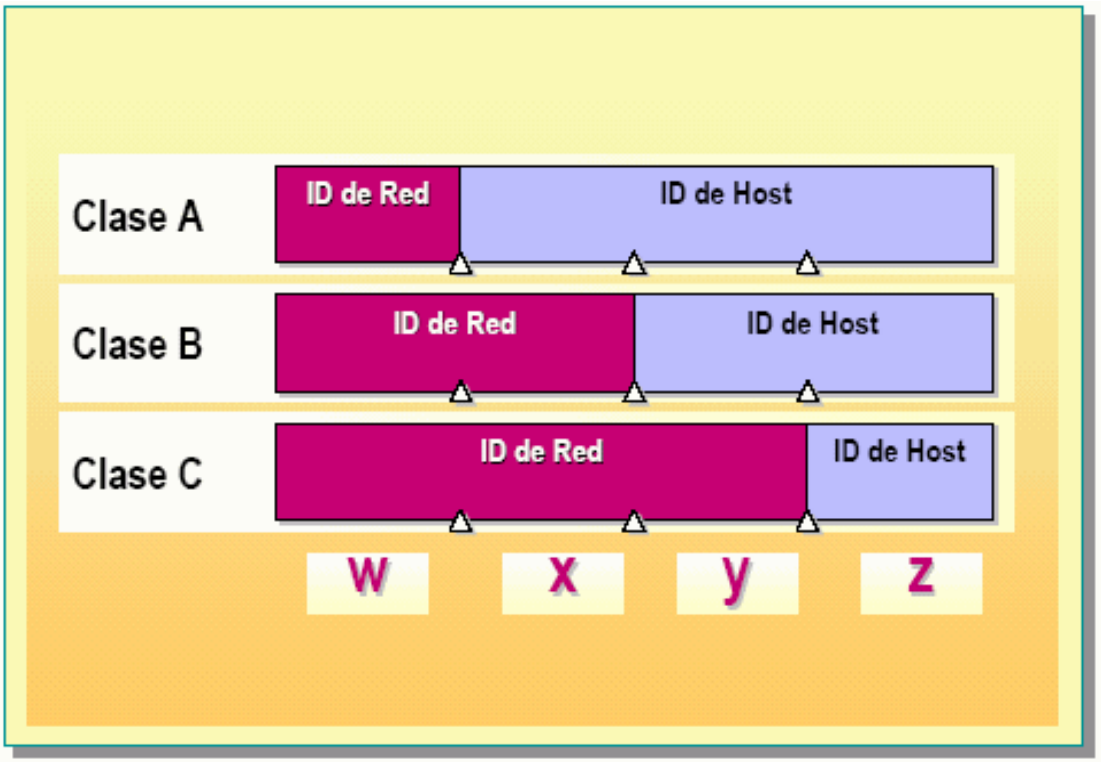
Clase IP	Rango	N° de redes	N° de estaciones
A	1.0.0.0 - 127.255.255.255	127	16777216
B	128.0.0.0 – 191.255.255.255	16384	65536
C	192.0.0.0 – 223.255.255.255	2097152	256
D	224.0.0.0 – 239.255.255.255	X	X
E	240.0.0.0- 247.255.255.255	X	X

Es importante destacar que existen direcciones reservadas, y por tanto no asignables a un ordenador dentro de una red:

- La IP con número de estación todo a ceros (por ejemplo 192.168.1.0): se usa para indicar la **red actual** completa (una red debe tener también una IP por cuestiones de encaminamiento).
- Las IP con todos los bits de número de estación a 1 (por ejemplo 192.168.1.255) se usan para difusión (**broadcast**), es decir, para enviar un mensaje a todas las estaciones dentro de la misma subred, es decir, para todas las direcciones que tienen el mismo identificador de red).
- La dirección del **router**. No tiene por qué ser una dirección concreta, aunque por comodidad se suele elegir la primera dirección de la red. Típicamente los routers toman la primera dirección de la red, por ejemplo, en este caso sería la dirección 192.168.1.1, aunque puede tomar otra.

Vídeo explicativo sobre direccionamiento IPv4:

https://www.youtube.com/watch?v=qEE0s9cnj34



Puerta de Enlace

Recordamos el concepto de **"Puerta de enlace"** o **gateway**.

La puerta de enlace es un dispositivo con el que podemos interconectar redes que tienen protocolos y/o arquitecturas diferentes.

Traduce la información del protocolo utilizado en una red al utilizado en la red de destino, haciendo de nexo entre las direcciones IP de ambas, para que se comuniquen entre sí.

Para esto se utilizan routers o bien ordenadores con, al menos, dos tarjetas de red. Cada puerta de enlace lleva asociada una dirección IP. Entre las más comunes suelen estar: `192.168.1.1, 192.168.0.1, 10.0.0.1,` etc…

La puerta de enlace suele indicar la salida de internet y coincide con un **router**, un **gateway** o un **servidor proxy**.

Subnetting.

Para optimizar el funcionamiento de una red y que esta se ajuste a las necesidades de una compañía se utiliza el concepto de subnetting, o división en subredes más pequeñas a partir

de una dirección de red determinada. Esta división es posible **mediante el uso de máscaras de red ajustadas**. De esa manera, un subconjunto de dispositivos forman una subred, siendo visibles solo entre ellos, mientras otros forman otras subredes diferentes.

Ejemplo:

Supongamos que tenemos la red `172.18.0.0`, es decir, un conjunto de ordenadores con direcciones comprendidas entre la `172.18.0.1` y la `172.18.255.254` y máscara de red de 16 bits `255.255.0.0.` Con esta configuración, tendríamos 2 bytes (16 bits) para direcciones de hosts, lo que nos permitiría direccionar 2^16 máquinas dentro de la misma red. Es decir, descontando la primera y última dirección de la red (dirección de red y dirección de broadcast respectivamente) las máquinas conectadas serían:

`172.18.0.1`
`172.18.0.2`
`172.18.0.3`
`...`
`172.18.255.253`
`172.18.255.254`
`(65534 equipos diferentes)`

Supongamos que por necesidades empresariales sabemos dividir la red global en un número de subredes más eficientes, separando el tráfico de cada una. Podemos plantearnos si el dato de partida es el número máximo de equipos que queremos incluir en cada red o el número de departamentos que queremos tener en la empresa (número de subredes).

Planteamiento 1: número de equipos por red

Nos piden que realicemos un direccionamiento de red teniendo en cuenta el número de equipos que puede albergar cada subred. Por ejemplo, sabemos que el departamento más grande de la compañía tiene 200 ordenadores, pero nos dicen que consideremos un posible crecimiento y le demos a la red una capacidad para albergar hasta 400 equipos.

Partimos de la dirección de red original: `172.18.0.0/16`

- Pasamos 200 a binario: `1100 1000` (requiere 8 bits de host)
- Pasamos 400 a binario: `1 1001 0000` (requiere 9 bits de host)

Es decir, si queremos tener subredes que permitan dimensionar la red actual, con un máximo de 200 equipos por departamento, podríamos utilizar un esquema que dedicara 8 bits a las direcciones de host.

Pero si quisiéramos tener hasta 400 equipos conectados en la misma subred, necesitaríamos disponer de 9 bits para las direcciones de host.

- ¿Qué margen tenemos para modificar la máscara de red?

Si la dirección de los equipos era originalmente:

`RRRRRRRR.RRRRRRRR.HHHHHHHH.HHHHHHHH`

Teníamos una máscara de red con 16 bits:

`11111111.11111111.00000000.00000000`

Lo que haremos será modificar la máscara de red añadiéndole bits hasta ajustar el tamaño necesario para los hosts que nos solicitan, de manera que las direcciones de la red quedarán así:

`RRRRRRRR.RRRRRRRR.RRRRRRRH.HHHHHHHH`

Nueva máscara de red:

`11111111.11111111.1111111`**0**`.00000000`

De esta forma, las direcciones de las subredes que vamos a crear serán todas las variaciones posibles de los bits que hemos añadido a la dirección de red global.

Dirección de red original: `172.18.0.0`

`10101100.00010010``.00000000.00000000`

Aplicamos la NUEVA MÁSCARA DE RED:

`11111111.11111111.11111110.00000000` **`(255.255.254.0) => /23`**

Nuevas subredes:

```
10101100.00010010.0000000 0.00000000 – 172.18.0.0 / 23
10101100.00010010.0000001 0.00000000 – 172.18.2.0 / 23
10101100.00010010.0000010 0.00000000 – 172.18.4.0 / 23
10101100.00010010.0000011 0.00000000 – 172.18.6.0 / 23
10101100.00010010.0000100 0.00000000 – 172.18.8.0 / 23
10101100.00010010.0000101 0.00000000 – 172.18.10.0 / 23
10101100.00010010.0000110 0.00000000 – 172.18.12.0 / 23
10101100.00010010.0000111 0.00000000 – 172.18.14.0 / 23
10101100.00010010.0001000 0.00000000 – 172.18.16.0 / 23
(...)
10101100.00010010.1111101 0.00000000 – 172.18.250.0 / 23
10101100.00010010.1111110 0.00000000 – 172.18.252.0 / 23
10101100.00010010.1111111 0.00000000 – 172.18.254.0 / 23
```

Es decir: *tendríamos 128 subredes que pueden direccionar 510 equipos cada una.*

Los hosts de cada red se obtienen haciendo variar los bits de la parte de host entre 0 y 1:

```
Red 172.18.0.0/23 => 172.18.0.1 – 172.18.1.254 (510 hosts)
Red 172.18.2.0/23 => 172.18.2.1 – 172.18.3.254 (510 hosts)
(...)
```

Planteamiento 2: número de redes necesarias

Nos piden que realicemos un direccionamiento de red que nos permita separar los equipos en 6 departamentos diferentes.

Partimos de la red original: `172.18.0.0/16`

`10101100.00010010.00000000.00000000`

`RRRRRRRR.RRRRRRRR.HHHHHHHH.HHHHHHHH`

- Pasamos 6 (el número de redes que queremos subdividir) a binario: `110`. Es decir, se requieren 3 bits de red adicionales a la **máscara de red** de la nueva estructura. Con 3 bits podemos crear 8 redes (**`000-111`**). Aunque el enunciado solo pide 6 subredes, obtendremos como resultado 8 subredes. Nos quedarán sin usar dos de ellas.

`RRRRRRRR.RRRRRRRR.RRRHHHHH.HHHHHHHH`

Nueva máscara de red: pasaría de /16 a /19

`11111111.11111111.11100000.00000000 => 255.255.224.0 (/19)`

Las nuevas direcciones de red se obtienen con las diferentes combinaciones de bits correspondientes a la nueva máscara:

`10101100.00010010.00000000.00000000 – 172.18.0.0 / 19`

`10101100.00010010.00100000.00000000 – 172.18.32.0 / 19`

`10101100.00010010.01000000.00000000 – 172.18.64.0 / 19`

`10101100.00010010.01100000.00000000 – 172.18.96.0 / 19`

`10101100.00010010.10000000.00000000 – 172.18.128.0 / 19`

`10101100.00010010.10100000.00000000 – 172.18.160.0 / 19`

`10101100.00010010.11000000.00000000 – 172.18.192.0 / 19`

`10101100.00010010.11100000.00000000 – 172.18.224.0 / 19`

Es decir, *tendríamos 8 redes con capacidad para direccionar 2^13 (8192) hosts cada una.*

Los hosts de cada red se obtienen haciendo variar los bits de la parte de host entre 0 y 1:

```
Red 172.18.0.0/19 => 172.18.0.1 – 172.18.63.254 (8190 hosts)
Red 172.18.32.0/19 => 172.18.32.1 – 172.18.63.254 (8190 hosts)
Red 172.18.64.0/19 => 172.18.64.1 – 172.18.95.254 (8190 hosts)
Red 172.18.96.0/19 => 172.18.96.1 – 172.18.127.254 (8190 hosts)
Red 172.18.128.0/19 => 172.18.128.1 – 172.18.159.254 (8190 hosts)
Red 172.18.160.0/19 => 172.18.160.1 – 172.18.191.254 (8190 hosts)
Red 172.18.192.0/19 => 172.18.192.1 – 172.18.223.254 (8190 hosts)
Red 172.18.224.0/19 => 172.18.224.1 – 172.18.255.254 (8190 hosts)
```

CIDR (Supernetting, Agregación de redes)

A la inversa que el subnetting, el CIDR ("Classless Inter-domain routing", también denominado "Resumen de Rutas") consiste en la simplificación de varias direcciones de redes o subredes en una sola dirección IP mediante el uso de una máscara de red menos restrictiva.

Por ejemplo, dadas las siguientes redes, debemos conseguir resumirlas en una sola:

```
172.16.3.0/26, 172.16.3.64/26, 172.16.3.128/26, 172.16.3.192/26
```

Para calcular el resumen de rutas solo se toma en consideración los Bits comunes de todas las direcciones de red, mientras que el resto de bits se ignora.

```
172.16.3.0 /26:              10101100.00010000.00000011.00000000
172.16.3.64 /26:             10101100.00010000.00000011.01000000
172.16.3.128 /26:            10101100.00010000.00000011.10000000
172.16.3.192 /26:            10101100.00010000.00000011.11000000

PATRON:                                  10101100.00010000.00000011.00000000
MASCARA (24 bits comunes):               11111111.11111111.11111111.00000000
Dirección de red global: 172.16.3.0 / 24, Mascara: 255.255.255.0
```

VLSM (Máscaras de subred de longitud variable)

Al aplicar la técnica estándar de subnetting, es posible que el resultado no se ajuste exactamente a nuestras necesidades. Nos pueden sobrar redes que quedarán vacías. También es posible que necesitemos un número de hosts que no se reparta equitativamente entre las diferentes redes, por lo que dividir una red en subredes de la misma capacidad podría ser ineficiente. Para mejorar la división de subredes y conseguir un mejor aprovechamiento se utiliza el direccionamiento con máscaras de longitud variable. El resultado será un número de redes determinado con diferentes máscaras de red para cada una de ellas.

Ejemplo de VLSM:

Nuestra empresa dispone de la siguiente dirección de red global: 10.5.126.0 /23

Necesitamos crear cuatro departamentales con las siguientes características:

- RED A : 130 equipos (+1 de red + 1 de broadcast)

- RED B: 70 equipos (+1 de red + 1 de broadcast)
- RED C: 40 equipos (+1 de red + 1 de broadcast)
- RED D: 10 equipos (+1 de red + 1 de broadcast)

Partimos de: `10.5.126.0/23`, es decir:

```
Red         00001010.00000101.01111110.00000000 (10.5.126.0)
Máscara     11111111.11111111.11111110.00000000 (255.255.254.0)
```

La red principal utiliza 23 bits, y por tanto , nos quedan 9 Bits para direcciones de hosts.

Solución VLSM:

La **red A** requiere 132 hosts. Necesitaremos un prefijo /24 , porque cogeremos 8 bits para hosts (2^8=256). Con 7 bits solo llegaríamos a 128 equipos.

Asi pues, esta red quedaría: **`10.5.126.0/24`**

```
Red A       00001010.00000101.01111110.00000000 (10.5.126.0)
Máscara     11111111.11111111.11111111.00000000 (255.255.255.0)
Broadcast:00001010.00000101.01111110.11111111 => 10.5.126.255
```

Y los equipos serán: `10.5.126.1 – 10.5.126.254 => 254 hosts`

La **red B** requiere 72 direcciones. Necesitaremos 7 bits para direccionarlos (2^7=128)

Partímos de la última dirección de la red usada anteriormente (sabemos que el broadcast de la red anterior es `10.5.126.255`) y obtendremos:

La dirección de la RED B será **`10.5.127.0/25`**

```
Red B       00001010.00000101.01111111.00000000 (10.5.127.0)
Máscara     11111111.11111111.11111111.10000000 (255.255.255.128)
Broadcast:00001010.00000101.01111111.01111111 => 10.5.127.127
```

Equipos de la red B: `(10.5.127.1 – 10.5.127.126) => 126 hosts.`

La **red C** requiere 42 direcciones. Necesitaremos 6 bits (2^6=64)

Partimos de la última dirección usada anteriormente (`10.5.127.127`)

La siguiente red será la **`10.5.127.128/26`**

Sabemos que necesitamos 6 bits de hosts, por lo que la máscara será de 26 bits.

```
Red C      00001010.00000101.01111111.10000000 (10.5.127.128)
Máscara    11111111.11111111.11111111.11000000 (255.255.255.192)
Broadcast:00001010.00000101.01111111.10111111 => 10.5.127.191
```

Equipos en la red C: `10.5.127.129 – 10.5.127.190 => 62 hosts.`

La **red D** requiere 12 direcciones (2^4=16, por tanto serán 4 bits de hosts => /28)

Será la red **`10.5.127.192/28`** con broadcast 10.5.127.207

```
Red D      00001010.00000101.01111111.11000000 (10.5.127.192)
Máscara    11111111.11111111.11111111.11110000 (255.255.255.240)
Broadcast:00001010.00000101.01111111.11001111 => 10.5.127.207
```

La red D comprenderá los equipos `10.5.127.193 – 10.5.127.206 (14 hosts)`

Encaminamiento (routers)

El proceso por el que se envía información desde una máquina origen a una máquina destino, independientemente de que ambas se encuentren en la misma red o en redes diferentes, se conoce como encaminamiento a nivel IP. El protocolo IP es el responsable del encaminamiento.

Los encaminadores o routers son los dispositivos de red (nivel 3) que permiten enlazar diferentes redes para transferir información entre ellas. Los equipos también realizan encaminamiento, siendo capaces de determinar si el destino de la información se encuentra en la misma red que el equipo de origen o es necesario enviarlo a otra. En caso de que el destino esté en la misma red, el equipo puede enviar la información directamente, sin intermediación del router. Si el destino está en una red desconocida, el equipo enviará la información a su encaminador, puerta de enlace o router por defecto, para que este envíe la información al destino correcto (envío indirecto).

Los equipos, así como los routers, disponen de "tablas de encaminamiento" donde se almacena la información necesaria para enviar la información a su destino. Al arrancar los equipos, las tablas de encaminamiento se inicializan con información de las rutas a las redes adyacentes. La información de enrutamiento se puede añadir mediante

encaminamiento estático (manual, es decir, mediante ficheros de configuración), o encaminamiento dinámico (el propio router actualiza sus tablas mediante protocolos como RIP, OSPF).

Vídeo ejemplo sobre el uso de la herramienta **Packet Tracer** para simular redes y enrutamiento entre ellas:

https://www.youtube.com/watch?v=nzXdm6GJgmo

El nivel de transporte en TCP/IP. Protocolos TCP y UDP

El nivel de transporte proporciona los elementos que permiten diferenciar y gestionar simultáneamente varios orígenes y destinos en una misma comunicación, y múltiples comunicaciones en cada equipo. Para ello se utiliza el concepto de "puerto de comunicaciones". Cada proceso a nivel de aplicación tiene asociado uno o varios puertos, a través de los cuales está accesible. Cada puerto se identifica por un número binario de 16 bits (0-65535).

La identificación de puertos sigue un convenio según el uso que se hace de ellos:

- **Puertos conocidos (0-1023)**: Reservados para aplicaciones y servicios estandarizados, como HTTP, FTP, etc... Las aplicaciones cliente se conectan a estos puertos para acceder a los servicios.
- **Puertos registrados (1024-49151)**: aplicaciones no estándar, instaladas por el usuario. Se pueden asignar dinámicamente a clientes si no hay ningún servicio que los esté utilizando.
- **Puertos dinámicos (49152-65535)**: empleados para iniciar conexiones desde el cliente.

El protocolo **UDP** (User Datagram Protocol) proporciona un servicio "no orientado a conexión", es decir, no se realiza un establecimiento de conexión previo a la conexión, ni tiene control de flujo. Puede ocurrir con este protocolo que se entreguen segmentos duplicados o desordenados de la información.

El protocolo **TCP** (Transmission Control Protocol) proporciona un servicio "orientado a conexión", es decir, obliga al establecimiento de una conexión antes de empezar a transmitir. Ofrece control de flujo y de errores.

Traducción de direcciones de red: NAT

El crecimiento exponencial del número de ordenadores conectados a internet obligó a diseñar mecanismos y técnicas para aprovechar de una manera eficiente el espacio de direcciones disponibles. Cada ordenador conectado a internet no puede ser visible desde el resto de la red con una dirección única, porque no habría suficientes direcciones IPv4. En su lugar, las organizaciones disponen de algunas direcciones públicas a través de las que salen a internet, y disponen de NAT (Network Address Translation), un servicio de traducción de direcciones que permite a cada uno de los sistemas de la red privada acceder a internet con una dirección pública, en lugar de la red que utiliza internamente para comunicarse con el resto de sistemas de la organización. Los encaminadores o routers conectados a internet realizan esta operación de manera transparente, incluso reescribiendo los datagramas que ha de encaminar hacia internet.

.4 Virtualización

Para la realización de ejercicios y prácticas de Servicios en Red será necesario utilizar máquinas virtuales.

Una **máquina virtual** es un programa que emula un ordenador, es decir, utiliza una serie de recursos del ordenador principal (anfitrión) para tener un sistema "invitado" dentro del primero. Los sistemas invitados aparentan tener una BIOS y unos componentes de hardware determinados, como memoria, discos, tarjetas de red, etc... todos ellos simulados a partir de subconjuntos de recursos del anfitrión.

Al tratarse de una capa intermedia entre el sistema físico y el sistema operativo cargado sobre el hardware emulado, la velocidad de ejecución del último es menor, pero suficiente para usarse incluso en entornos profesionales de producción.

Existen diferentes soluciones de virtualización de sistemas, siendo las más comunes VMWare y Oracle VirtualBox. VMWare permite ejecutar máquinas virtuales con Windows, Linux, NetWare o Solaris x86. VirtualBox puede ejecutarse sobre Linux, Mac OS, OS/2, Windows y Solaris, y permite virtualizar sistemas FreeBSD, GNU/Linux, OpenBSD, OS/2, Windows y Solaris. Esta herramienta fue desarrollada por Innotek, compañía adquirida por Sun Microsystems en 2008.

En términos técnicos, "Virtualización" es la creación de una versión virtual (lógica) de algún

recurso tecnológico, como puede ser una plataforma de hardware, un sistema operativo, un dispositivo de almacenamiento u otros recursos de red.

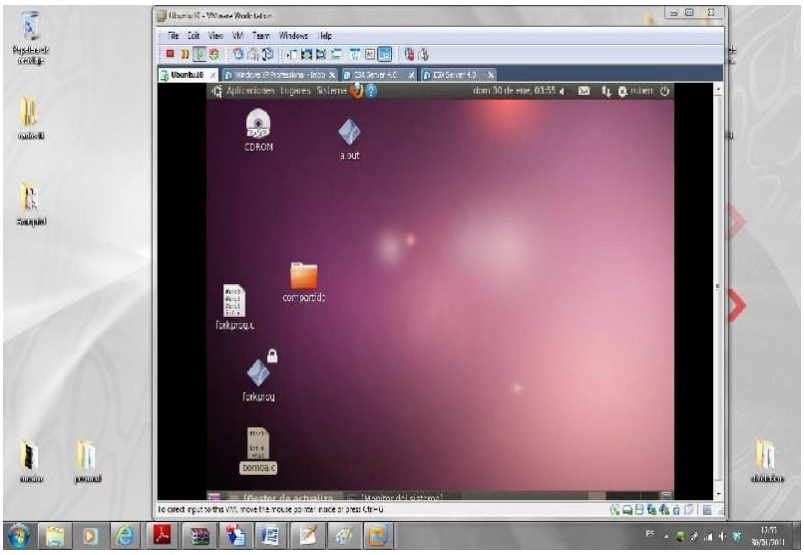
Cada una de estas máquinas virtuales generadas por el sistema operativo ofrece a las aplicaciones una serie de recursos virtuales (espacio de almacenamiento, impresora, video, etc.) de modo que un error en la aplicación no afecte al hardware real del sistema informático, sino a este hardware virtual.

Conceptos. Anfitrión y huésped

Los dos conceptos más importantes para entender qué es la virtualización son los de anfitrión e invitado. Ambos conceptos se refieren a nuestros sistemas operativos, y por lo tanto deberíamos hablar de sistema operativo anfitrión y sistema operativo invitado.

El **anfitrión** (o "host") es el sistema operativo del ordenador en el cual instalamos nuestro programa de virtualización y que asignará o prestará determinados recursos de hardware a la máquina virtual que creemos.

El **invitado** ("guest" o "huésped") es el sistema operativo que instalamos en el ordenador virtual que hemos creado, mediante nuestro programa de virtualización y al cual hemos asignado determinados recursos para funcionar.



En este ejemplo podemos ver como sobre un Windows 7 (SO anfitrión) se ha instalado una máquina virtual sobre la que está corriendo un Linux Ubuntu (SO invitado).

Es decir, el anfitrión alberga al invitado. Un anfitrión puede tener varios invitados, no está limitado solo a uno.

Requisitos hardware

Para construir la máquina virtual tenemos que asignar determinados recursos de hardware, como espacio en disco duro, memoria RAM, número de procesadores, etc. que el anfitrión cederá o compartirá con el invitado.

Cuando tengamos nuestra máquina virtual, será necesario instalar en ella un sistema operativo, que funcionará con las mismas reglas que lo hace en un ordenador normal, actualizaciones, licencias, instalación de software adicional, etc.

Imaginemos un ordenador en el que tenemos instalado un Windows XP. Si en dicho ordenador instalamos un software de virtualización y creamos una máquina virtual que ejecute Windows 7 por ejemplo, dicho ordenador estará realmente ejecutando dos sistemas operativos al mismo tiempo (el host y el guest), y todos sus recursos se estarán repartiendo entre ambos. Si el anfitrión tiene recursos limitados de memoria y CPU es probable que las máquinas virtuales no funcionen con buen rendimiento. Es conveniente como mínimo contar con 2 GB de RAM, suficiente espacio en disco duro y, lo más importante, un microprocesador potente que pueda dividir su tiempo de proceso entre los dos SO.

Tipos de máquinas virtuales

Máquinas virtuales de proceso

Una máquina virtual de proceso, a veces llamada "máquina virtual de aplicación", se ejecuta como un proceso normal dentro de un sistema operativo y soporta un solo proceso. La máquina se inicia automáticamente cuando se lanza el proceso que se desea ejecutar y se detiene para cuando éste finaliza. Su objetivo es el de proporcionar un entorno de ejecución independiente de la plataforma de hardware y del sistema operativo, que oculte los detalles de la plataforma subyacente y permita que un programa se ejecute siempre de la misma forma sobre cualquier plataforma.

El ejemplo más conocido sería la **máquina virtual de Java**.

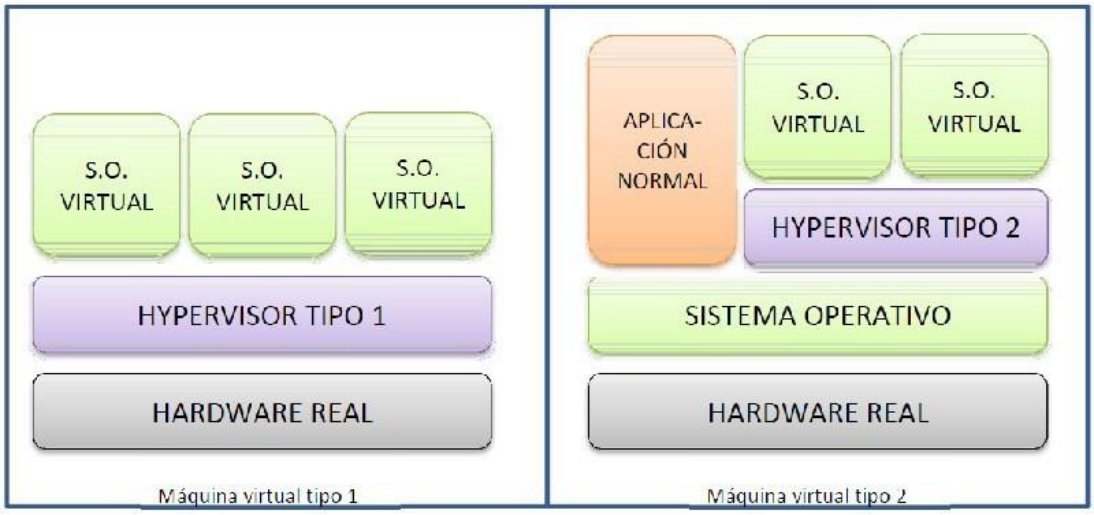
Máquinas virtuales de sistema (o de hardware)

Permiten a la máquina física dividirse entre varias máquinas virtuales, cada una ejecutando su propio sistema operativo.

A la capa de software que permite la virtualización se la llama monitor de máquina virtual o "hypervisor" (no es un sistema operativo completo). Hay dos tipos de hypervisores:

· **De tipo 1**. El hypervisor o monitor se ejecuta directamente sobre nuestro hardware y nos permite crear máquinas virtuales, por lo tanto desaparece la necesidad de contar con un sistema operativo anfitrión, solo tendremos sistemas huéspedes, y el anfitrión será directamente nuestro monitor o hypervisor.

· **De tipo 2**. Es el que hemos visto anteriormente, en el cual existe un sistema operativo sobre el hardware del sistema, sobre el que funciona un monitor o hypervisor que crea los sistemas operativos invitados.



Otros modelos de Virtualización

- **Virtualización de almacenamiento:** Los arrays y pools de discos representan estructuras de datos virtualizadas. Por ejemplo: RAIDs, Volúmenes LVM en Linux, Sistemas de ficheros ZFS en Solaris, sistemas de ficheros distribuidos (GFS, DFS...) o SAN (Storage Area Network).
- **Virtualización de redes:** técnica empleada para simular switches e interfaces de red virtuales: Open vSwitch, Crossbow, VMWare Nsx, etc...
- Memoria virtual
- **Unión de interfaces de red** (Ethernet Bonding)
- Balanceo de carga
- **Virtualización de aplicaciones:** con el desarrollo de sistemas cada vez más potentes se están usando de una manera cada vez más extendida los servicios de virtualización de aplicaciones, lo que facilita herramientas para el cloud computing (Dockers, Citrix, etc...)

Ventajas y desventajas de la virtualización

La virtualización de sistemas tiene estas ventajas:

- Aislamiento de aplicaciones y usuarios
- Consolidación de servicios / servidores
- Simplificación de la administración

- Compatibilidad y portabilidad (SO, software, hardware)
- Flexibilidad y escalabilidad
- Mejor gestión de los recursos físicos => Ahorro de costes
- Ejecución de software heredado (software antiguo que todavía satisface alguna necesidad, vinculado a una versión específica de sistema operativo o hardware que ya no está activo)
- Evaluación y entornos de pruebas/dessarrollo
- Alta disponibilidad y recuperación
- Favorece el Cloud Computing
- Posibilidad de mover la plataforma a otro entorno/lugar de manera simple.

Por contra, la virtualización también presenta algunas desventajas:

- Aislamiento real
- Los fallos de hardware pueden tener consecuencias más graves
- Formación requerida
- Menor rendimiento
- Es difícil virtualizar determinadas aplicaciones, servicios, recursos, etc...
- Licencias
- Copias de seguridad

Software de virtualización

VirtualBox

Virtual Box es la solución gratuita para virtualizar sistemas operativos. Para su manejo nos remitimos a los manuales:

- Manual de VirtualBox (Oracle) =>

https://www.virtualbox.org/manual/

- Manual práctico (Universidad de Barcelona) : http://bd.ub.edu/preservadigital/sites/bd.ub.edu.preservadigital/files/Tutoriales_VirtualBox.pdf

Vmware

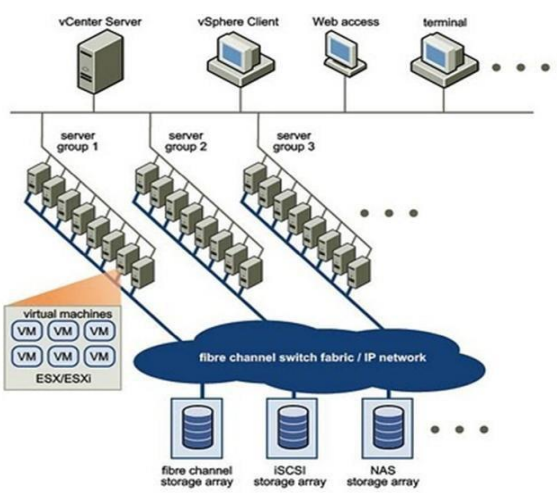
VMware es, hoy en día, la plataforma líder en sistemas virtualizados. Es una solución de alto rendimiento y con grandes posibilidades de ampliación. Al contrario que VirtualBox, que es un único programa, VMware presenta varias soluciones para la virtualización:

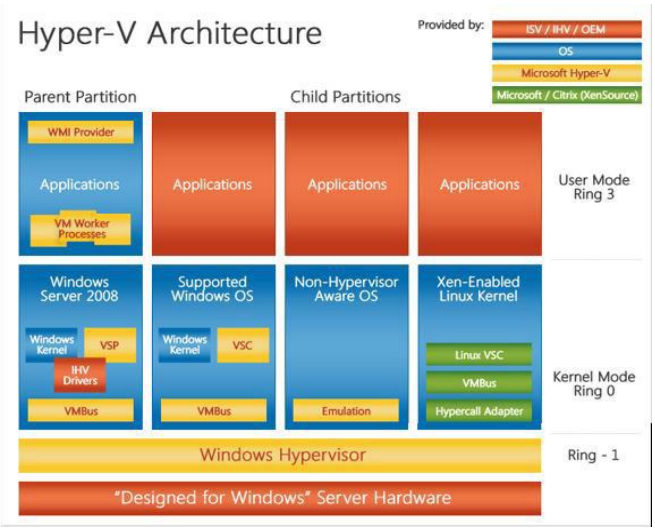
• VMware Converter: permite virtualizar nuestro propio equipo o hacerlo con cualquier otro de nuestra red. Se usa para migrar a sistemas virtuales sin perder ninguna funcionalidad. Permite elegir particiones a virtualizar (por ejemplo, la partición del sistema pero no la de datos)

• VMware player: es un hypervisor de tipo 2 de virtualización completa o nativa, similar a VirtualBox. Nos permite crear y ejecutar máquinas virtuales. Es ligero y tiene un buen rendimiento. Reconoce los dispositivos USB y permite compartir carpetas fácilmente.

• VMware View: Parecido al VMware player pero sin la posibilidad de crear máquinas virtuales. Esta especialmente indicado para ser usado en máquinas con pocos recursos que se encargarán de presentar una máquina virtual, normalmente alojada en un servidor de la empresa.

Hyper-V





Hyper-V es un virtualizador de sistemas basado en Windows Server, que trabaja mediante

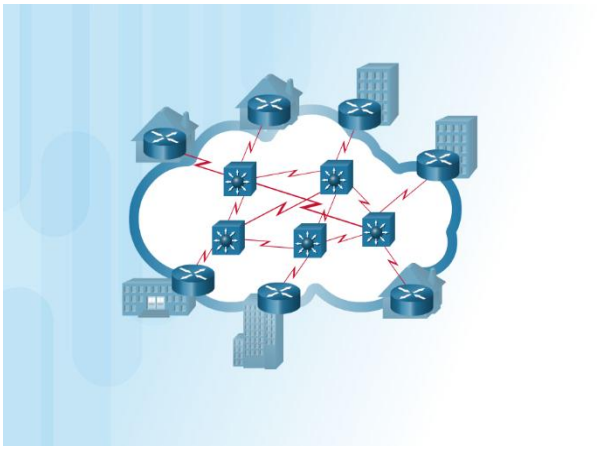
lo que denomina particiones. Una partición es un contenedor lógico, creada por el hypervisor, en el que se ejecuta un sistema operativo virtualizado. Hyper-V también permite la virtualización de sistemas operativos Linux en sus últimas versiones.

Existe una partición raíz en la que se ejecuta Windows Server, que tiene acceso directo al hardware y permite crear particiones hijas donde corren los otros sistemas operativos virtualizados.

Existen otras soluciones para virtualizar sistemas, como Virtual PC, KDM (Linux)...

.5 Servicios de interconexión a WAN

Una red de área amplia, o WAN, (Wide Area Network en inglés), es una red de ordenadores que une varias redes locales, aunque sus miembros no estén todos en una misma ubicación física. Muchas WAN son construidas por organizaciones o empresas para su uso privado, otras son instaladas por los proveedores de internet (ISP) para proveer conexión a sus clientes.



Hoy en día, internet brinda conexiones de alta velocidad, de manera que un alto porcentaje de las redes WAN se basan en ese medio, reduciendo la necesidad de redes privadas WAN. Las redes privadas virtuales utilizan cifrado y otras técnicas para generar una red dedicada sobre comunicaciones en internet, aumentan continuamente.

Tipos de acceso

Hay varias opciones disponibles para la conectividad WAN:

Opción:	Descripción	Ventajas	Desventajas	Ancho de banda	Ejemplos de protocolos
Línea dedicada	Peer-to-peer de conexión entre dos ordenadores, o redes de área local (LAN)	El más seguro	Caro		PPP , HDLC , SDLC , HNAS

Opción:	Descripción	Ventajas	Desventajas	Ancho de banda	Ejemplos de protocolos
Conmutación de circuitos	Se crea un camino o circuito dedicado entre los puntos finales. Su mejor ejemplo son las conexiones de acceso telefónico.	El más barato	Configuración de llamadas	28-144 kbit/s	PPP , RDSI
Conmutación de paquetes (Orientado por conexión)	Se establece un circuito virtual a través del cual se envían paquetes de longitud variable.	Más eficaz que la conmutación de circuitos	Necesidad de mayor lógica de control		X.25 , Frame-Relay
Conmutación de paquetes (sin conexión)	Los paquetes se transportan punto a punto, a través de una red interna, sin establecer un circuito virtual permanente. La red tiene la responsabilidad de entregar el paquete, pero no se garantiza el destino final. Internet funciona de esta manera.	Muy robusto y bajo costo operativo	No garantiza la entrega.		IPv4 , IPv6
Conmutación de celdas	Es igual que en la conmutación de paquetes, pero utiliza células de longitud fija en lugar de paquetes de longitud variable. Los datos se dividen en celdas de longitud fija, y luego son transportados a través de circuitos virtuales.	Antes del 2000, esta fue vista como la mejor opción para el uso simultáneo de voz y datos. Con las altas velocidades de enlace en las redes modernas, esta	El "overhead" o exceso de información de control puede ser considerable		ATM (Asynchronous Transfer Mode)

Opción:	Descripción	Ventajas	Desventajas	Ancho de banda	Ejemplos de protocolos
		ventaja carece de sentido.			

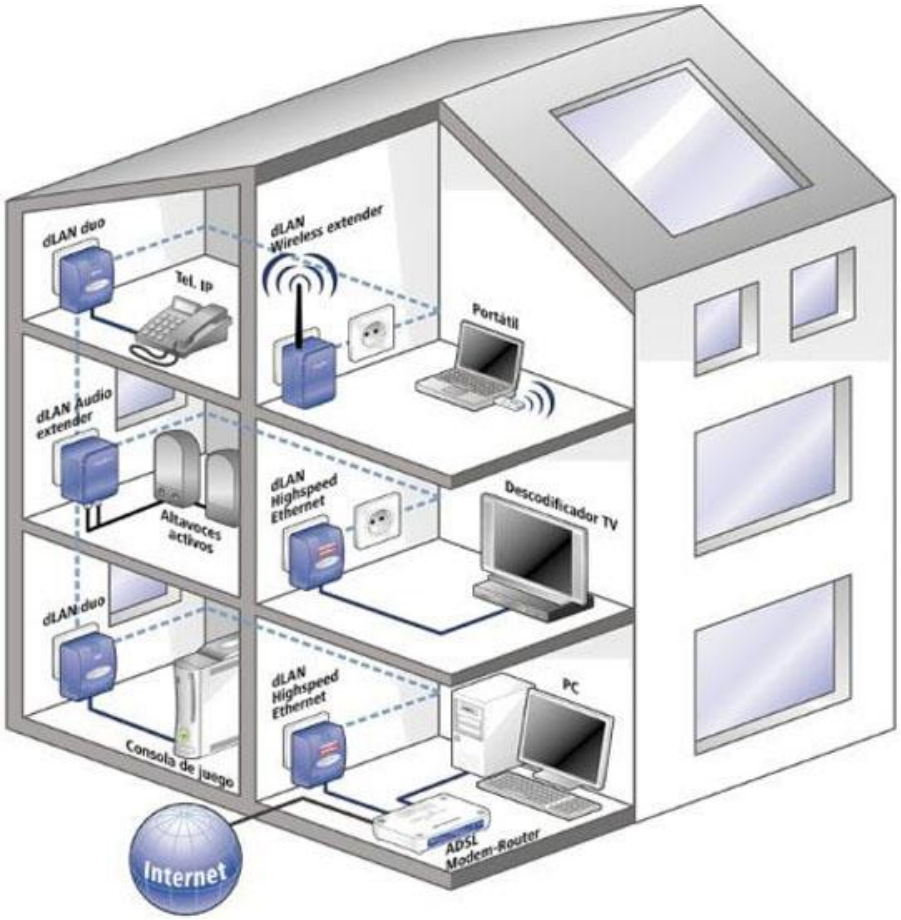
Las tasas de transmisión han aumentado con el tiempo y seguirán aumentando. Alrededor de 1960 se comunicaba a una tasa de 110 bits por segundo en el borde de la WAN y una tasa de 56 kbit/s a 64 kbit/s se consideraba "rápida".

En 2016 los hogares estaban conectados a Internet con ADSL o Fibra óptica a velocidades entre 1 Mbit/s y 300 Mbit/s. Las conexiones en el núcleo de una WAN puede variar de 1 Gbit/s a 300 Gbit/s.

Con la proliferación del bajo coste de conexión a Internet, muchas empresas y organizaciones recurren a las VPN para interconectar sus redes, creando una red WAN de esa manera. Empresas como Citrix, Cisco, New Edge Networks y Check Point ofrecen soluciones para crear redes VPN.

En este artículo podemos ver un buen resumen de las tecnologías existentes e históricas para la conexión a internet:

https://tecnologia-informatica.com/tipos-conexion-internet/


dLAN duo
Tel. IP
dLAN Wireless extender
Portátil
dLAN Audio extender
Altavoces activos
dLAN Highspeed Ethernet
Descodificador TV
dLAN duo
Consola de juego
dLAN Highspeed Ethernet
PC
Internet
ADSL Modem-Router