

Ejercicio 1. Ponte el esquema del modelo OSI delante, repasa que hacía cada una de sus capas y trata de completar el siguiente texto con la capa correspondiente.

Un usuario está intentando acceder a una página "<https://www.lawebdemiempresa.com>" alojada en un servidor.

El protocolo de la capa de _____RED_____ usado en este caso es HTTP. Cuando abrimos el navegador y escribimos la url, HTTP se encarga de solicitar el archivo index de nuestra web a través de una petición o comando (paquete request) (asumiremos que la sesión TCP/IP ya ha sido iniciada).

Una vez generada la petición, el paquete request tiene que llegar a su destino. Aquí vamos a omitir las capas de _____PRESENTACION_____ y _____SESION_____, por lo que el paquete baja por las capas del modelo OSI hasta llegar a la capa de _____TRANSPORTE_____, en la cual se crea una cabecera TCP que se añade a la PDU. TCP es el protocolo de transporte que garantiza que los datos sean entregados en su destino sin errores y en el mismo orden en que se transmitieron, por lo que la cabecera TCP es una secuencia de números y otros datos destinados a cumplir con este propósito.

HTTP va encapsulado en TCP, encontraréis que en inglés dicen «rides on» o «sits on» debido al diseño top-down del modelo OSI. También se le suele llamar al protocolo TCP el «worker», y al protocolo IP el «workhorse» haciendo símil con el jinete y el caballo de batalla. TCP a su vez va encapsulado en IP, por lo que TCP «rides on» IP.

Una vez hecho su trabajo TCP, bajamos a la capa de _____RED_____ en la que IP se encarga de crear su cabecera la cual contiene información sobre la dirección lógica de envío y la añade al PDU. Una vez hecho esto envía el paquete a la capa de _____ENLACE DE DATOS_____ en la cual se crea una cabecera Ethernet, conteniendo ésta información sobre la dirección Ethernet física. Aquí ya tenemos el paquete completo y sólo queda enviarlo por el medio físico donde se transmite por medio de ceros y unos.

El paquete viajará por el cable de red llegando a nuestro servidor web, donde comenzará a leerlo e irá ascendiendo por las distintas capas. Primero leerá la cabecera de _____ENLACE DE DATOS_____ – Ethernet, en la que la tarjeta de red comprueba a través de su dirección física que el paquete ha llegado a su destino correcto. Una vez procesada esta información, la cabecera Ethernet es desechada y se procesa la cabecera IP. Aquí se comprueba que el paquete se ha recibido correctamente y que no ha sido fragmentado, una vez comprobado esto se desecha su cabecera y procede a leer la cabecera TCP. En esta cabecera se comprueba que el paquete ha llegado en la secuencia correcta, y una vez comprobado se desecha también para dejar sólo los datos de la capa de _____APLICACION_____ (nuestra petición), los cuales se envían directamente al servidor web que almacena nuestra web. En respuesta a esta petición del cliente, el servidor debería enviar un paquete TCP ACK (acknowledge – acuse de recibo) para que el cliente sepa que su petición ha sido recibida, seguido del archivo index.

Ejercicio 2: Dada la máquina con IP 18.120.16.250 y máscara de red 255.0.0.0, calcula:

- a) Clase de red a la que pertenece.

Como **18** está entre **1 y 126**, la IP **pertenece a la clase A**.

- b) Dirección de subred.

18.0.0.0

- c) Dirección de broadcast de la subred.

18.255.255.255

Ejercicio 3: Transforma a notación CIDR (notación abreviada) las siguientes direcciones y máscaras

Convertimos la máscara a binario.

- **Contamos los bits en 1** (estos indican la parte de la red).
- **El resultado es el número después del / en CIDR.**

1. tipo c 192.168.1.1-255.255.255.0 sería 192.168.1.1/24

2. tipo c 192.168.2.10-255.255.248.0

- Tiene **21 bits en 1** → 192.168.2.10/21

3. tipo a 10.0.0.2- 255.240.0.0

11111111 11110000

Tiene **12 bits en 1** → 10.0.0.2/12

4. tipo a 10.3.10.8-255.255.224.0

Tiene **19 bits en 1** → 10.3.10.8/19

5. tipo a 8.3.7.3-255.192.0.0

Tiene **10 bits en 1** → 8.3.7.3/10

6. tipo b 172.16.3.4-255.255.128.0

Tiene **17 bits en 1** → 172.16.3.4/17

7. tipo a 25.4.0.3-255.252.0.0

Tiene **14 bits en 1** → 25.4.0.3/14

8. tipo b 180.30.4.6-255.255.255.248

Tiene **29 bits en 1** → 180.30.4.6/29

9. tipo b 140.25.8.3-128.0.0.0

Tiene **1 bit en 1** → 140.25.8.3/1

10. tipo c 220.5.3.2-255.255.255.252

Tiene 30 bits en 1 → 220.5.3.2/30

Ejercicio 4: Indica la dirección de red y máscara (en notación CIDR) a las que pertenecen las siguientes direcciones

1. 192.168.1.1-255.255.255.0

192.168.1.0 (dirección de red).

La máscara tiene **24 bits en 1 → /24**.

Dirección de red: **192.168.1.0/24**

2. 192.168.2.10-255.255.248.0

192.168.0.0 (dirección de red)

192.168.0.0/21

3. 10.0.0.2-255.240.0.0

10.0.0.2

255.240.0.0

10.0.0.0/12

4. 10.3.10.8-255.255.224.0

10.3.10.8 - 255.255.224.0 → 10.3.0.0/19

ACLARACIONES TEÓRICAS SOBRE SUBNETING

Una **red** es un conjunto de dispositivos (como computadoras, routers, etc.) conectados entre sí. Cada dispositivo necesita una **dirección IP** única dentro de la red para poder comunicarse con otros dispositivos.

Las direcciones IP se dividen en **partes**: la parte que identifica la **red** y la parte que identifica el **dispositivo** dentro de esa red. Esto se conoce como la **dirección de red** y el **identificador de nodo**.

Red de tipo C:

Las **redes tipo C** son un tipo de red que tiene un rango de direcciones específicas. En este tipo de redes, **los tres primeros** son para identificar la **red** y **el último byte** es para identificar el **nodo o host**. Su máscara por defecto es **255.255.255.0**

¿Cómo se dividen las redes en subredes?

A veces necesitamos **dividir** una red en varias **subredes** para organizar mejor la red o para distribuir direcciones IP de manera más eficiente. Para hacer esto, tomamos parte de los **bits** que originalmente estaban destinados a los nodos y los usamos para **crear nuevas subredes**.

Ahora, vamos a explicar cómo **crear 16 subredes**.

Subredes:

Para crear **16 subredes**, necesitamos tomar una parte de los **bits del identificador de nodo**. En una dirección IP, el identificador del nodo está en el último byte (8 bits). Si queremos 16 subredes, necesitamos usar **4 bits** de ese byte, porque:

- Cada bit adicional en la máscara de red duplica el número de subredes disponibles.
- **2 elevado a 4** (es decir, $2 * 2 * 2 * 2$) da como resultado **16 subredes**.

Entonces, tomamos los **4 bits más significativos** (los que están más a la **izquierda**) del identificador de nodo para crear las subredes

Máscara de red:

La **máscara de red** es lo que nos ayuda a separar la parte de la **red** de la parte del **nodo**. Es como una especie de "filtro" que le dice a un dispositivo qué parte de la dirección IP corresponde a la red y qué parte corresponde al nodo.

En este caso, como tomamos 4 bits del identificador del nodo, la máscara de red es la siguiente:

- En **binario**: **11111111.11111111.11111111.11110000**
- Esto significa que los primeros 28 bits (los primeros tres bytes completos y los 4 bits del último byte) son para la **red**.
- Los **4 bits restantes** son para el **identificador de nodo** (es decir, para los dispositivos dentro de la red).

En decimal:

Cuando convertimos la máscara de red a formato decimal, obtenemos **255.255.255.240**.

Resumen:

- La **máscara de red** 255.255.255.240 (o 11111111.11111111.11111111.11110000 en binario) nos permite **dividir** una red tipo C en **16 subredes**.
- Al tomar los **4 bits más a la izquierda** del identificador del nodo, creamos esas subredes adicionales.

De esta forma, aunque originalmente solo teníamos una red con direcciones para muchos dispositivos, ahora podemos tener **16 subredes** más pequeñas, cada una con

un conjunto de dispositivos identificados por esos 4 bits adicionales.

Ejercicio 5: Actividad Guiada de Subneting. Si tenemos la red tipo C 194.168.100.0:

- a) ¿Qué máscara hay que aplicarle para dividirla en 16 subredes?
- b) ¿Cuántos nodos o host podrán contener cada una de esas 16 subredes?
- c) ¿Cuáles serían los nombres de las subredes?
- d) ¿Cuál sería la dirección IP del nodo con identificador 4 de cada una de estas subredes?
- e) ¿A qué subred pertenece el nodo cuya dirección IP es la 194.168.100.107?

A continuación, tienes la solución.

Solución:

Apartado a: Sabemos que las redes tipo C utilizan los tres bytes más significativos para la dirección de la red y el menor para el identificador del nodo. Para dividir una red en subredes se toman los bits más significativos del identificador del nodo. Como queremos formar 16 subredes tendremos que tomar 4 bits (**2 elevado a 4**) del identificador del nodo, concretamente los 4 bits más significativos (más a la izquierda). Por eso la máscara de red la mostramos de la siguiente manera:

11111111.11111111.11111111.11110000

identificador de red Identificador de nodo

En decimal **255.255.255.240**

Apartado b: Si miramos la máscara de red que hemos obtenido vemos que nos quedan 4 bits para identificar los nodos. (los 4 a 0). Por lo tanto, serían 16 hosts. Pero también sabemos que los 4 a uno son la dirección de broadcast y los 4 a 0 son el identificador de la subred. Por lo tanto debemos quitar estas dos y tenemos 14 nodos para cada una de ellas.

Apartado c: Los nombres de las subredes se obtienen dando a los 4 bits que hemos tomado para formar el identificador de subred todos los posibles valores, dejando los bits de identificador de nodo a cero. Estas 16 subredes serían:

Subred		Subredes
11000010.10101000.01100100	0000 0000	194.168.100. 0
	0001	16
	0010	32
	0011	48
	0100	64
	0101	80
	0110	96

0111	112
1000	128
1001	144
1010	160
1011	176
1100	192
1101	208
1110	224
1111	240

Apartado d: Para obtener la dirección IP del nodo con identificador 4 de cada una de estas subredes vamos a dar el valor 4 (0100 en binario) a los 4 bits que corresponden al nodo en cada una de las subredes. Por tanto las direcciones IP serían:

Subred		IP del nodo 4
11000010.10101000.01100100	0000 0100	194.168.100. 4
	0001	20
	0010	36
	0011	52
	0100	68
	0101	84
	0110	100
	0111	116
	1000	132
	1001	148
	1010	164
	1011	180
	1100	196
	1101	212
	1110	228
	1111	244

ACLARACIÓN TEÓRICA

Cada subred tiene **14 hosts**, con direcciones de host que van desde el **primer host** (la primera dirección disponible en la subred) hasta el **último host** (la última dirección antes de la dirección de difusión).

La dirección IP del **nodo 4** en cada subred será la **cuarta dirección disponible**. En cada subred, la primera dirección es la dirección de red, y la última es la dirección de difusión, por lo que las direcciones de hosts comienzan en la segunda dirección. Entonces:

- Para la subred **194.168.100.0**, el nodo 4 tiene la IP **194.168.100.4**.
- Para la subred **194.168.100.16**, el nodo 4 tiene la IP **194.168.100.20**.
- Para la subred **194.168.100.32**, el nodo 4 tiene la IP **194.168.100.36**.
- Para la subred **194.168.100.48**, el nodo 4 tiene la IP **194.168.100.52**.
- Y así sucesivamente para las demás subredes.

1 1 1 1 1 1 1 1.1 1 1 1 1 1 1.1 1 1 1 1 1.1 1 1 1 0 0 0 0 Máscara de red 255.255.255.240

AND

1 1 0 0 0 0 1 0 . 1 0 1 0 1 0 0 0 . 0 1 1 0 0 1 0 0 . 0 1 1 0 1 0 1 1 Dirección IP 194.168.100.107

1 1 0 0 0 0 1 0 . 1 0 1 0 1 0 0 0 . 0 1 1 0 0 1 0 0 . 0 1 1 0 0 0 0 0 Subred 194.168.100.96

Ahora intenta resolver tú el siguiente ejercicio:

Una organización ha conseguido un rango de direcciones IP válido de clase C: 200.54.2.x. Se quiere particionar en 4 subredes de más de 30 equipos cada una. Indicar cuál sería el particionamiento correcto y proponer para cada subred, rango de direcciones, máscara de subred, indicar dirección de subred y de broadcast de cada una y el nº de equipos que cada subred puede albergar.

Subred	Dirección de subred	Dirección de broadcast	Rango de direcciones	Máscara	Equipos disponibles
Subred 1	200.54.2.0	200.54.2.63	200.54.2.1 - 200.54.2.62	/26	62
2	200.54.2.64	200.54.2.127	200.54.2.65 - 200.54.2.126	/26	62
3	200.54.2.128	200.54.2.191	200.54.2.129 - 200.54.2.190	/26	62
4	200.54.2.192	200.54.2.255	200.54.2.193 - 200.54.2.254	/26	62