Unidad 4

Fundamentos de Redes (II)

Planificación y Administración de Redes

4

Índice

Introducción

Como vimos en unidades anteriores el modelo OSI se dividía en 7 capas, de las cuales vimos en la unidad anterior el Nivel físico y el nivel de enlace.

En esta unidad se verán los siguientes niveles: el de red, el de transporte y el de aplicación.

Los niveles correspondientes a sesión y presentación no se estudiarán ampliamente debido a que no tienen una importancia mayor de la que se explica en las primeras unidades.

En cuanto al nivel de red nos referimos, veremos sus principales características y cómo funcionan sus diversas funciones.

Tendremos que hacer hincapié en los direccionamientos del protocolo IP, el IPv4 y el IPv6 ya que son las bases para la configuración de las redes de hoy en día y cómo podemos comunicarnos entre distintos equipos.

Además de todo esto, en el nivel de red también existen los protocolos ARP e ICMP que nos ayudarán a trabajar con las redes y a diagnosticar posibles fallos.

En cuanto al nivel de transporte, veremos que existen dos protocolos principales asociados a este dependiendo de si se establece o no una conexión, que serán TCP y UDP. También veremos cómo funcionan los sockets y los puertos de comunicaciones para entablar una conversación.

Por último, se trabajará con el nivel de aplicación, sabiendo que este se basa en distribuir los servicios de red y cómo funciona de manera muy general cada servicio, pues los más importantes de estos se verán con detenimiento el próximo curso en el módulo 'Servicios de red e Internet'.

Todo esto se trabajará en base a los distintos sistemas y veremos como podremos configurar en su mayoría esto en cualquier equipo independientemente del sistema operativo que use.

Al finalizar esta unidad

- Comprenderemos cómo funciona el nivel de red, sus funciones y sus principales protocolos.
- Sabremos como trabaja el protocolo IP, tanto en IPv4 como en IPv6.
- Seremos capaces de entender las direcciones IPv4, sus tipos y como clasificarlas.
- Realizaremos la división de redes en subredes para adecuarnos a la necesidad de cada red.
- Podremos configurar los adaptadores de red de los distintos sistemas operativos.
- Conoceremos cómo funciona el protocolo ARP.
- Seremos capaces de diagnosticar el estado de una red con el protocolo ICMP.
- Sabremos cómo funciona IPv6 y como pasar de las direcciones IPv4 a las nuevas direcciones IPv6.
- Comprenderemos cómo funciona el nivel de transporte y sus funciones.
- Identificaremos que es un socket y un puerto y cómo funcionan.
- Podremos identificar cómo funciona el protocolo TCP.
- Podremos identificar cómo funciona el protocolo UDP.
- Sabremos en que se basa el nivel de aplicación.
- Habremos introducido los principales servicios de red y sus características más fundamentales.

4.1. Nivel de red

4.1.1. Objetivo

Como se trató en la unidad 1 cuando hablábamos de la arquitectura de redes, el nivel de red es el tercer nivel estipulado en el modelo OSI y es el encargado de que los paquetes se intercambien entre los distintos equipos.

4.1.2. Funciones del nivel de red

Direccionamiento IP

Para que se pueda establecer una comunicación entre distintos hosts es necesario que estén identificados, y esta identificación en el nivel de red se hace por direcciones IP. La asignación de las distintas direcciones únicas para los equipos es el llamado **Direccionamiento IP**.

Enrutamiento de paquetes

Cuando se transmite la información y sabe efectivamente donde tiene que llegar, el problema que se presenta es que no hay un único camino posible, Es probable que haya varios caminos a cada cual más complicado o liviano para la información.

El enrutamiento precisamente establece la que es la mejor ruta para la información en base a las características de la red, como el retardo o la distancia. Estas rutas son variables y cambiantes, por lo que puede que los paquetes lleguen hasta desordenados, pero es trabajo del receptor volver a ordenarlos.

Encapsulación de segmentos/desencapsulación de tramas

Como se vio anteriormente, el emisor crea el mensaje que se va a enviar en el nivel de aplicación. Posteriormente el nivel de transporte lo divide en segmentos de un tamaño concreto y lo pasa al nivel de red.

Una vez que llega al nivel de red este lo encapsula en un paquete añadiendo la cabecera donde se encuentran las direcciones IP del origen y del destino. Para llegar al destino, la información encapsulada irá pasando por varios routers que le indicarán la ruta a seguir hasta el siguiente y cuando llega al host de destino su nivel de red se encarga de desencapsular la trama.

Control de congestión

Si nos encontramos con que un router tiene de forma repentina una carga de trabajo superior a la que puede procesar se produce una congestión. Este problema, extrapolable a la red porque en cuanto un router quede parado la red se colapsa, se soluciona con aciertas técnicas de prevención que se estudiarán en otro momento.

4.1.3. Protocolos del nivel de red

El nivel de red contempla los siguientes protocolos:

- Protocolos de direccionamiento y encapsulación: **IP** (IPv4 e IPv6)
- Protocolo de resolución de direcciones: **ARP y RARP**.
- Protocolo de diagnóstico de red: **ICMP** (ICMPv4 e ICMPv6).
- Protocolo de enrutamiento: **RIP, OSPF, IS-IS, BGP y EIGRP**.
- Protocolo de seguridad: **IPSec e IGMP** (envío en multicast).
- Protocolo de control de congestión: **ECN**.

4.2. Protocolo IP

4.2.1. Características

Las principales características del protocolo IP son:

- **No está orientado a la conexión.** Cada paquete puede tomar distintas rutas.
- **No es fiable.** Puede que en el camino los paquetes lleguen desordenados, pero también puede ser que no lleguen o lleguen dañados.

El funcionamiento del protocolo IP es el siguiente: un paquete IP es enviado en el campo de datos de la trama en cada una de las redes que atraviese hasta llegar al destino, y cada vez que sufra un cambio de red el datagrama o paquete sufrirá cambios de encapsulación, volviéndose a encapsular en cada nueva red que entre.

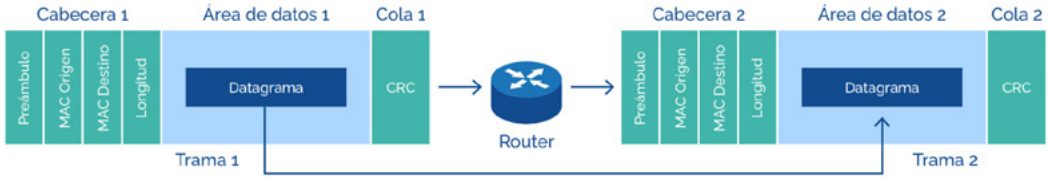

Imagen 1. Encapsulación/Desencapsulación de datagramas

4.2.2. Formato del datagrama IPv4

Un datagrama en IPv4 consta de dos partes fundamentales:

- La cabecera que ocupa de 20 a 60 Bytes.
- El área de datos que ocupa de 0 a 65535 Bytes.

Imagen 2. Datagrama IPv4

Las cabeceras tienen una extensión que puede ir variando ya que el número de opciones que comprende siempre será indeterminado, lo que hace que el router tarde más en decidir cuál sería la ruta correcta.

4.2.3. Formato del datagrama IPv6

Al contrario que lo que pasaba con el datagrama de IPv4, el datagrama IPv6 consta de tres partes fundamentales:

- La cabecera, con una longitud fija de 40 Bytes.
- La cabecera de extensión que es opcional
- El área de datos con una longitud variable.

Carga útil, de 0 a 65535 Bytes

Imagen 3. Datagrama IPv4

Podemos encontrar lo siguiente en el datagrama de IPv6:

- **Versión:** se define la versión del protocolo, que es IPv6, por lo que su valor es 6.
- **Clase de tráfico:** aquí se expresa cual es la prioridad del datagrama en el tráfico de la red. Los valores del 0 al 7 se usan para el control de congestión y los valores del 8 al 15 cuando no hay control de congestión.
- **Etiqueta de flujo:** se usa para el tratamiento de la calidad del servicio (Quality of Service, QoS).
- **Cabecera siguiente:** indicaremos de que tipo de cabecera se trata la que sigue a la fija de IPv6, es decir, la cabecera de extensión, y si no existe, la cabecera de transporte en TCP o UDP. Existen 4 tipos de cabeceras de extensión:
  - Hop-By-Hop.
  - Routing
  - Fragment
  - Authentication-Header
  - ESP
  - Destination Options
  - No Next Header
- **Límite de saltos:** comprendemos aquí el número de saltos que puede dar el datagrama antes de que sea desechado, es decir, el tiempo de vida o TTL, que es 128 por lo general.
- **Dirección IP de origen o de destino:** es la dirección IP en formato IPv6 con 8 grupos de 4 cifras Hexadecimales que se encuentran separados por dos puntos.

Como la cabecera siempre tiene una extensión de 40 bytes, el router no necesita leer nada más para saber cuál es la ruta que debe de seguir.

4.3. Direccionamiento IPv4

4.3.1. Direcciones IPv4

Las direcciones lógicas o direcciones en IPv4 son una manera de identificación para los hosts única. Con esto se consigue que se pueda producir la conexión de un equipo a la red.

Las direcciones IPv4 se encuentran formadas por 4 Bytes o 32 bits separados en cuatro grupos (1 Byte u 8 bits por grupo) escribidos en notación decimal y separados por un punto. Cada byte puede tomar hasta 28 valores, empezando por el 0, es decir, del 0 al 255. Un ejemplo sería: 10.0.0.24.

Hay dos tipos de distinciones de direcciones IPv4:

- **Según su uso:**
  - **Públicas:** se pueden visualizar en internet.
  - **Privadas:** solo son visibles en la red interna a la que pertenezcan.
- **Según su asignación:**
  - **Estáticas:** se asignan de manera manual, tarea que corresponde al administrador de la red.
  - **Dinámicas:** las asigna de manera automática un servidor DHCP.

4.3.2. Clases

Los 4 Bytes que forman la dirección IPv4 determinan los siguientes tres datos principales:

- La clase.
- El identificador de red o netid.
- El identificador de host o hostid.

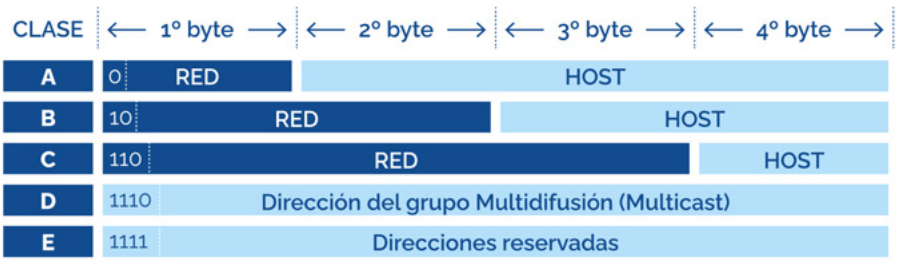

Imagen 4. Clases de direcciones IPv4

La asignación de direcciones que se propone anteriormente es el llamado esquema de direccionamiento con clase.

Direcciones de Clase A

Este tipo de direcciones usa un byte para indicar la red. Su primer bit siempre tiene un valor fijo que es 0, lo que hace que queden disponibles otros 7 bits y también se tienen que descartar las direcciones 0 y 127 ya que se encuentran reservadas.

Por lo tanto, el número de redes que se pueden obtener son $2^7 - 2 = 126$.

Los otros tres bytes se usan para indicar el número de host dentro de cada red, lo que hace posible que se puedan tener $2^{24} - 2$ host, es decir, 1 677 214.

Por lo tanto, si hacemos una suma tendremos que el número de direcciones totales que podemos obtener son 2^{31}, es decir, 2 147 483 648 direcciones distintas. Este espacio de direcciones supone el 50% del total de direcciones de IPv4.

NOTA

Como hay 27 host posibles por cada red y tenemos 224 posibilidades de red, el número total de direcciones se calcula haciendo el producto de ambos.

Siguiendo las propiedades de las potencias, sería:

$2^7 * 2^{24} = 2^{(7+24)} = 2^{31}$

Direcciones de Clase B

Se usan dos bytes en esta clase de direcciones para indicar la red, con los dos primeros bits tomando el valor fijo 10. Esto se traduce en que quedan libres 14 bits.

Por tanto, el número de redes disponible es 2^{14}, es decir, 160384.

En cada red se usarán dos bytes para indicar los hosts de cada red, quedando el número máximo de host por red en $2^{16} - 2 = 650534$.

Queda por lo tanto un total de 2^{30} direcciones posibles, 1 073 741 824, que suponen un total del 25% del espacio de direcciones de IPv4.

Direcciones de Clase C

Las direcciones de clase C usan tres bytes para indicar la red, con tres bits con un valor fijo de 110, quedando libres 21 bits.

Por tanto, tenemos un máximo de 2^{21} o 2 097 152 redes.

Solo nos queda un byte libre para que se indique el host dentro de cada una de las redes, por lo que hay un máximo de $2^8 - 2$ host posibles por red, es decir, 254.

Las direcciones totales de que tenemos ahora son 2^{29} (536 870 912). Estas comprenden un 12,5% del espacio total de IPv4.

Direcciones de Clase D

Son las llamadas direcciones Multicast, donde los 4 primeros bits siempre tienen el valor 1110.

Estas direcciones nos otorgan la oportunidad de enviar los datagramas a un grupo en concreto de estaciones de una subred donde son más de uno los destinatarios. En estas direcciones tampoco se hace un barrido a todas las direcciones, que es el llamado broadcast.

Comprenden un total de 2^{28} o 268 435 456 direcciones, es decir, un 6,25% de las direcciones totales de IPv4.

Direcciones de Clase E

Son las direcciones que se reservan para un uso experimental donde los 4 primeros bits toman el valor 1111 de manera fija.

Al igual que las direcciones de Clase D, son un total de 228 direcciones disponibles y ocupan un 6,25% del total del espacio de direcciones de IPv4.

En el siguiente cuadro se puede apreciar un resumen completo de las distintas clases:

Primer octeto de cada clase				
Clase	Bits iniciales	Rango de direcciones (primer octeto)	Número de redes	Número de host por red
A	0	1 – 126	126	16 777 214
B	10	128 – 191	16 384	65 534
C	110	192 – 223	2 097 152	254

4.3.3. Direcciones reservadas

Estas son las direcciones que un host no puede tomar dentro de una red:

- **Dirección de red.** Son las direcciones que se usan para indicar una red completa.
- **Dirección de difusión a toda la red o broadcast.** Este tipo de direcciones nos permite enviar cierta información a todos los elementos de la red.

El rango válido de direcciones IP se encuentra entre ambos valores.

4.3.4. Direcciones especiales

Son un tipo de direcciones que sí que puede que un host tome, pero no identifican como tal a un elemento, sino que tienen un significado muy específico.

Direcciones IP especiales		
Nombre	**Dirección de red**	**Descripción**
Mi propio host	0.0.0.0	Se trata de la dirección que tiene un equipo asignado antes de que tenga una configuración de red, Los routers también la usan cuando no tienen disponible otra mejor.
Bucle local o Loopback	127.0.0.1 – 127.255.255.255	Se trata de la dirección local que se usa para pruebas, si se envía información a dicha dirección no salen al exterior además de ser tratados como paquetes de entrada.
Enlace local	169.254.0.0 – 169.254.255.255	Si falla la configuración de IP dinámica del sistema, se asigna automáticamente esta dirección el sistema. Son las llamadas APIPA.

4.3.5. Direcciones públicas

Para que se puedan visualizar desde internet, los equipos deben de tener una dirección IP que sea pública y se haya configurado manualmente en la estación de red, La institución internacional que se encarga de la asignación de dichas direcciones a cada una de las organizaciones que las soliciten es el ICANN, con el propósito de que no se repitan dichas direcciones.

Esta institución a su vez delega las gestiones en otros cinco Registros Regionales de Internet, RIR, que son:

- ARIN, para América Anglosajona.
- RIPE NCC para Europa, Oriente Medio y Asia Central.
- APNIC para Asia y la región ubicada en el Pacífico.
- LACNIC para América Latina.
- AFRINIC para África.

Estas instituciones solo otorgan el rango de direcciones a una organización, ya que es tarea del administrador asignar a cada host su determinada IP.

4.3.6. Direcciones privadas

Cada empresa tiene su propia red interna donde todos los equipos se encuentran interconectados entre ellos y además algunos con conexión a internet, pero por lo general no tienen la posibilidad de ser vistos desde Internet. Para identificar a estos equipos se usan las direcciones privadas que no tienen coste y que con el mecanismo llamado NAT, al llegar al router se transforman en una dirección pública. Este mecanismo se verá más adelante en este mismo módulo.

Existen los siguientes intervalos de red para direcciones privadas;

Direcciones privadas (direccionamiento con clase)			
Clase	Red	Número de redes	Rango de direcciones total
A	10.0.0.0	1	10.0.0.0 – 10.255.255.255
B	Desde 172.16.0.0 hasta 172.31.0.0	16	172.16.0.0 – 172.31.255.255
C	Desde 192.168.0.0 hasta 192.168.255.0	256	192.168.0.0 – 192.168.255.255

Como hemos comentado antes, las direcciones privadas son únicas en cada organización o empresa distinta, pero pueden repetirse entre ellas diferenciando de las públicas que son únicas en todo el mundo.

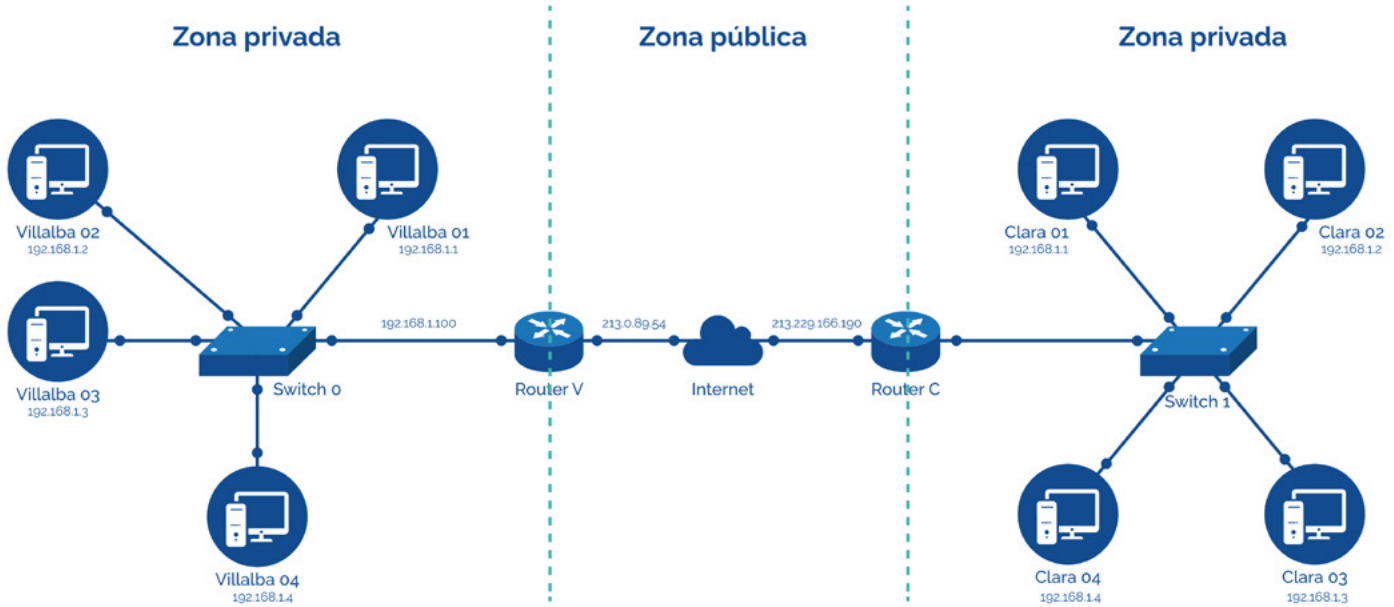

Imagen 5. Direcciones privadas y públicas

4.3.7. Máscaras de subred

Una máscara de subred es una dirección que enmascara una dirección IP con la intención de indicarnos las IPs máximas que pueden tener en la misma red.

Máscaras de subred de cada clase		
Clase	Máscara de subred en sistema decimal con puntos	Máscara de subred en sistema binario
A	255.0.0.0	11111111 00000000 00000000 00000000
B	255.255.0.0	11111111 11111111 00000000 00000000
C	255.255.255.0	111111111 11111111 11111111 00000000

En la notación binaria los números indican la parte que va a corresponder a la red y los ceros la parte que corresponde a los hosts.

Entonces, si lo que queremos es saber si una dirección en concreto pertenece a nuestra misma red, las máquinas realizarán una operación lógica entre la IP y la máscara:

Dirección de red = (Dirección IP) **AND** (Máscara de subred)

Esta operación se puede realizar de forma inmediata en dos ocasiones:

- Cuando el byte de la máscara es 255 → el byte en la dirección se repetirá.
- Cuando el byte de la máscara es 0 → el byte de la subred se quedará en 0.

Si lo que queremos calcular ahora es la dirección de difusión de una red, haremos lo siguiente:

Dirección de difusión = (Dirección IP) **OR (NOT** (Máscara de subred))

De nuevo, podemos realizar esta operación de manera inmediata sí:

- El byte de la máscara es 255 → el byte de la dirección se repetirá.
- El byte de la máscara es 0 → el byte de la dirección de difusión es 255.

Notación CIDR o notación con barra

Esta notación lo que hace es exponer la dirección IP y justo a continuación una barra que tiene seguidamente el número de unos que se contiene en la máscara de subred, un ejemplo de esto sería 192.168.1.10/24.

4.4. Subredes

4.4.1. Necesidad de las subredes

Como se ha dicho anteriormente, las direcciones de IPv4 constan de 32 bits en los que una parte indica la dirección de la red y la otra la del host en específico. Esto se traduce en una jerarquía en las direcciones IP.

Si desde Internet se quiere alcanzar un host, primeramente, hay que identificar la red en la que se encuentra y luego en la red ubicarnos en el host.

Realmente, hay dos esquemas de jerarquía en la que las estaciones no se agrupan y se encuentran todas en el mismo nivel.

Para que no existan solo estos dos niveles, se crean subredes, que son una división de las redes en otras de menor rango interconectadas por routers.

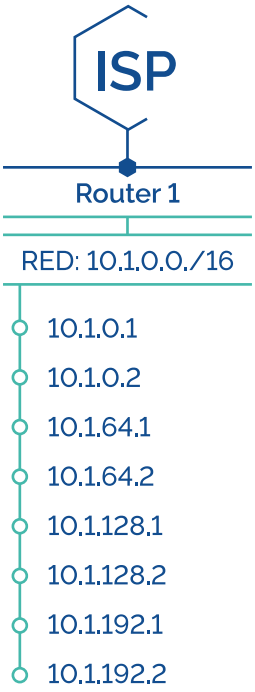

Imagen 6. Dos niveles de Jerarquía

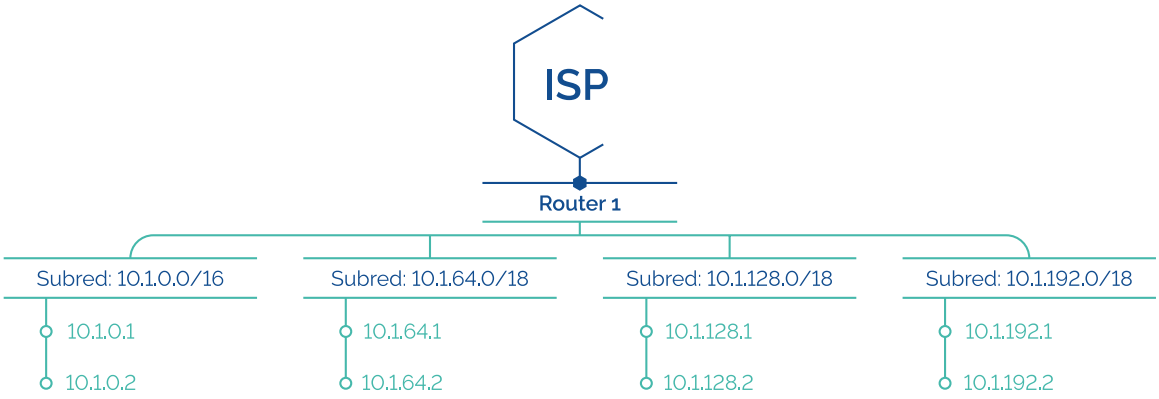

Imagen 7. Red con subredes

En el diagrama anterior podemos ver que las cuatro subredes primero aparecían como una única red que accede a Internet, lo que hacía tener que pasar por todos los hosts para enviar un paquete, con la división, se va directo.

Si por ejemplo quisiéramos enviarlo a la dirección: 10.1.64.1, tendríamos dos modos:

Sin subredes

Es una dirección de clase B que pertenece a la red 10.1.0.0, y en nuestro ejemplo pasaríamos por dos hosts antes de llegar.

Dirección IP sin subred	
Red	Host
10.1.	64.1

Con subredes

Es una dirección de clase B que en este caso pertenece a la subred 10.1.64.0/18 y ahora pasaríamos directamente a este host a través de la subred.

Dirección IP con subred		
Red	Subred	Host
10.1.	64.	1

Si nos fijamos en los ejemplos, si lo que queremos es crear subredes, deberemos de quitar a la red original bits que estaban destinados a los hosts.

Calcular la dirección de subred

Si queremos calcular el identificador de una subred, se usarán los siguientes tres criterios:

1. Si el byte de la máscara es 255 el byte de la dirección IP se repite.
2. Si el byte de la máscara es 0, el byte de la dirección también será 0.
3. Los demás bytes se pasarán a binario y se usará el operador lógico AND

Un ejemplo sería:

Dirección IP:	10.	1.	64.	1		01000000
Máscara:	255.	255.	192.	0	AND	11000000
Subred:	10.	1.	64.	0		01000000

4.4.2. Método para la creación de subredes (subnetting)

Para poder explicar esto de manera adecuada, se hará con 3 ejemplos distintos.

Ejemplo 1, dirección CLASE A.

Tenemos la dirección de red 10.0.0.0 y queremos crear N = 7 subredes.

Tenemos entonces que hacer lo siguiente:

- Hallar la máscara de la subred.
- Calcular el número de host por cada una de las subredes.
- Hallar la dirección de cada subred, el intervalo de sus direcciones IP válidas para los hosts y su dirección de difusión.

Solución:

Si queremos crear N subredes, tendremos que saber que: $\mathbf{2^{nº\ de\ bits\ de\ subred} >= N}$

El número de bits de nuestra subred tendrá que ser por lo tanto 2^3 (8 > 7).

Se rellena con ceros hasta hacer un byte y lo pasamos a decimal: 11100000 = 224.

La máscara de subred será por lo tanto **255.224.0.0 = 11111111 .11100000.00000000.00000000 = /11.**

Esto desencadena el que el número de host por subred será $2^{21} - 2$, es decir, 2 097 150.

Si queremos crear las subredes, crearíamos una tabla como la siguiente:

7 subredes (obviando la nula) de la red original				
Nº de subred	2º Byte	Dirección de subred	Rango de host	Dirección de Broadcast
1	001 00000 = 32	10.32.0.0.	10.32.0.1 – 10.63.255.254	10.63.255.255
2	010 00000 = 64	10.64.0.0	10.64.0.1 – 10.95.255.254	10.95.255.255
3	011 00000 = 96	10.96.0.0	10.96.0.1 – 10.127.255.254	10.127.255.255
4	100 00000 = 128	10.128.0.0.	10.128.0.1 – 10.159.255.254	10.159.255.255
5	101 00000 = 160	10.160.0.0.	10.160.0.0 – 10.191.255.254	10.191.255.255
6	110 00000 = 192	10.192.0.0	10.192.0.1 – 10.223.255.254	10.223.255.255
7	111 00000 = 224	10.224.0.0.	10.224.0.1 – 10.255.255.254	10.255.255.255

Si cogemos tres bits para formar una subred podremos crear hasta 23 subredes, siempre empezando por 0, por lo que irían de la 0 a la 7.

Ejemplo 2, dirección CLASE B.

Tenemos la dirección de red 172.16.0.0 y queremos crear N = 10 subredes.

Deberemos de realizar lo siguiente:

- Hallar la máscara de subred.
- Calcular el número de host que puede haber en cada subred.
- Hallar el intervalo de direcciones en el que cada subred podrá otorgar direcciones a los hosts y saber la dirección de difusión.

Solución:

El número de bits para formar la subred es 4.

Rellenamos con ceros y pasamos a decimal, quedando: 11110000 = 240.

La máscara de subred es, por tanto,

255.255.240.0 = 11111111.11111111.11110000.00000000 = /20.

El número de host por subred máximo será $2^{12} - 2 = 4094$ (32 – 20 = 12).

10 subredes empezando por la 1 de la red original				
Nº	3º Byte	Dirección de subred	Rango de host	Dirección de broadcast
1	0001 0000 = 16	172.16.16.0	172.16.16.1 – 172.16.31.254	172.16.31.255
2	0010 0000 = 32	172.16.32.0	172.16.32.1 – 172.16.47.254	172.16.47.255
3	0011 0000 = 48	172.16.48.0	172.16.48.1 – 172.16.65.254	172.16.63.255
4	0100 0000 = 64	172.16.64.0	172.16.64.1 – 172.16.79.254	172.16.79.255
5	0101 0000 = 80	172.16.80.0	172.16.80.1 – 172.16.95.254	172.16.95.255
6	0110 0000 = 96	172.16.96.0	172.16.96.1 – 172.16.111.254	172.16.111.254
7	0111 0000 = 112	172.16.112.0	172.16.112.1 – 172.16.127.254	172.16.127.255
8	1000 0000 = 128	172.16.128.0	172.16.128.1 – 172.16.143.254	172.16.143.255
9	1001 0000 = 144	172.16.144.0	172.16.144.1 – 172.16.159.254	172.16.159.255
10	1010 0000 = 160	172.16.160.0	172.16.160.1 – 172.16.175.254	172.16.175.255

Como se usan 4 bits para formar la subred, tenemos hasta 24 posibilidades distintas de subredes que formar, empezando siempre por la 0, hasta la 15.

Ejemplo 3, dirección CLASE C.

Vamos a crear N = 6 subredes a partir de la dirección que determina la red 192.168.123.0.

Tendremos que hacer lo siguiente:

- Hallar la máscara de la subred.
- Calcular cuantas direcciones IP máximas se pueden usar para host y su intervalo por cada subred.
- Hallar la dirección de la subred, además de la dirección de difusión.

Solución:

El número de bits para formar la subred es igual a 3.

Rellenamos con ceros los demás y pasamos a decimal para poder obtener la máscara de subred.

11100000 = 224.

La máscara de subred es, por tanto,

255.255.255.224 = 11111111.11111111.11111111.11100000 = /27.

Los hosts por subred por tanto se quedan en 25 .2 = 30 (5 porque son el número de ceros en la dirección de la máscara).

6 subredes para una dirección de red de Clase C				
Nº	4ª Byte	Dirección de subred	Rango de host	Dirección de broadcast
1	001 00000 = 32	192.168.123.32	192.168.123.33 – 192.168.123.62	192.168.123.63
2	010 00000 = 64	192.168.123.64	192.168.123.65 – 192.168.123.94	192.168.123.95
3	011 00000 = 96	192.168.123.96	192.168.123.97 – 192.168.123.126	192.168.123.127
4	100 00000 = 128	192.168.123.128	192.168.123.129 – 192.168.123.158	192.168.123.159
5	101 00000 = 160	192.168.123.160	192.168.123.161 – 192.168.123.190	192.168.123.191
6	110 00000 = 192	192.168.123.192	192.168.123.193 – 192.168.123.222	192.168.123.223

4.4.3. Herramientas para subnetting

Para IPv4, tenemos dos herramientas que funcionan muy bien. La primera y más usada es Online IP Subnet Calculator.

Su URL es https://www.subnet-calculator.com/

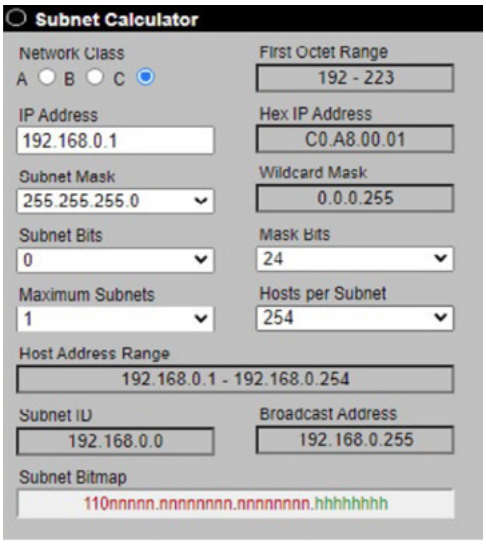

Imagen 8. Online IP Subnet Calculator

La otra que también funciona muy bien es VLSM (CIDR) Subnet Calculator.

Su URL es http://vlsmcalc.net/

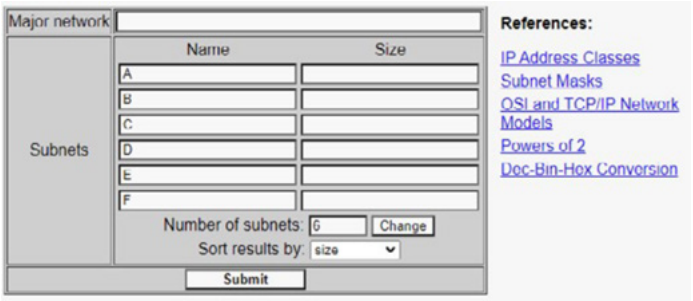

Imagen 9. VLSM Subnet Calculator

Para IPv6, tenemos la aplicación online Calculadora de Redes. (También funciona con IPv4).

Su URL es https://www.calculadora-redes.com/ipv6.php

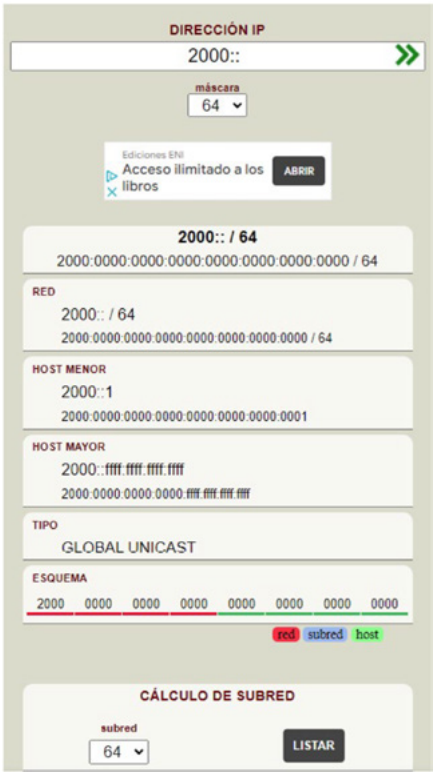

Imagen 10. Calculadora de redes

4.5. Configuración del adaptador de red

Hay tres tipos de direcciones IP que se configuran de manera normal:

- **Estática:** la dirección IP es asignada de manera manual, ya sea por un administrador de toda la red o del equipo local.
- **Dinámica:** la dirección IP viene designada por un servidor DHCP nada más haya conexión a red.
- **Alternativa:** si no se cumple ninguna de las anteriores, se asigna una por el equipo de manera aleatoria.

4.5.1. Configuración en Windows

Windows instala por defecto los protocolos TCP/IP del tipo IPV4 e IPv6 la primera vez que detecte una interfaz de red activa.

Esto hará que se cree una conexión local que es configurable en cada adaptador de red disponible en el equipo, y para realizar dicha configuración debemos de seguir los siguientes pasos:

1. Primero vamos hasta **Inicio → Panel de control → Redes e Internet → Centro de redes y recursos compartidos**
2. Seleccionamos a la izquierda la opción "Cambiar configuración del adaptador"

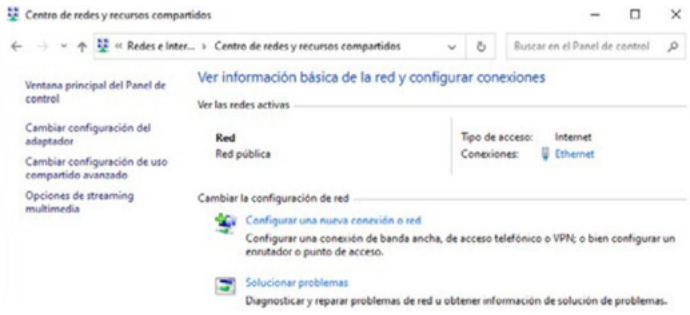

Imagen 11. Configuración de red en Windows 1

3. Lo siguiente que haremos será seleccionar el adaptador que deseamos, realizar click derecho y seleccionar **"Propiedades"**.

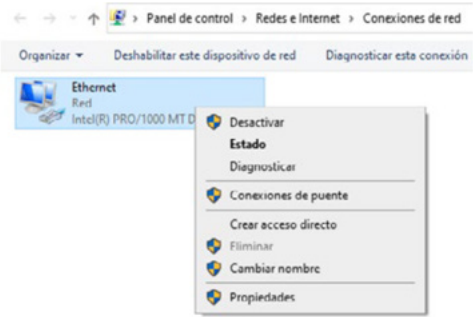

Imagen 12. Configuración de red en Windows 2

4. Veremos que entonces aparecen un listado con todos los protocolos de red que registra dicho adaptador de red y los que están marcados quiere decir que están desinstalados.

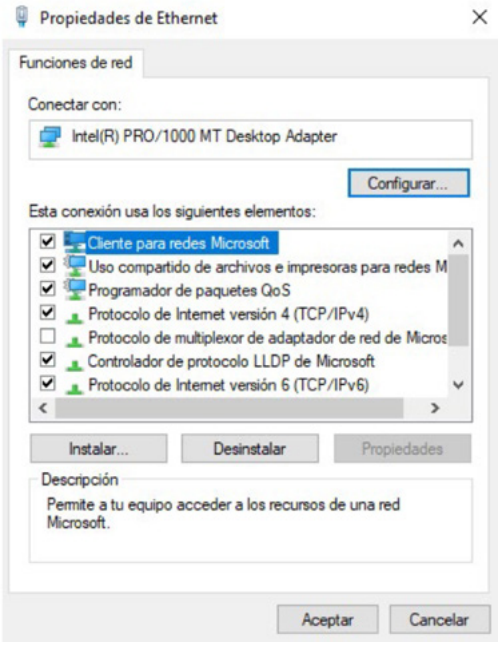

Imagen 13. Configuración de red en Windows 3

5. El protocolo que a nosotros nos interesa es IPv4, que como podemos ver pone "Protocolo de Internet versión 4 (TCP/IPv4)".

6. Lo seleccionamos y veremos que tiene dos secciones principales, una para la IP y otra para la dirección DNS.

7. Por defecto, como podremos ver, Windows tiene de manera general activada la opción de DHCP, es decir, obtener tanto la dirección IP como los servidores DNS de manera automática.

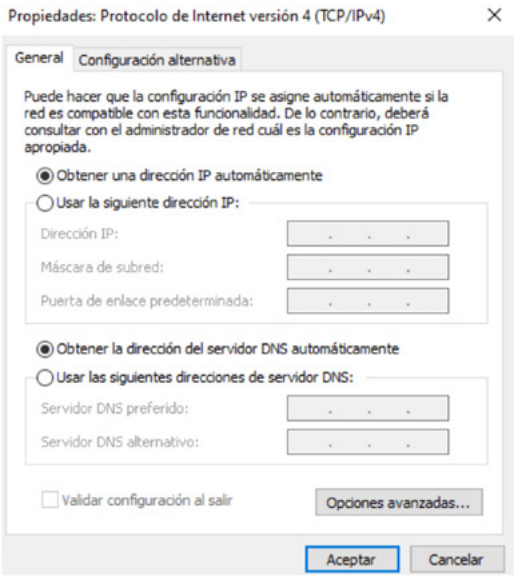

Imagen 14. Configuración de red en Windows 4

8. Para realizar una configuración manual de las direcciones IP del equipo y servidores DNS, deberíamos seleccionar la opción "Usar la siguiente dirección IP" y "Usar las siguientes direcciones de servidor DNS". Pero para realizar esta acción, debemos de tener un conocimiento de las IPS y su funcionamiento, que aún no tenemos, es por esto por lo que, de momento, lo dejaremos automático.

9. Por último, tenemos la opción de "Opciones avanzadas", que nos muestra lo siguiente:

Imagen 15. Configuración de red en Windows 5

10. Estas opciones sirven para que pueda haber más de una dirección IP a la vez, para una configuración más específica del DNS y para habilitar dominios de Windows respectivamente.

Vamos ahora a comprobar que tenemos conexión a internet, para eso lo primero que vamos a hacer es comprobar si tenemos realmente IP.

Nos vamos a:

Inicio → Símbolo del Sistema

Una vez aquí ejecutamos el siguiente comando:

```
ipconfig /all
```

Y debe de mostrarnos una serie de atributos como podemos ver a continuación:

```
C:\Users\Miguel>ipconfig /all

Configuración IP de Windows

   Nombre de host. . . . . . . . . . : DESKTOP-ERDMQUM
   Sufijo DNS principal  . . . . . :
   Tipo de nodo. . . . . . . . . . . : híbrido
   Enrutamiento IP habilitado. . . . : no
   Proxy WINS habilitado . . . . . : no

Adaptador de Ethernet Ethernet:

   Sufijo DNS específico para la conexión. . :
   Descripción . . . . . . . . . . . . . . . : Intel(R) PRO/1000 MT Desktop Adapter
   Dirección física. . . . . . . . . . . . . : 08-00-27-F8-19-F6
   DHCP habilitado . . . . . . . . . . . . . : sí
   Configuración automática habilitada . . . : sí
   Vínculo: dirección IPv6 local. . . : fe80::8183:19cb:11c6:f0ba%6(Preferido)
   Dirección IPv4. . . . . . . . . . . . . . : 10.0.2.15(Preferido)
   Máscara de subred . . . . . . . . . . . . : 255.255.255.0
   Concesión obtenida. . . . . . . . . . . . : martes, 25 de enero de 2022 7:05:47
   La concesión expira . . . . . . . . . . . : miércoles, 26 de enero de 2022 7:05:49
   Puerta de enlace predeterminada . . . . . : 10.0.2.2
   Servidor DHCP . . . . . . . . . . . . . . : 10.0.2.2
   IAID DHCPv6 . . . . . . . . . . . . . . . : 101187623
   DUID de cliente DHCPv6. . . . . . . . . . : 00-01-00-01-29-80-22-18-08-00-27-F8-19-F6
   Servidores DNS. . . . . . . . . . . . . . : 10.10.30.1
   NetBIOS sobre TCP/IP. . . . . . . . . . . : habilitado
```

Imagen 16. ipconfig /all

Aquí podemos observar que nos da un detalle más amplio de lo que realmente tenemos en red, ya que nos dice la IP, la MAC, la Máscara de red, la puerta de enlace predeterminada, etc.

Como podemos ver en la imagen anterior nuestra IP es 10.0.2.15, entonces, ahora vamos a otro equipo y realizaremos la acción ping para comprobarlo.

Ping es una herramienta provista por el 90% de los equipos con sistema operativo y red que lanza una serie de paquetes a una dirección en concreto, ya sea IP o DNS y nos avisa de la pérdida que hay, el tiempo que tarda y si está disponible dicha dirección, es decir, si tenemos conexión con esa dirección. Lógicamente, la comprobación la haremos desde otro equipo y queda tal que así:

```
C:\Users\Miguel>ping 10.0.2.15

Haciendo ping a 10.0.2.15 con 32 bytes de datos:
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 10.0.2.15:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

Imagen 17. ping

Como podemos ver, hay conexión y llegamos a este equipo, por lo que ya podríamos decir que tenemos el equipo listo para conectarse a la red y trabajar.

Opciones de C:\>ipconfig y su significado

- **C:\>ipconfig** → Muestra información acerca de la configuración del protocolo TCP/IP.
- **C:\>ipconfig /all** → Muestra una información más detallada.
- **C:\>ipconfig /displaydns** → Muestra la caché de resolución de DNS.
- **C:\>ipconfig /flushdns** → Vacía y reinicia la caché de DNS.
- **C:\>ipconfig /release** → Libera la IP concedida por el DHCP.
- **C:\>ipconfig /renew** → Renueva la concesión de IP por DHCP.

4.5.2. Configuración en Linux

Para la configuración de la red TCP/IP en Linux y del DNS tenemos varias opciones, una gráfica y otra por la línea de comandos o Terminal.

Vamos a ver la primera, en nuestro ejemplo usaremos Debian, que es una distro de Linux de las más usadas. Vamos a ver el proceso:

1. Lo primero que haremos será desplazarnos a la esquina superior derecha donde vemos que se encuentra el incono de red y nos da acceso a sus preferencias.

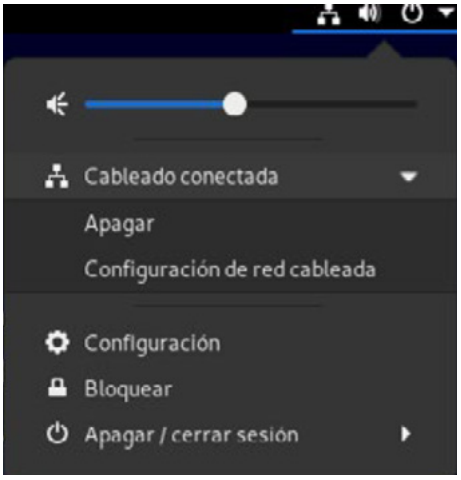

Imagen 18. Configuración de red en Debian 1

2. Una vez que estemos en la configuración de red veremos que tenemos varias opciones, entre ellas destaca la que nosotros queremos, que es la de red y que vemos en la siguiente imagen que nos aparece para configurar la conexión cableada (aparece la velocidad), la VPN, y el Proxy de red.

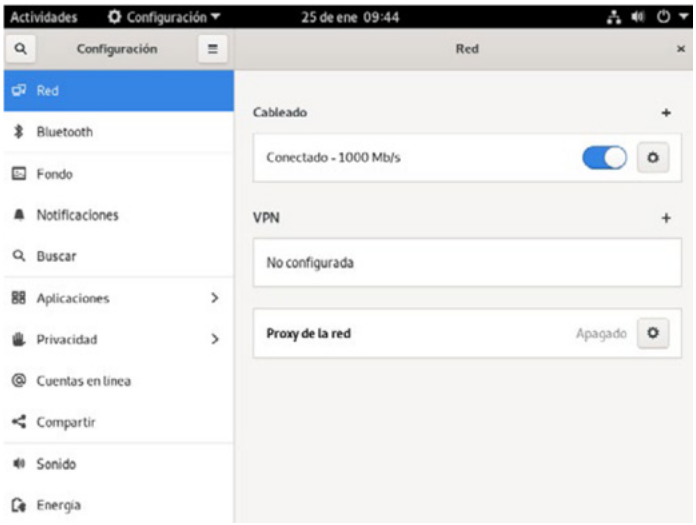

Imagen 19. Configuración de red en Debian 2

Aquí hay que destacar que en nuestro equipo no tenemos conexión Wifi, pero también aparecería ahí.

3. Si lo abrimos veremos que hay varias pestañas, relativas: Detalles, Identidad, IPv4, IPv6 y Seguridad.

4. En la primera pestaña podemos ver que nos dice la dirección IPv4 e IPv6 que nos corresponde, la MAC, el Gateway y el DNS predefinido además tres opciones que a la hora de la verdad no se suelen usar mucho. Sí que hay que poner ímpetu en que como podemos ver, podríamos eliminar el perfil de conexión, pero nos quedaríamos sin red.

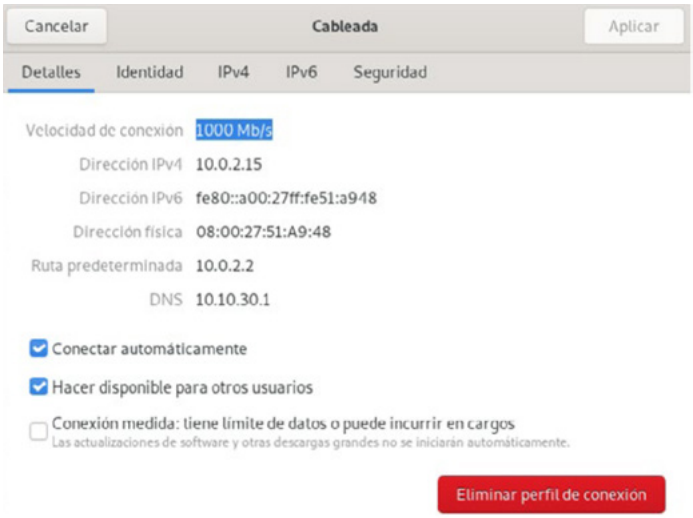

Imagen 20. Configuración de red en Debian 3

5. Vamos a fijarnos simplemente en la pestaña IPv4, que es la que nos interesa ahora mismo, como podemos ver, de momento tenemos como pasaba con Windows, la opción de DHCP activada, y podríamos cambiarla a manual. Además, también tenemos una opción adicional que es desactivarla, activar solo enlace local y compartir la IP con otros equipos, pero de momento, no nos interesan, al igual que las rutas, que se verán en el módulo de Redes de este ciclo. La pestaña se ve tal que así:

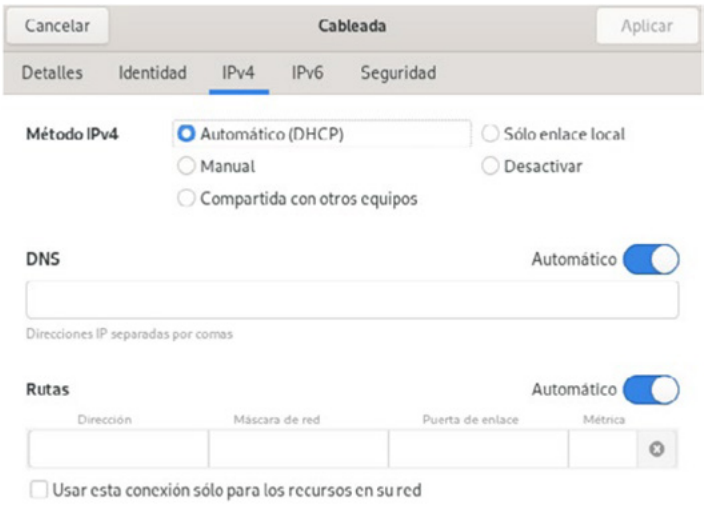

Imagen 21. Configuración de red en Debian 4

Ahora, para configurar la red por la línea de comandos se debería de hacer lo siguiente:

1. Vamos a abrir el terminal, nos vamos a Actividades y buscamos "Terminal".
2. Una vez abierto, tenemos que loguearnos como root, que es el usuario administrador de los sistemas Linux por excelencia, para eso, ejecutamos el comando:

   ```
   su root
   ```

3. Ahora nos pedirá la contraseña de este usuario, la metemos y ya estaríamos logueados como root para poder realizar tareas administrativas en el equipo.

Imagen 22. Configuración de red en Debian 5

4. Una vez aquí, tenemos que conocer como se llama nuestra tarjeta de red, para eso lo que haremos será ejecutar el siguiente comando: ip a. Este comando nos dirá las interfaces de red disponibles en el equipo:

```
root@debian:/home/miguel# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
    link/ether 08:00:27:51:a9:48 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 81341sec preferred_lft 81341sec
    inet6 fe80::a00:27ff:fe51:a948/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Imagen 23. Configuración de red en Debian 6

5. Como vemos, la interfaz de red activa ahora mismo se llama enp0s3.
6. Vale, ahora lo que tenemos que hacer es dirigirnos a la siguiente ruta, /etc/network/interfaces.d, y esto lo hacemos con el comando cd:

```
root@debian:/home/miguel# cd /etc/network/interfaces.d/
root@debian:/etc/network/interfaces.d#
```

Imagen 23. Configuración de red en Debian 7

7. Aquí, si lanzamos el comando ls, veremos que no tenemos ahora mismo ningún fichero, pero en un principio, si quisiéramos realizar la configuración específica, deberíamos de crear un fichero con el nombre de la interfaz y realizar una cierta configuración.

4.6. Protocolo ARP

El protocolo ARP, cuyas siglas hacen referencia a Address Resolution Protocol) se encarga de asociar una dirección IP con una dirección MAC o dirección física.

En todo tipo de red, los dispositivos tienen una dirección MAC asociada a su adaptador de red que es única y ayuda a identificarlos.

El protocolo ARP por tanto se encarga de encontrar dicha dirección física a través del conocimiento de su dirección IP, siempre que estén en la misma red local.

Entonces, este protocolo se usa cuando cualquier dispositivo de nuestra red quiere establecer comunicación con otro con la intención de conocer su dirección física. En este momento el dispositivo envía una petición ARP con la IP de destino a toda la red, respondiendo únicamente el poseedor de dicha dirección con la intención de conocer el emisor dicha dirección física y añadirla a su tabla ARP.

Comando arp

El comando arp que dispone Windows nos ayuda a visualizar y modificar las tablas ARP donde se muestra la asociación de direcciones IP con direcciones físicas.

Sus principales opciones son:

- **C:\>arp -a** → Muestra la tabla ARP para cada una de las interfaces.
- **C:\>arp -s dirección_IP dirección_MAC** → Añade una entrada con estas direcciones a la tabla ARP.
- **C:\>arp -d dirección IP** → Elimina de la tabla la entrada con dicha dirección.

```
PS C:\Users\Administrador> arp -a

Interfaz: 10.0.0.24 --- 0x8
  Dirección de Internet          Dirección física      Tipo
  10.0.0.25             08-00-27-26-79-40     dinámico
  10.255.255.255        ff-ff-ff-ff-ff-ff     estático
  224.0.0.22            01-00-5e-00-00-16     estático
  224.0.0.251           01-00-5e-00-00-fb     estático
  224.0.0.252           01-00-5e-00-00-fc     estático
  239.255.255.250       01-00-5e-7f-ff-fa     estático

Interfaz: 10.0.2.15 --- 0xf
  Dirección de Internet          Dirección física      Tipo
  10.0.2.2              52-54-00-12-35-02     dinámico
  10.0.2.255            ff-ff-ff-ff-ff-ff     estático
  224.0.0.22            01-00-5e-00-00-16     estático
  224.0.0.251           01-00-5e-00-00-fb     estático
  224.0.0.252           01-00-5e-00-00-fc     estático
  239.255.255.250       01-00-5e-7f-ff-fa     estático
  255.255.255.255       ff-ff-ff-ff-ff-ff     estático
PS C:\Users\Administrador>
```

Imagen 24. arp -a

Este ejemplo está sacado de nuestro servidor de pruebas donde tenemos dos tarjetas de red, una para la red interna, que es la primera y la segunda que nos da conexión con internet, pero se puede observar que ambas muestran la misma información.

4.7. Protocolo ICMP. Diagnóstico de redes

Como se ha nombrado antes, el protocolo IP no es 100% fiable, ya que puede haber una pérdida de datagramas o llegar defectuosos o desordenados. Para intentar controlar más esta situación existe el protocolo **ICMP**, Internet Control Message Protocol, que tiene la función de hacerle saber al remitente si ha habido algún error en la entrega del datagrama. El protocolo ICMP solo comunica el error, es trabajo correspondiente a las capas superiores el intentar subsanar este error.

Además de comunicar solamente el error, su función también es transportar distintos tipos de mensajes de control que viajan en el campo de datos del datagrama IP.

Imagen 25. Mensaje ICMP encapsulado en un datagrama IP

Los mensajes que se mandan en ICMP, ya sea ICMPv4 o ICMPv6, tienen una cabecera en la que se incluyen tres campos: tipo, código y CheckSum. El área de datos dependerá del tipo de mensaje enviado.

Imagen 26. Formato de mensaje ICMP

En el siguiente cuadro se detallan los tipos de mensajes de ICMPv4 más comunes:

Tipos de mensajes ICMPv4 más comunes		
Tipo	**Significado**	
0	Echo Reply (Respuesta de eco)	
3	Destination Unreachable (Destino inalcanzable)	Código
		0 = Red inaccesible
		1 = Host inaccesible
		2 = Protocolo inaccesible
8	Echo Request (Solicitud de eco)	
11	Time Exceeded (Tiempo excedido)	

4.7.1. Mensajes de solicitud y respuesta de eco (Echo y Echo Reply)

Dos mensajes muy importantes en cuanto a ICMPv4 se refiere son los de solicitud y respuesta de eco, que tienen respectivamente el código 8 y 0. Estos mensajes se usan para comprobar si existe comunicación entre dos hosts de la red.

Estos mensajes se encargan de comprobar que, en las siguientes capas: física, acceso al medio y de red, se encuentra todo correcto.

Hay que tener en cuenta que solo se fija en estas capas, por lo que no nos aporta información acerca de las capas de transporte y aplicación que también podrían fallar.

Para poder detectar estos mensajes el principal comando usado es el comando **PING**, que es la herramienta básica de comprobación de comunicación en la red.

```
PS C:\Users\Administrador> ping

Uso: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
          [-r count] [-s count] [[-j host-list] | [-k host-list]]
          [-w timeout] [-R] [-S srcaddr] [-c compartment] [-p]
          [-4] [-6] nombre_destino

Opciones:
    -t                 Hacer ping al host especificado hasta que se detenga.
                       Para ver estadísticas y continuar, presione
                       Ctrl-Interrumpir; para detener, presione Ctrl+C.
    -a                 Resolver direcciones en nombres de host.
    -n count           Número de solicitudes de eco para enviar.
    -l size            Enviar tamaño de búfer.
    -f                 Establecer marca No fragmentar en paquetes (solo IPv4).
    -i TTL             Período de vida.
    -v TOS             Tipo de servicio (solo IPv4. Esta opción está desusada y
                       no tiene ningún efecto sobre el campo de tipo de servicio
                       del encabezado IP).
    -r count           Registrar la ruta de saltos de cuenta (solo IPv4).
    -s count           Marca de tiempo de saltos de cuenta (solo IPv4).
    -j host-list       Ruta de origen no estricta para lista-host (solo IPv4).
    -k host-list       Ruta de origen estricta para lista-host (solo IPv4).
    -w timeout         Tiempo de espera en milisegundos para cada respuesta.
    -R                 Usar encabezado de enrutamiento para probar también
                       la ruta inversa (solo IPv6).
                       Por RFC 5095 el uso de este encabezado de enrutamiento ha
                       quedado en desuso. Es posible que algunos sistemas anulen
                       solicitudes de eco si usa este encabezado.
    -S srcaddr         Dirección de origen que se desea usar.
    -c compartment     Enrutamiento del identificador del compartimiento.
    -p                 Hacer ping a la dirección del proveedor de virtualización
                       de red de Hyper-V.
    -4                 Forzar el uso de IPv4.
    -6                 Forzar el uso de IPv6.

PS C:\Users\Administrador>
```

Imagen 27. Comando ping y sus opciones

A continuación, vamos a ver una serie de opciones de ping, un proceso que es adecuado seguir para detectar problemas de red:

1. **Lo primero que comprobaremos será el ping al bucle local (localhost).**
    a. `C:\>ping 127.0.0.1`
    b. Si tenemos un error con este ping quiere decir que el protocolo TCP/IP se encuentra en mal estado y deberíamos de reinstalarlo.
    c. Si funciona seguimos con el siguiente paso

```
PS C:\Users\Administrador> ping 127.0.0.1

Haciendo ping a 127.0.0.1 con 32 bytes de datos:
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 127.0.0.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\Users\Administrador>
```

Imagen 28. Ping a localhost

2. Ping a la dirección IP de nuestro equipo

   a. `C:\>ping dirección_IP_equipo`

   b. Si nos da error, es porque seguramente haya un problema con la tarjeta de red o con el cable, las medidas principales serían reinstalar los controladores, comprobar la tarjeta y testear el cable.

   c. Si esto funciona, pasamos al paso siguiente.

```
PS C:\Users\Administrador> ping 10.0.0.24

Haciendo ping a 10.0.0.24 con 32 bytes de datos:
Respuesta desde 10.0.0.24: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.0.24: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.0.24: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.0.24: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 10.0.0.24:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\Users\Administrador> ping 10.0.2.15

Haciendo ping a 10.0.2.15 con 32 bytes de datos:
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.15: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 10.0.2.15:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\Users\Administrador>
```

Imagen 29. Ping a ambas direcciones IP del equipo

3. Ping a la puerta de enlace predeterminada o gateway.

   a. **C:\>ping dirección_IP_gateway**

   b. Si este apartado nos da error quiere decir que posiblemente no esté bien configurado el gateway de la red o que haya un problema con algún otro dispositivo intermedio de red. Las medidas a tomar son: comprobar que funciona de manera correcta el router o proxy, testear los dispositivos que hay hasta llegar a la puerta de enlace, comprobar el estado del cable y comprobar la configuración del gateway.

   c. Si funciona pasamos al siguiente paso.

```
PS C:\Users\Administrador> ping 10.0.2.2

Haciendo ping a 10.0.2.2 con 32 bytes de datos:
Respuesta desde 10.0.2.2: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.2: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.2: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.0.2.2: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 10.0.2.2:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\Users\Administrador>
```

Imagen 30. Ping a gateway

4. **Ping a otro host de la misma red local.**

   a. `C:\> ping dirección_IP_hostlocal`

   b. Si nos da error puede ser porque el host no responda simplemente, pero también puede haber un problema con el router o proxy o los cables entre ambos equipos.

   c. Las medidas que se deben tomar son: testear los dispositivos de red intermedios, comprobar que no hay aplicaciones que impidan la conexión, comprobar que el otro host se encuentra encendido y conectado a red y comprobar el estado del cable de red.

   d. Si todo funciona pasamos al paso siguiente.

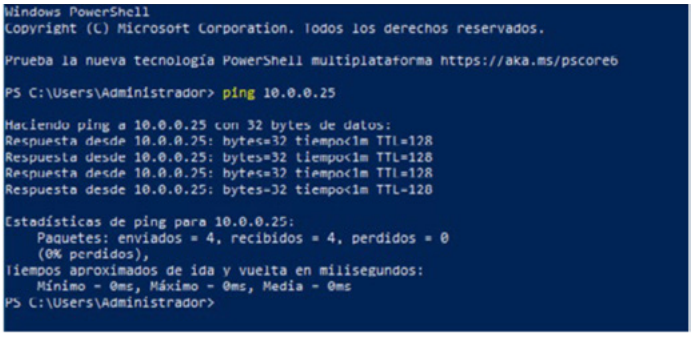

Imagen 31. Ping a host en red local

5. **Ping a un host remoto.**

   a. `C:\>ping dirección_IP_remota`

   b. Si esto falla puede ser porque el host remoto no funcione, pero no debe de ser el caso, a no ser que el host esté saturado o tenga bloqueadas estas peticiones.

   c. De igual modo, las medidas a tomar deben ser: probar otro host remoto y comprobar la configuración del router o proxy.

   d. Si todo funciona bien pasamos al siguiente paso.

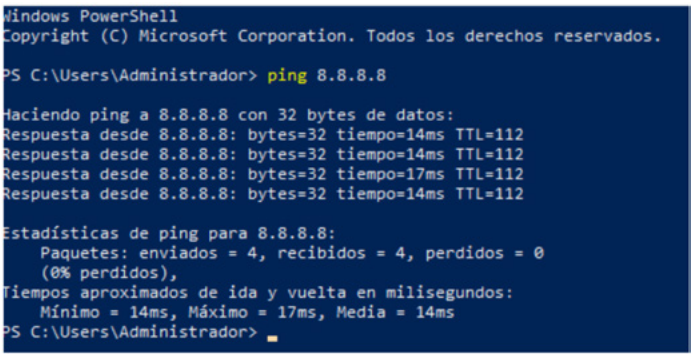

Imagen 32. Ping a google (host remoto)

6. **Ping a la dirección DNS de un host remoto.**

   a. `C:\>ping nombre_host.`

   b. Si se produce un error o bien es por lo anterior, o porque los servidores DNS no se encuentra bien configurados o sin respuesta.

   c. Las medidas a tomar deben ser las anteriores además de la comprobación de nuestro DNS interno y del de los servidores principales.

   d. Si todo funciona tenemos una conectividad completa.

```
PS C:\Users\Administrador> ping google.es

Haciendo ping a google.es [142.250.200.131] con 32 bytes de datos:
Respuesta desde 142.250.200.131: bytes=32 tiempo=16ms TTL=113
Respuesta desde 142.250.200.131: bytes=32 tiempo=15ms TTL=113
Respuesta desde 142.250.200.131: bytes=32 tiempo=15ms TTL=113
Respuesta desde 142.250.200.131: bytes=32 tiempo=16ms TTL=113

Estadísticas de ping para 142.250.200.131:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 15ms, Máximo = 16ms, Media = 15ms
PS C:\Users\Administrador>
```

Imagen 33. Ping a google.es (DNS remoto)

4.7.2. Mensajes de tiempo excedido (Time exceded)

Los datagramas que se envían tienen un campo que indica su TTL o tiempo de vida para que un mensaje al tiempo de ser enviado, si no encuentra destinatario, desaparezca y no esté dando vueltas de manera indefinida por la red.

Cada vez que el datagrama atraviesa un router, este tiempo de vida disminuye en una unidad y cuando llega a 0, el router envía un mensaje ICMP de Tipo 11, Time Exceeded al origen. Este mensaje indica que el tiempo de vida del paquete ha llegado a 0.

El comando **TRACERT** lleva consigo implementado un bucle para el envío de mensaje ICMP tipo 8 con un TTL progresivo de 1 a 30 que indica los saltos necesarios para llegar a un destino. En la siguiente imagen se puede ver los saltos necesarios para llegar a google.es.

4.8. Direccionamiento IPv6

IPv6, Internet Protocol Version 6, es la sexta y más novedosa versión del protocolo IP que se creó para sustituir a IPv4, pero aún se encuentra esta última implementada en la gran mayoría de dispositivos de acceso a la red.

La necesidad de crear e implantar esta nueva versión ha sido debido a que IPv4 se estaba quedando sin direcciones disponibles debido al uso tan generalizado de internet.

Cuando se comenzó con el uso de IPv4 en 1981 no se imaginaba que se necesitaría tal cantidad de direcciones, ya que hay hasta un total de 232 combinaciones de IPv4 posibles (1 294 967 296).

Hay algunos mecanismos que se han ido desarrollando que permiten que el agotamiento de las IPv4 se ralentice, los más eficientes son:

- El uso de redes privadas con NAT que hacen que toda una intranet salga por la misma dirección IP Pública.
- El uso de DHCP.
- El alojamiento de sitios webs virtuales que hacen que varios tengan una misma IP Pública.
- Un exhaustivo control de la asignación de direcciones en los registros regionales de Internet.

Aun así, el número de direcciones disponible sigue decreciendo exponencialmente.

El IANA ya se quedó sin bloques de direcciones disponibles para asignación en 2011 y los registros regionales están cerca de hacerlo.

4.8.1. Formatos de direcciones IPv6

Formato preferido

Las direcciones IPv6 cuentan con 128 bits para medir su longitud y estos se escriben en ocho grupos de cuatro dígitos hexadecimales que se separan entre sí por dos puntos.

Un ejemplo de dirección en IPv6 es: fff:fff:fff:fff:fff:fff:fff:fff (Dirección de broadcast).

Formato comprimido

Los grupos formados por cuatro ceros seguidos se pueden sustituir por el símbolo "::".

Todos los ceros a la izquierda de un grupo también pueden ser eliminados.

Si la dirección tiene más de una sola serie de grupos nulos y no están juntos, solo se puede hacer la distinción en uno de ellos, Porque no hay manera entonces de saber cuántos hay en cada sitio.

Un ejemplo se vería en el siguiente cuadro:

Equivalencia entre formato preferido y formato comprimido de direcciones IPv6	
Dirección IPv6 Preferida	Dirección IPv6 Comprimida
2001:0DB8:AC10:FE01:0000:0000:0000:0000	2001:0DB8:AC10:FE01::
2031:0000:130F:0000:0000:09C0:876A:130B	2031:0000:130F::09C0:876A:130B
FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF	FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF

Dirección IPv6 mapeada

Si tenemos originalmente una dirección IPv4 que queremos convertir en dirección IPv6, debe de realizarse un mapeo para su traducción.

La dirección IPV6 se forma del siguiente modo:

```
5 grupos de ceros:fff;dirección IPV4 que se pasa a dos grupos hexadecimales
```

Por ejemplo:

0:0:0:0:0:fff:10.0.0.24 = ::fff:10.0.0.24 = ::fff:0a00:0018

NOTA

Podemos darnos cuenta de que la dirección anterior tenía un cero solo en cada uno de los grupos, esto es debido a que también se puede poner así cuando el grupo es completamente nulo

Prefijo

El prefijo es la parte de la dirección donde los bits siempre tienen un valor fijo, y se escribe:

```
Prefijo= dirección IPv6/longitud
```

La longitud del prefijo siempre se indicará en sistema decimal, por ejemplo 2001:DB80:AAAA::/64, donde el 64 indica que los 64 primeros bits son fijos y los demás son variables.

En una dirección IPv6 los primeros 64 bits, los primeros 4 grupos, forman la dirección de subred y la segunda mita forma la dirección final de un host.

Como todas sus subredes tienen 64 bits de longitud, no existen las máscaras de red,

Si un prefijo es menor de 64, quiere referirse a que es una ruta o agregado.

4.8.2. Tipos de direcciones IPv6

Unicast (uno a uno)

Las direcciones unicast se usan para identificar una interfaz concreta de un grupo de direcciones IPv6. Un paquete que se envíe a una dirección unicast solo llegará de manera exclusiva a dicha dirección.

Formato de direcciones unicast IPv6:

```
Prefijo de enrutamiento (48-64) + Id. De subred (0-16)
+ Id. De Interfaz (64)
```

Imagen 34. Formato de dirección IPv6 Unicast

Dirección global unicast (Global Unicast Address).

Se trata de una serie de direcciones públicas enrutables que se pueden alcanzar desde cualquier parte de Internet.

Su prefijo es especifico y sus tres primeros bits siempre son 001, por lo que sus direcciones IPv6 públicas solo pueden comenzar con los números 2 o 3.

Dentro de estas direcciones existen los siguientes casos considerados especiales y de excepción:

Direcciones unicast especiales	
Dirección	**Descripción**
::/128	Dirección no especificada. Sería el equivalente a 0.0.0.0 en IPv4.
::1/128	Dirección de Loopback. Sería el equivalente a 127.0.0.0 en IPv4.
Fe80::/64	Prefijo para el enlace local o *link-local*. Estas direcciones se usan para la autoconfiguración del nodo y se asigna de manera predeterminada a todas las interfaces de red. Es el equivalente a la red 169.254.0.0/16 de la APIPA en IPv4. En IPv6 el enlace local es persistente.
Fec0::/48	Prefijo para el sitio local o *site-local*. Se usa para identificar los equipos de manera privada, Coincide con las redes privadas de IPv4.
2001:10::/28	Prefijo para el protocolo ORCHID.
2001:db8::/32	Prefijo para ejemplos.
2001:0:/32	Prefijo para el túnel Teredo desarrollado por Microsoft.
2002:XY:ZT/48	Prefijo para el túnel 6to4. XY:ZT es la representación hexadecimal de la dirección de IPv4 x,y,z,t.
Fe80::5efe:x.y.z.t	Dirección para túnel ISATAP.

Para poder manejar todas las direcciones IPv6 disponemos de una utilidad por comandos llamada netsh.

Para ver las direcciones IPv6 de un equipo se usa el comando:

```
C:\>netsh interface ipv6 show route
```

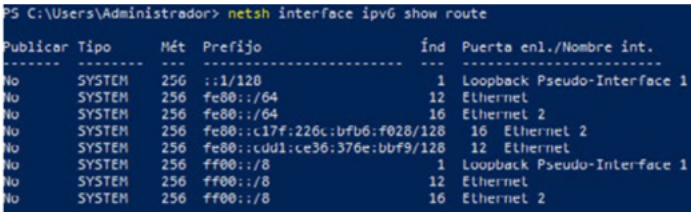

Imagen 35. Listado de direcciones IPv6 de un equipo

Multicast (uno a muchos)

Este tipo de direcciones son usadas para referirse a un grupo concreto de interfaces de IPv6. Un paquete se envía en multicast y entonces es recibido por todos los miembros del grupo. Un host puede encontrarse al mismo tiempo en varios grupos multicast.

La difusión de la información realizada por el protocolo ARP da lugar a un tráfico que reduce la velocidad de la red, es por eso por lo que en IPV6 se sustituye por ciertos grupos multicast.

Imagen 36. Formato de dirección IPv6 multicast

Se encuentra formada por lo siguiente:

- **Prefijo.** Siempre vale 1111 1111 = FF.
- **Flags.** Los tres primeros bits 0, R u P se encuentran reservados, mientras que el último, T, si vale 0 indica que el grupo multicast elegido es una dirección que se encuentra asignada de forma permanente y conocida. Si, por otro lado, el valor de T es 1, se trata de una dirección temporal.
- **Scope.** Indica en que ámbito de la red debe de propagarse el paquete multicast.
  - 1 → Nodo.
  - 2 → Enlace local.
  - 5 → Sitio local.
  - 8 → Organización Local.
  - E → Global.

Group ID. Identificador de grupo.

El Group ID se encarga de identificar a que grupo concreto de multicast nos referimos en un cierto ámbito. Hay una serie de direcciones multicast que se encuentran reservadas y estas son las que van desde FF01:. Hasta FF0F:::. Estos grupos son bien conocidos. Algunas de estas direcciones son las siguientes:

Algunas direcciones multicast reservadas		
Dirección	**Ámbito**	**Descripción**
Ff01::1	Nodo	Todos los nodos en la interfaz local.
Ff01::2	Nodo	Todos los *routers* en la interfaz local.
Ff02::1	Enlace local	Todos los nodos en el enlace local.
Ff02::2	Enlace local.	Todos los *routers* en el enlace local.
Ff02:0:0:0:1:ff00/104	Enlace local.	Nodo solicitado multicast
Ff02::c	Enlace local.	Protocolo SSDP para dispositivos UPnP.
Ff05::2	Sitio local	Todos los *routers* de un sitio

Se puede observar que en el anterior cuadro destacamos el nodo solicitado multicast, que es usado en la resolución de direcciones en una LAN IPv6 para sustituir al protocolo ARP.

Si queremos calcular la dirección del nodo solicitado multicast, cuyo acrónimo es SNMA tenemos que tener en cuenta que su prefijo es FF02:0:0:0:0:1:FF00::/104 donde los 24 bits restantes se rellenan con los 24 últimos bits de la dirección unicast. Luego, el protocolo NDP, siglas de Neighbor Discovery Protocol enviará mensajes multicast a la red para poder descubrir los disantos nodos restantes.

Anycast (uno a la más cercana)

Una dirección IPV6 anycast hace referencia, al igual que la dirección multicast, a varias interfaces de IPv6, pero en este caso solo una de las interfaces recibirá el paquete, normalmente la más próxima o cercana por lo general.

Aunque este tipo de direcciones tienen varios usos, su principal aplicación es para el protocolo de enrutamiento externo BGP usado en Internet.

El formato de las direcciones anycast es idéntico al de las direcciones multicast, pero en este caso la dirección de cada subred se encuentra reservada como dirección anycast a la que se la llama subred de router.

4.8.3. Índice de zona

Es común que un servidor por ejemplo cuente con más de un adaptador de red, y cada uno de los disponibles pertenecerá por lo general a distintos enlaces locales. En este caso, el prefijo es común al ser el mismo equipo, lo que hace que haya que indicar qué adaptador será el que se encargue de enviar y recibir el tráfico que cada enlace local del equipo. Esto se soluciona creando un identificador al que llamamos índice de zona. La manera de escribir dicho índice variará dependiendo del sistema operativo que se use:

- En **Windows** lo común es usar el símbolo "%" acompañado de un valor numérico.
- En **Linux** se suele identificar con la variable Alcance (scope).

4.8.4. Mecanismos de transición IPv4 a IPV6

Por mucho que las direcciones IPv4 ya se hayan agotado, se espera que el protocolo siga vigente durante muchos años más, entonces tendremos una coexistencia entre ambos protocolos. El principal motivo de esto es económico debido a que os ISP deben de cambiar los dispositivos de red ya que la mayoría solo funcionan por IPv4.

Aunque ambos protocolos son **incompatibles** entre ellos, hay varias técnicas que nos ayudarán a gestionar la interoperabilidad IPv4-IPv6 y son los llamados mecanismos de transición.

Los mecanismos de transición se clasifican en tres grupos:

- **Pila Dual (Dual Satack – Lite).**
- **Tunelización.**
- **Traducción de direcciones.**

Pila Dual

Este mecanismo abarca el soporte en un mismo dispositivo de la pila de protocolos IPv4 y de la de IPv6 de manera simultánea.

Esta herramienta nos ayuda a que cuando un nodo mande información se comporte con el mismo protocolo que usa el dispositivo de destino. Para que esto pueda suceder, ambos nodos deben de configurarse con direcciones IPv4 e IPv6.

Este mecanismo se encuentra presenta en cualquier sistema operativo actual, pero no en todos los dispositivos de internet.

Tunelización

Esta técnica consiste en encapsular los paquetes IPV6 dentro del campo de datos en los paquetes IPv4. Esto permite que se transmita un datagrama de IPV6 en una infraestructura IPv4 sin cambiar el mecanismo de enrutamiento que estuviera preestablecido.

Tenemos dos tipos de túneles:

- **Túneles manuales.** En este caso la encapsulación se hace directamente en el router y hace necesario entonces que los routers que se encuentren en los extremos tengan el mecanismo de Pila Dual implementado.

- **Túneles automáticos.** Se conectan los hosts que tienen implementado Pila Dual a través de un océano de IPv4. Los túneles más famosos son 6to4, ISATAP y Teredo, de hecho, este último pude incluso funcionar detrás de un router NAT.

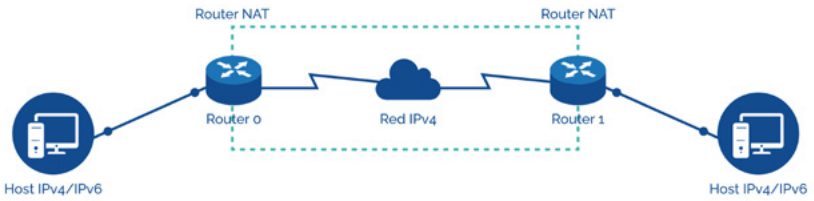

Imagen 37. Túnel Teredo

Traducción

Los mecanismos de transición o incluso las técnicas de traducción permiten un enrutamiento entre nodos solo de cada protocolo, es decir, si enrutamos IPv4 no se puede enrutar IPv6 y al revés exactamente igual.

Cuando se realiza la tunelización, por tanto, no hay encapsulación, sino una conversión del paquete IPv4 en IPv6 y viceversa.

Esta técnica puede operar en distintas capas:

Nivel de red.

- El método SIIT se usa para la definición de reglas para traducir las cabeceras. Además, define el rango de direcciones IPv6 que abarca las direcciones IPv4 que ya se han traducido y son de tipo ::ffff:9:a.b.c.d, siendo a.b.c.d la dirección IPv4 en su propio formato.
- El cuadro que tenemos a continuación nos muestra las reglas que ahora mismo hay para traducción de IPv6 e IPv4:

Traducción de cabeceras IPv6 e IPv4 mediante SIIT		
Campo	**IPv6**	**IPv4**
Version	6	4
HLEN (*Header length)*	N/A	5
ToS(*Type of Service)*	Xxxxxxx	Xxxxxxx
Total length	X	X + *IPv4 header*
Identification	N/A	0
Flags	N/A	MF = 0, DF = 1
Fragment offset	N/A	0
TTL (*Time to Live)*	X	X – 1
Protocol	*Next header* – X	X
Header checksum	-	Generated
Source address	0:ffff:0:a.b.c.d	a.b.c.d
Destination address	0::ffff:0:e.f.g.h	e.f.g.h

- **Nivel de transporte.** Un ejemplo sería SOCKS64.
- **Nivel de aplicación.** Un ejemplo sería ALG (Application Layer Gateway).

4.9. El nivel de transporte

4.9.1. Concepto

El 4º nivel del modelo OSI, el cual se sitúa por encima del nivel de red y por debajo del conjunto que hace referencia a sesión, presentación y aplicación es el nivel de transporte.

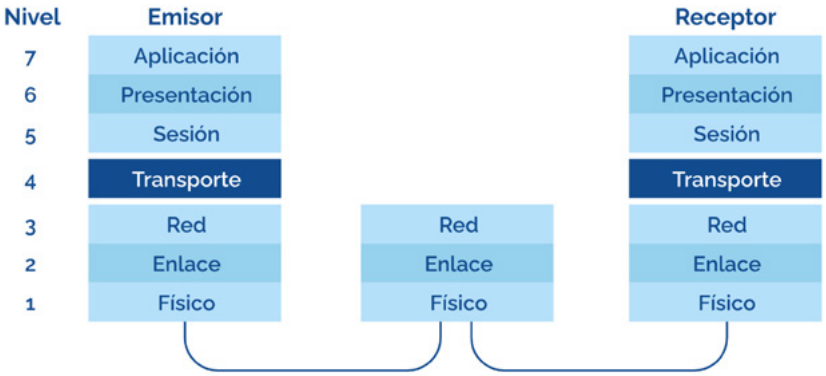

Imagen 38. Capa de transporte en equipo Emisor y Receptor

El nivel de aplicación le pasará el mensaje a este nivel en el emisor, que tendrá que dividirlo en segmentos y pasarle dichos segmentos al nivel de red, de modo que se asegure de que ha llegado de manera correcta al receptor.

En cambio, cuando el mensaje llegue al receptor, este se encargará de darle forma a todos los segmentos con el fin de conseguir el mensaje original y poder pasarlo sin ningún error al nivel superior, en este caso, el de aplicación.

Este nivel tiene lugar de manera exclusiva en el equipo terminal, diferenciando de los niveles anteriores, que sucedían en los dispositivos de interconexión. Es en el sistema operativo de cada equipo donde se implementará el protocolo de transporte correspondiente, lo que hace que se pueda considerar a este protocolo como una frontera entre los protocoles de la red y los protocolos que trabajan en las máquinas finales.

Los mensajes en este nivel son llamados TPDU, Transport Protocol Data Unit. También se pueden llamar segmentos, y se encuentran encapsulados en el campo de datos cuando nos referimos al nivel de red.

Imagen 39. Encapsulamiento del segmento TCP o UDP

4.9.2. Funciones del nivel de transporte

Direccionamiento en punto de servicio

Se entregan los paquetes siguiendo un proceso primero en el equipo emisor y después en el equipo receptor. Para esta tarea se usan los denominados puertos que veremos más adelante.

Gracias a esta función se pueden tener numerosas conexiones al mismo tiempo en un mismo equipo. Esto se denota por ejemplo en que podamos tener varias aplicaciones que usan internet al mismo tiempo abiertas.

Segmentación y ensamblado

Muchas veces el tamaño del mensaje que ha recibido es demasiado extenso como para poder almacenarlo en un datagrama de red, por lo que este nivel se encarga de dividirlo en diferentes segmentos.

En cada segmento se almacena un número de secuencia que hace que cuando llegan al receptor, este sepa como volver a ordenarlo de manera correcta. También lleva un fin de secuencia el último de los segmentos, para que cuando llegue a este el receptor sepa que el mensaje está completo.

Control de errores (segmentos dañados, perdidos o duplicados)

A nivel de transporte el control de errores se basa en detectar si hay algún segmento corrupto y comunicar esto. Normalmente para esta detección se usan algoritmos que se encuentran implementados por software como el CheckSum.

Como hemos visto antes, los segmentos llevan unos números de secuencia. Los cuales también sirven para reenviarlos o eliminarlos en casos de errores.

Control de flujo

Igual que en el nivel de enlace, usamos ventana deslizante, pero en este caso de extremo a extremo.

Control de la conexión

Este nivel permite que haya comunicaciones orientada a la conexión, en las que primero se establece la ruta a seguir de todos los segmentos que componen un mismo mensaje.

En última instancia se enviarán los datos y después la conexión se libera.

Multiplexación

Esta técnica consiste en la posibilidad de que dos o más comunicaciones se establezcan en un mismo medio con el fin de que la eficiencia de las conexiones cuando el tráfico no supera la capacidad total del enlace aumente.

4.10. Puerto y socket de comunicaciones

4.10.1. Concepto de puerto

Para hacer una distinción entre todos los procesos que ocurren en una red a la hora de enviar y recibir información, el nivel de transporte añade a las direcciones IP que se introdujeron en el nivel de red, los llamados puertos.

Un puerto se trata de un número de 16 bits que identifica a un proceso en la red.

Como tenemos que 2^{16} = 65 536, tenemos entonces que el rango de puertos que puede tener un equipo es desde 0 hasta 65 535.

Dependiendo de su utilidad, este rango de puertos se divide en tres clasificaciones:

- **Puertos bien conocidos** → asignados en el rango 0-1 023 por el ICANN.
  - Se trata de los puertos que más usa el sistema operativo y que suelen necesitar privilegios. Normalmente estos puertos son usados por las aplicaciones que trabajando como servicios y siempre están a la escucha de conexiones.
  - Ejemplos son: SSH (puerto 22), DNS (puerto 53), POP3 (puerto 110), LDAP (389), SMB (445), etc.
- **Puertos registrados** → se les asigna el rango 1 24-49 151.
  - Se encuentran en uso por las aplicaciones que deben de usarlos de manera temporal, es decir, que no se encuentran trabajando en todo momento. También poder que haya algún servicio que registrado externo al ICANN que los use.
  - Ejemplos son: NFS (puerto 2 049), Blizzard (usa varios), etc.
- **Puertos dinámicos o privados** → Se encuentran en el rango 49 152-65 535.
  - También se usan por aplicaciones de manera temporal, pero en este caso, estos no tienen significado alguno si no existe una conexión TCP que los use. Esto se traduce en que cada vez que se termine la conexión se liberan, por lo que ninguna aplicación o servicio tiene uno concreto asignado.

Clasificación de los puertos		
Tipo de puerto	**Rango**	**Utilidad**
Bien conocidos	0 al 1 023	Servicios de red registrados por el ICANN.
Registrados	1 024 al 49 151	Servicios de red registrados por otras entidades, Aplicaciones de usuario.
Dinámicos o privados	49 152 65 535	Aplicaciones de usuario.

4.10.2. Concepto de socket

Nos referimos a socket como la interfaz que hace posible la comunicación remota entre procesos.

Para identificar un socket se debe de identificar el siguiente modo: "dirección IP: puerto", esos deben de ir juntos, en pareja. Un ejemplo sería una conexión a SSH, de la IP 192.168.1.2, que sería 192.168.1.2:22.

A cada socket se le asigna un buffer, que es un espacio de memoria intermedia donde se almacenan los datos que se van a enviar o recibir.

Lógicamente a la hora de establecer una comunicación es necesario que haya un socket en cada uno de los extremos de la comunicación, es decir, local y remoto. Estos deben de estar trabajando conjuntamente y usar el mismo protocolo de transporte.

Si queremos ver las comunicaciones activas en un equipo se usa el comando `netstat`.

```
PS C:\Users\Administrador> netstat -n

Conexiones activas

  Proto  Dirección local        Dirección remota       Estado
  TCP    10.0.0.25:51662        10.0.0.24:53           SYN_SENT
  TCP    10.0.0.25:65390        208.67.222.222:53      SYN_SENT
  TCP    10.0.2.15:49671        20.54.122.82:443       ESTABLISHED
  TCP    10.0.2.15:51416        13.107.21.200:443      ESTABLISHED
  TCP    10.0.2.15:51417        13.107.21.200:443      ESTABLISHED
  TCP    10.0.2.15:51419        204.79.197.203:443     ESTABLISHED
  TCP    10.0.2.15:51420        23.200.66.154:443      ESTABLISHED
  TCP    10.0.2.15:56082        52.184.212.181:443     TIME_WAIT
  TCP    10.0.2.15:56084        8.247.215.254:80       ESTABLISHED
  TCP    10.0.2.15:56085        93.184.220.29:80       ESTABLISHED
  TCP    10.0.2.15:56086        23.8.28.136:443        ESTABLISHED
  TCP    10.0.2.15:56087        40.125.122.176:443     TIME_WAIT
  TCP    10.0.2.15:56088        40.125.122.176:443     TIME_WAIT
  TCP    10.0.2.15:56090        20.67.183.221:443      TIME_WAIT
  TCP    10.0.2.15:56092        104.208.16.90:443      TIME_WAIT
  TCP    10.0.2.15:56093        104.208.16.90:443      TIME_WAIT
  TCP    10.0.2.15:56094        23.203.161.130:443     ESTABLISHED
  TCP    10.0.2.15:56095        20.67.183.221:443      TIME_WAIT
  TCP    10.0.2.15:56605        52.142.114.176:443     ESTABLISHED
  TCP    10.0.2.15:56606        23.62.185.32:443       ESTABLISHED
  TCP    10.0.2.15:56607        23.62.185.32:443       ESTABLISHED
  TCP    10.0.2.15:56608        204.79.197.222:443     ESTABLISHED
  TCP    10.0.2.15:56609        93.184.220.29:80       ESTABLISHED
  TCP    10.0.2.15:56611        191.237.255.66:443     ESTABLISHED
  TCP    10.0.2.15:56613        13.107.246.254:443     ESTABLISHED
  TCP    10.0.2.15:56614        13.107.42.254:443      ESTABLISHED
  TCP    10.0.2.15:56615        20.42.73.25:443        TIME_WAIT
  TCP    10.0.2.15:56617        52.184.220.11:443      ESTABLISHED
  TCP    10.0.2.15:56618        23.8.26.58:443         ESTABLISHED
  TCP    10.0.2.15:56619        23.8.28.136:443        ESTABLISHED
  TCP    10.0.2.15:56620        67.26.103.254:80       ESTABLISHED
  TCP    10.0.2.15:56621        185.43.182.73:80       TIME_WAIT
  TCP    10.0.2.15:56622        185.43.182.50:80       TIME_WAIT
  TCP    10.0.2.15:56623        52.113.194.132:443     ESTABLISHED
  TCP    10.0.2.15:56624        40.126.31.9:443        ESTABLISHED
  TCP    10.0.2.15:56625        40.126.31.9:443        ESTABLISHED
PS C:\Users\Administrador>
```

Imagen 40. netstat -n

Podemos ver que por ejemplo la primera conexión usa el protocolo TCP, su socket local es 10.0.0.25:51 662, su socket remoto es 10.0.0.24:53.

4.11. Protocolo TCP (Transmission Control Protocol)

El protocolo de Control de transmisión, TCP, se encuentra definido en el estándar RFC 793 y se trata de uno de los protocolos más fiables y robustos funcionando sobre una red que no detecta ni corrige errores mediante el protocolo IP.

TCP es un protocolo orientado a la conexión, pero como también es tan fiable, su diseño es bastante complejo.

Los segmentos que realiza TCP cuando recibe los datos son de un máximo de 64 Kb y este se encarga de queden insertados en el área de datos de un datagrama de red.

4.11.1. Funcionamiento

Toda conexión TCP consta de tres etapas que vamos a describir a continuación: establecimiento de conexión, transferencia de datos y cierre de la conexión.

Establecimiento de la conexión

Para establecer una conexión TCP se usa un procedimiento llamado negociación en tres pasos que es más conocido por su nombre en inglés, Three-way handshake.

- **Paso 1:** el cliente abre un puerto de manera activa y envía un paquete SYN al servidor.
- **Paso 2:** lo siguiente que se hará será comprobar si por parte del servidor el puerto que debe de recibir la comunicación está abierto, que es lo mismo que decir si algún servicio está en escucha en ese puerto.
  - Si por cualquier razón este puerto no está abierto, se devuelve al cliente un paquete de respuesta que tiene el bit RST activado, significando esto que se ha rechazado la conexión. Si por otra parte el puerto sí que está activo, se mandará también un paquete de respuesta, pero con SYN-ACK, lo que significa que se ha reconocido la conexión.
- **Paso 3:** por último, el cliente responde de nuevo al servidor con un ACK, lo que da lugar a que haya terminado la comunicación en tres pasos.

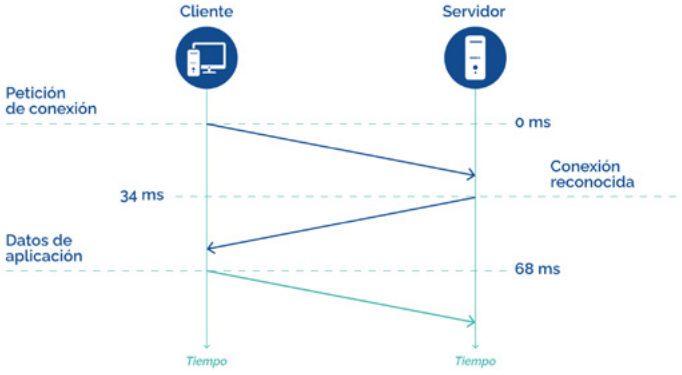

Imagen 41. Negociación en tres pasos

Transferencia de datos

Cuando se encuentra en la transferencia de datos, el protocolo TCP se encarga de usar una serie de herramientas que aseguren la fiabilidad y robustez del mismo protocolo. Algunas de estas herramientas son: el CheckSum para la detección de errores o el uso de las secuencias para ordenar los diferentes segmentos.

Fin de la conexión

Mientras que en la conexión se usaba una conexión en tres pasos, en el fin de la conexión se hace una "negociación en cuatro pasos" como se puede ver a continuación:

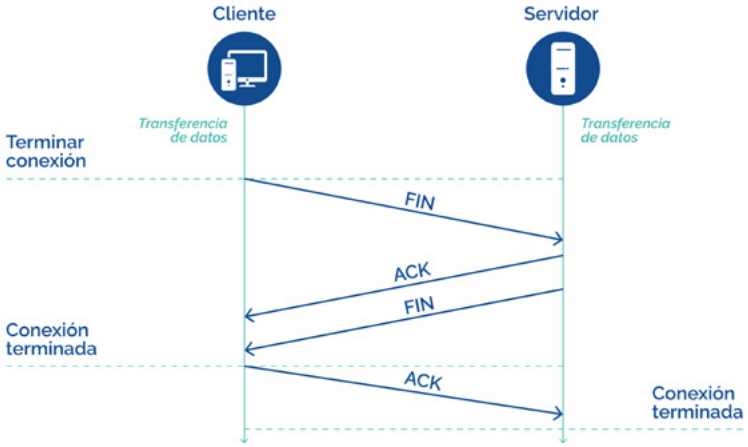

Imagen 42. Negociación en cuatro pasos

La fase final de la conexión usa la negociación en cuatro pasos como se puede ver más arriba, y la conexión se da por terminada en ambos extremos.

Cada uno de sus extremos querrá finalizar su conexión, para lo que transmitirá un paquete FIN que llegará al otro extremo como un ACK, es por esto por lo que una desconexión sin problemas requerirá de do segmentos para cada uno de estos paquetes, lo que da 4 pasos de negociación.

Puede que se dé el caso de que alguno de los lados finalice la conexión, pero el otro no, en este caso se considerará que la conexión está medio abierta. Un lado de la conexión (en el que sigue abierta), podrá enviar datos, pero el otro extremo no.

4.11.2. Estados de una conexión TCP

A continuación, se muestran los principales estados de una conexión TCP.

Estados de una conexión TCP		
Estado	**Lado**	**Significado**
LISTENING	Servidor	El servidor está a la espera de recibir peticiones SYN.
SYN_SENT	Cliente	El cliente ha enviado una petición SYN al servidor.
SYN_RCVD	Servidor	El servidor ha recibido una petición SYN desde un cliente.
ESTABLISHED	Ambos	Conexión establecida. El cliente ha recibido un reconocimiento SYN-ACK. El servidor ha recibido un reconocimiento ACK.
Cierre pasivo o remoto.		
CLOSE_WAIT	Ambos	FIN recibido y ACK enviado desde *ESTABLISHED*.
LAST_ACK	Ambos	FIN enviado y esperando ACK desde *CLOSE_WAIT*.
CLOSED	Ambos	Conexión terminada. ACK recibido desde *LAST_ACK*.
Cierre activo o local.		
FIN_WAIT_1	Ambos	FIN enviado y esperando ACK desde *ESTABLISHED*.
FIN_WAIT_2	Ambos	ACK recibido desde *FIN_WAIT_1*.
TIME_WAIT	Solo uno	FIN recibido y ACK enviado desde *FIN_WAIT_2*.
CLOSED	Ambos	Conexión terminada. *Timeout* = 2MSL desde *TIME_WAIT*.

Los primeros cuatro estados que vemos están directamente relacionados con el establecimiento de la conexión mediante el uso de la negociación en tres pasos.

Una vez que la conexión se ha establecido se procede con el intercambio de información entre ambos extremos y durante todo este proceso ambos sockets se encontrarán en estado established.

En último lugar se procederá con el cierre de la conexión, donde dependiendo de que sockets se cierre primero tenemos dos opciones:

- **Cierre pasivo.** En este cierre el socket remoto envía su paquete FIN con la intención de cerrar su mitad. Entonces el socket local pasará al estado close_wait mientras que el encargado de iniciar la conexión ejecute un close() que cierre el socket local.
- **Cierre activo.** En este cierre es el socket local el que procede con el cierre de la conexión, lo que pasa es que ahora pasa al estado fin_wait_1 y el socket remoto pasa al estado fin_wait_2 cuando confirma este cierre. Una vez que el socket remoto cierra del todo su parte el socket local pasa al estado time_wait y al poco tiempo se cierra de manera automática.

Este diagrama de transición de estados se encuentra recogido en el RFC793.

4.11.3. Formato de segmento TCP

Segmento TCP																															
OCTETO	0								1								2								3							
BIT	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	Puerto de origen																Puerto de destino															
32	Número de secuencia																															
64	Número de acuse de recibo (si ACK es establecido)																															
96	Longitud de Cabecera				Reservado				NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Tamaño de ventana														
128	Suma de verificación																	Puntero urgente (si URG es establecido)														
160 ...	Opciones (Si la Longitud de Cabecera > 5, relleno al final con "0" bytes si es necesario)																															

4.12. Protocolo UDP (User Datagram Protocol)

El protocolo UDP funciona a nivel de transporte y está basado en el intercambio de datagramas.

Características

- **No está orientado a la conexión.**

  Este protocolo permite que los datagramas se envíen a través de la red, pero sin la necesidad de establecer previamente una conexión porque es en el mismo datagrama donde hay la información necesaria para el direccionamiento. Dicha incorporación se encuentra en la cabecera.

- **No es fiable.**

  Como no hay conexión, tampoco hay confirmación de entrega o de rechazo, por lo que no se puede saber si los paquetes han llegado en buen estado o si ni siquiera han llegado. Tampoco tiene un control de flujo, lo que hace que también puedan llegar los paquetes desordenados causando un caos en la comunicación.

Utilidad

Este protocolo se usa principalmente para protocolos de nivel de aplicación como BOOTP o DHCP. En estos protocolos la información que se manda es tan poca que realmente no importa si el intercambio de paquetes se realiza de manera orientada o no a la conexión.

A la hora de transmitir audio y video a tiempo real, el conocido streaming, también se usa UDP porque es imposible que los paquetes que se envíen estén dañados o desordenados debido a los requisitos tan estrictos que se ponen para que el usuario pueda visualizar bien el contenido.

Formato

Segmento UDP		
Bits	**0 – 15**	**16 - 31**
0	Puerto origen	Puerto destino
32	Longitud del Mensaje	Suma de verificación
64	Datos	

4.13. El nivel de aplicación

4.13.1. Conceptos

El nivel de aplicación se trata del séptimo y último nivel del modelo OSI que hemos ido viendo a lo largo de esta unidad.

La función de dicho nivel es principalmente poder proporcionar al usuario servicios de múltiple carácter, los cuales pueden ser de cualquier tipo. No obstante, sí que hay una serie de estos servicios que son comunes y se pueden estandarizar.

Varias organizaciones internacionales como pueden ser la IUT o el W3C, se han encargado de que haya una regulación en dichos servicios de cara a su funcionamiento.

En el siguiente cuadro podemos observar los principales servicios y que protocolos se encargan de su funcionamiento:

Principales servicios de red y sus protocolos	
Servicio de red	**Protocolos**
Resolución de nombres de dominio	DNS
Configuración dinámica de host	APIPA, AVAHI, BOOTP, DHCP
Transferencia de ficheros	FTP, TFTP
Páginas web	HTTP, HTTPS
Correo	SMTP, POP3, IMAP4
Mensajería instantánea (Chat)	IRC, XMPP
Streaming de audio/vídeo	RTSP
Monitorización de redes	SNMP
Directorio	LDAP
Administración remota	Telnet, SSH, RDP

Es importante saber que los conceptos de servicio y protocolo son diferentes, pues servicios es la funcionalidad concreta que se le ofrece al usuario y en cambio, un protocolo es una implementación por software que se encarga específicamente de que esta función se pueda llevar a cabo.

En el nivel de aplicación **se interactúa con el usuario**, y es el único de todos los niveles en el que esto ocurre, por lo que por lo general los protocolos de este nivel cuentan con interfaces gráficas y no gráficas que hagan posible este intercambio de información.

4.14. Servicios de red

4.14.1. Asignación dinámica de direcciones (DHCP)

DHCP son las siglas de Dynamic Host Control Protocol (Protocolo de configuración dinámica de host) es un protocolo de red que trabaja en el nivel de aplicación y se encarga de asignar de manera automática las direcciones IP relativas a cada host, además de otros parámetros de red con la intención que de que el administrador no tenga que realizar tal tarea además de evitar colisiones de direcciones IP.

Entre sus numerosas ventajas destacan la siguientes:

- Ahorro de tiempo por parte del administrador.
- Mejor servicio al no depender de intervención externa.
- Cambios de red por parte de equipos sin necesidad de excesiva configuración.
- Elimina el uso de IP fijas en su mayoría, lo que es mejor para conexiones eventuales.
- Se ahorran direcciones IP.

De manera general, este servicio es muy usado en la intranet de cualquier red, porque, por ejemplo, si nosotros nos ponemos una conexión internet en casa, el router lleva implementado el DHCP y cada vez que nos conectemos estaremos cogiendo una IP que se nos ha asignado.

Vamos a ver como trabajaría un servidor DHCP de manera resumida:

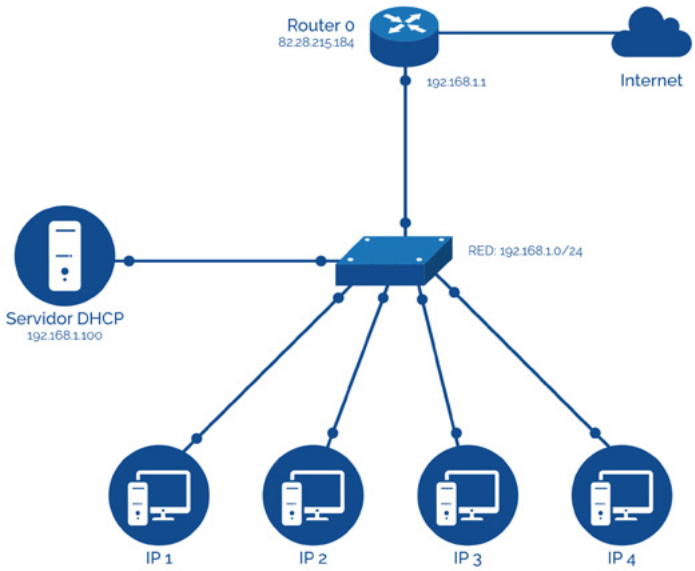

Imagen 43. Configuración de IP automática mediante DHCP

El diagrama anterior nos muestra como funcionaría el servidor DHCP, y sería del siguiente modo:

1. En primera instancia el router tendría una IP pública con la que se conectaría a internet y una IP privada dentro de la red, esta IP privada hará como default gateway de salida.
2. Después, tenemos un dispositivo, switch, hub… lo que sea, que hará de conmutador para establecer la red, también tendrá su propia IP.
3. Ahora, si el servidor de DHCP no lo tenemos en el router, sino a parte, se creará con una IP asignada fija y se establecerá el servicio.
4. En última instancia tenemos a los terminales finales que recibirán las IPs del servidor, del siguiente modo:
    - Generalmente las primeras 10 se reservarán, de modo que se empezaría a asignar a partir de la propia 10, porque recordemos que la dirección 192.168.1.0 se cuenta.
    - De ese modo, quedarían así si no pasa nada:
        - IP 1: 192.168.1.10
        - IP 2: 192.168.1.11
        - IP3: 192.168.1.12
        - IP4: 192.168.1.13

Para ejecutar todo esto será necesario que haya un administrador que mediante cierto software defina este tipo de direcciones, las IP reservadas, los rangos, y todo lo demás,

Las herramientas más usadas para esto son:

- El rol de Servidor DHCP en Windows Server.
- El paquete isc-dhcp-server en sistemas Linux.

La configuración de dichos servicios se verá con detenimiento en el próximo curso en el módulo de 'Servicios de red e internet'.

4.14.2. Resolución de nombres de dominio (DNS)

DNS son las siglas de Domain Name Service (Servidor de nombres de dominio), y es el protocolo encargado de asociar un nombre en concreto con una dirección IP con la intención de poder localizar un host sin necesidad de escribir su dirección IP completa.

Las funciones básicas que realiza el DNS son:

- Conversión de nombres de dominio a direcciones IP.
- Localización de los servidores de correo dentro de cada dominio.

La primera cuestión por la que es tan usado DNS es porque un nombre es mucho más fácil de interpretar y recordar para los usuarios que las direcciones IP, y, además, es posible que esta IP cambie, mientras que el nombre de dominio es improbable, lo que lo hace aún más fiable.

Antes del DNS esta información de dominios venía caracterizada por el uso del archivo hosts, donde se guardaban todas las relaciones entre nombres e IPS, pero daba lugar a muchos problemas naturales de cara al uso de un solo fichero. Es por esto por lo que surgió DNS como una gran base de datos donde almacenar estas relaciones.

El archivo hosts sigue estando, pero se usa mucho más reducidamente y para casos muy concretos.

Vamos a ver con el ejemplo anterior (usado para DHCP) como sería un servidor DNS:

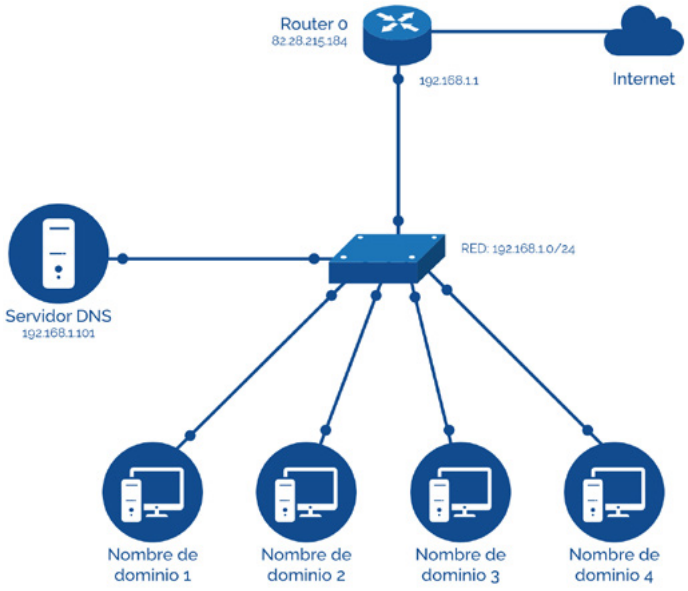

Imagen 44. Servidor DNS

Así, como podemos ver el servidor DNS tendrá una IP distinta, aunque no tendría por qué y, además, cada registro de DNS daría nombre a cada uno de los host.

Para configurar un servidor de DNS se usa principalmente:

- El rol de servidor de DNS en Windows Server.
- El paquete bind en los sistemas Linux.

4.14.3. Transferencia de archivos FTP

El siguiente protocolo por presentar es el protocolo FTP, File Transfer Protocol, o protocolo de transferencia de archivos. Este protocolo se encarga de la transferencia de grandes bloques de datos a través de una red.

Casi todos los contenidos alojados en páginas Web se suben al servidor usando este protocolo.

Los puertos que usa el protocolo FTP son:

- El puerto 20 que se usa para establecer el flujo de datos entre cliente y servidor.
- El puerto 21 que se usa para el flujo de control de los datos, es decir, el envío de instrucciones al servidor.

Esto quiere decir que funciona de la siguiente manera, mientras que los datos son transferidos a través del puerto 20, el flujo que trabaja en el puerto 21 está a la escucha.

Si por lo que sea tenemos el caso en el que el archivo a pasar a través de FTP es muy grande, el funcionamiento con doble puerto puede causar algún problema derivado de la interrupción de la sesión FTP.

Casi todos los sistemas operativos tienen un comando ftp que se puede usar para establecer conexiones:

```
root@debian:~# ftp
ftp>
```

Imagen 45. Comando ftp

Además, por ejemplo, desde un navegador web también se podría acceder a un servicio de FTP de manera gráfica.

Por último, las aplicaciones que se usan para configurar un servidor FTP principalmente son:

- El rol de servidor de archivos de Windows Server.
- El paquete tftp de sistemas Linux.

4.14.4. Páginas web. HTTP/HTTPS

HTTP son las siglas de Hyper Text Transfer Protocol y se trata del protocolo diseñado para la transferencia de las páginas web desde un servidor hasta un cliente, generalmente mediante un navegador web.

Una página web, como se verá en el módulo 'Lenguaje de Marcas y Sistemas de Gestión de la Información', suele estar compuesta por una serie de archivos HTML que acompañados de otros archivos CSS y JavaScript va a dar forma a una serie de textos e imágenes (generalmente), para mostrarnos el contenido con formato.

Si por ejemplo queremos ver de qué elementos se compone una página Web, se puede usar el inspeccionador de elementos del navegador Web, en este caso Google Chrome:

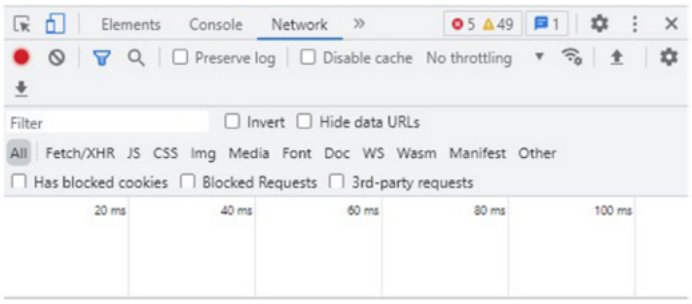

Imagen 46. Pestaña Network de inspeccionador de Google Chrome

Para realizar la acción anterior solo será necesario que hagamos clic derecho sobre la página en cuestión y seleccionemos la opción 'Inspeccionar'.

Los servidores Web que se encuentren en funcionamiento estarán siempre a la espera de recepción de peticiones http que enviará un cliente web a través por lo general de un navegador. Cada vez que le llegue la petición al servidor, si el recurso que se ha solicitado existe, enviará los archivos solicitados para su visualización por parte del cliente.

Realmente el servidor solo manda los archivos, porque es el cliente, o el navegador en cuestión quien se encarga de dar forma a los contenidos sirviéndose de los lenguajes de etiquetado o programación web que lleven los archivos para su visualización de manera correcta. Este proceso recibe el nombre de renderización.

Si, además, queremos que el protocolo que se use sea seguro, en vez de usar HTTP, se usará HTTPS, que cifra todos los datos intercambiados entre el cliente y el servidor.

Si usamos HTTPS para nuestras conexiones por el navegador web, en la barra de tareas aparecerá el protocolo mencionado además de un candando que nos da algo de información sobre la seguridad del sitio Web.

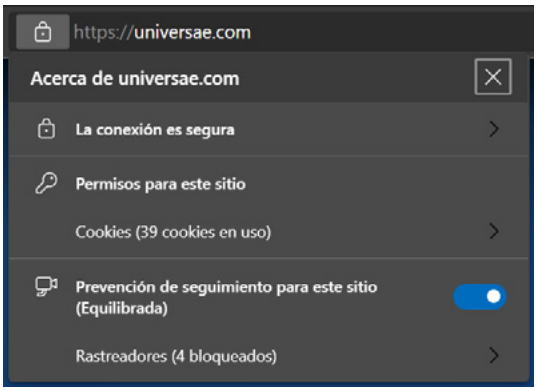

Imagen 47. Visibilidad de HTTPS en un navegador web

4.14.5. Correo. SMTP y POP3/IMAP4

Cuando hablamos de servicios de correo vamos a usar dos tipos distintos de protocolos: **SMTP y POP3/IMAP4.**

SMTP

El protocolo **SMTP** se usa para el **envío** de los mensajes por parte de un cliente a un servidor de correo directamente, el suyo propio, y para el envío de correo entre servidores.

Este protocolo se encuentra orientado a la conexión y se basa en texto, además, su protocolo de transporte es TCP.

POP3 e IMAP4

Distinto al anterior, son los protocolos **POP3 e IMAP4**, ya que estos se encargan de la **recepción** del correo por parte de un usuario final.

Estos dos protocolos son no orientados a la conexión, por lo que su protocolo de transporte es UDP.

POP3, funciona del siguiente modo:

- Un cliente de correo hará una solicitud.
- Si hay algo de correo pendiente el servidor lo enviará.
- Siguiente a esto, el servidor borrará los mensajes de modo que solo se encuentren disponibles los almacenados localmente.

IMAP4, funciona del siguiente modo:

- En este protocolo el correo no se descarga de manera automática en el cliente.
- Cuando el usuario decida visualizar el correo será cuando se descargue para que se pueda ver.
- Esto último hace que haya mensajes sin leer que se hayan borrado del servidor.

Como ya sabemos, hoy en día hay multitud de clientes de correo, el más famoso posiblemente sea Outlook o Gmail, de Microsoft y Google respectivamente.

Estos últimos ya se pueden visualizar desde cualquier dispositivo y eso es debido al almacenamiento de mensajes por parte del servidor, lo que también nos da la opción de crear carpetas en el propio servidor para organizar la información.

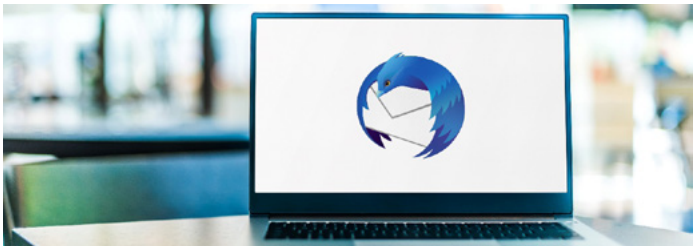

Imagen 48. Thunderbird, otro cliente de correo electrónico

4.14.6. Streaming. RTSP

El servicio de streaming usa el protocolo **RTSP**, que se encuentra no orientado a la conexión. Esto se puede permitir debido a que el servidor mantiene una sesión que se asocia un identificador para suplir la falta de conexión.

La mayoría de las sesiones RTSP usan el puerto 554/TCP para los datos de control del reproductor del contenido y UDP para el audio y el vídeo.

Se pueden abrir varias conexiones de transporte hasta el servidor y cerrarlas en una misma sesión RTSP.

Este protocolo tiene similitudes de sintaxis y operación con HTTP, pero se diferencia de este en:

- Un servidor RTSP necesita que el estado de la conexión se mantenga.
- Se pueden lanzar peticiones desde ambos lados de la conexión.

4.14.7. Monitorización de red. SNMP

En la capa de aplicación también encontramos el protocolo SNMP que facilitará que haya un intercambio de información entre dispositivos de red.

Como se trata de un protocolo de red que trabaja sobre la capa de aplicación, esto el permite que la monitorización de los dispositivos pueda ser muy amplía, abarcando a varios fabricantes en distintas redes y que usen plataformas distintas.

El protocolo SNMP se basa en dos conceptos, Gestor y Agente:

- **Gestor** es el equipo que ejecuta el cliente SNMP.
- **Agente** es un equipo donde el servidor de SNMP se ejecuta, por lo que se controla desde el gestor.

El funcionamiento es el siguiente:

- Periódicamente el gestor solicitará cierta información al agente además de ordenarle si es necesario que ejecute cualquier tarea.
- El agente advertirá al gestor si encuentra algo que le parece inusual en la red.

4.14.8. Directorio. LDAP

El servicio de directorio que usa el protocolo LDAP (este protocolo se verá en 'Implantación de Sistemas Operativos'), se trata del servicio de red que nos ayuda en la identificación de los recursos que hay en esta y otorga el acceso a estos para usuarios y aplicaciones.

La **estructura lógica** de un servicio de directorio se basa en:

- Dominios.
- Unidades Organizativas.
- Árboles.
- Bosques.

www.universae.com