

# Práctica 7 - Redes y Comunicaciones

## Introducción

1. **¿Qué servicios presta la capa de red? ¿Cuál es la PDU en esta capa? ¿Qué dispositivo es considerado sólo de la capa de red?**
  - La función principal de la capa de red es engañosamente simple: **transporta paquetes desde un host emisor a un host receptor**. En la realización de esta tarea podemos identificar dos importantes funciones de la capa de red: **Reenvío (forwarding) y Enrutamiento (routing)**.  
El PDU de esta capa es el Paquete/Datagrama IP.  
El dispositivo principal de esta capa es el router.
2. **¿Por qué se lo considera un protocolo de mejor esfuerzo?**
  - Se lo considera protocolo best-effort ya que hace todo lo posible para entregar los datos, pero no garantiza que todos los paquetes llegarán al destino, ni de que lo harán en el orden correcto.
3. **¿Cuántas redes clase A, B y C hay? ¿Cuántos hosts como máximo pueden tener cada una?**

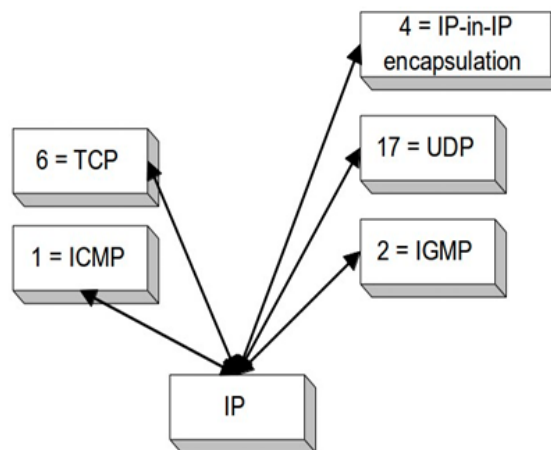
Clase	Cantidad de Redes	Cantidad de host
A	$2^7 = 128$	$2^{24} - 2 = 16777214$
B	$2^{14} = 16384$	$2^{16} - 2 = 65534$
C	$2^{21} = 2097152$	$2^8 - 2 = 254$

4. **¿Qué son las subredes? ¿Por qué es importante siempre especificar la máscara de subred asociada?**
  - Los prefijos de longitud fija por clase provocan un uso ineficiente en el espacio de direcciones y muchos equipos, produce escasez de direcciones.  
Esto supone la aparición de las subredes, que básicamente permite que haya subgrupos en las redes, se utiliza para generar redes dentro de la red. Para ello toma una parte del hostid.  
La división en subredes plantea que si una red de clase desperdicia muchas direcciones IP entonces la misma sea dividida en N subredes más pequeñas que aprovechen mejor el espacio de direccionamiento.

Las máscaras se utilizan para saber en una dirección IP qué bits son de red y qué bits son de host.

5. ¿Cuál es la finalidad del campo Protocol en la cabecera IP? ¿A qué campos de la capa de transporte se asemeja en su funcionalidad?

- **Protocolo:** Este campo solo se suele emplear cuando un datagrama IP alcanza su destino final. El valor de este campo indica el protocolo específico de la capa de transporte al que se pasarán los datos contenidos en ese datagrama IP y es utilizado para mux/demux. Por ejemplo, un valor de 6 indica que los datos se pasan a TCP, mientras que un valor igual a 17 indica que los datos se pasan a UDP. Se asemeja la funcionalidad al campo que indica el puerto destino en las cabeceras de los protocolos de capa de transporte.



## División en subredes

6. Para cada una de las siguientes direcciones IP (172.16.58.223/26, 163.10.5.49/27, 128.10.1.0/23, 10.1.0.0/24, 8.40.11.179/12) determine:

a. ¿De qué clase de red es la dirección dada (Clase A, B o C)?

- 172.16.58.223/26 → Clase B.
- 163.10.5.49/27 → Clase B.
- 128.10.1.0/23 → Clase B.
- 10.1.0.0/24 → Clase A.
- 8.40.11.179/12 → Clase A.

b. ¿Cuál es la dirección de subred?

```

10101100 . 00010000 . 00111010 . 11011111 172.16.58.223
11111111 . 11111111 . 11111111 . 11000000 /26
----- AND
10101100 . 00010000 . 00111010 . 11000000 172.16.58.192

10100011 . 00001010 . 00000101 . 00110001 163.10.5.49
11111111 . 11111111 . 11111111 . 11100000 /27
----- AND
10100011 . 00001010 . 00000101 . 00100000 163.10.5.32

10000000 . 00001010 . 00000001 . 00000000 128.10.1.0
11111111 . 11111111 . 11111110 . 00000000 /23
----- AND
10000000 . 00001010 . 00000000 . 00000000 128.10.0.0

00001010 . 00000001 . 00000000 . 00000000 10.1.0.0
11111111 . 11111111 . 11111111 . 00000000 /24
----- AND
00001010 . 00000001 . 00000000 . 00000000 10.1.0.0

00001000 . 00101000 . 00001011 . 10110011 8.40.11.179
11111111 . 11110000 . 00000000 . 00000000 /12
----- AND
00001000 . 00100000 . 00000000 . 00000000 8.32.0.0

```

c. ¿Cuál es la cantidad máxima de hosts que pueden estar en esa subred?

- $172.16.58.192 \rightarrow 2^6 - 2 = 62 \text{ hosts útiles}$   
 $163.10.5.32 \rightarrow 2^5 - 2 = 30 \text{ hosts útiles}$   
 $128.10.0.0 \rightarrow 2^9 - 2 = 510 \text{ hosts útiles}$   
 $10.1.0.0 \rightarrow 2^8 - 2 = 254 \text{ hosts útiles}$   
 $8.32.0.0 \rightarrow 2^{20} - 2 = 1048574 \text{ hosts útiles}$

d. ¿Cuál es la dirección de broadcast de esa subred?

- $172.16.58.192 \rightarrow 172.16.58.255$   
 $163.10.5.32 \rightarrow 163.10.5.63$   
 $128.10.0.0 \rightarrow 128.10.1.255$   
 $10.1.0.0 \rightarrow 10.1.0.255$   
 $8.32.0.0 \rightarrow 8.47.255.255$

e. ¿Cuál es el rango de direcciones IP válidas dentro de la subred?

- 172.16.58.192 → **Rango válido:** 172.16.58.193 ... 172.16.58.254
- 163.10.5.32 → **Rango válido:** 163.10.5.33 ... 163.10.5.62
- 128.10.0.0 → **Rango válido:** 128.10.0.1 ... 128.10.1.254
- 10.1.0.0 → **Rango válido:** 10.1.0.1 ... 10.1.0.254
- 8.32.0.0 → **Rango válido:** 8.32.0.1 ... 8.47.255.254

**7. Su organización cuenta con la dirección 128.50.10.0. Indique:**

**a. ¿Es una dirección de red o de host?**

- Es una dirección de host ya que pertenece a la clase B, si fuera una dirección de red de la clase B solo tendría los 2 primeros octetos con valor, al tener el tercer octeto con valor delata que es de host.

**b. Clase a la que pertenece y máscara de clase.**

- Es de clase B y la máscara default es 255.255.0.0.

**c. Cantidad de hosts posibles.**

- $2^{16} - 2 = 65534$  *hosts posibles.*

**d. Se necesitan crear, al menos, 513 subredes. Indique:**

**i. Máscara necesaria.**

- $2^h \geq 513 \rightarrow h = \log_2(513) \rightarrow 10$   
La máscara tiene que ser /26 → 255.255.255.192.

**ii. Cantidad de redes asignables.**

- La máscara original era /16 y la nueva es /26, eso quiere decir que tomamos prestados 10 bits originales del host, por lo tanto tenemos  $2^{10} = 1024$  *redes asignables.*

**iii. Cantidad de hosts por subred.**

- Tenemos 6 bits de host disponible, por lo tanto tenemos →  $2^6 - 2 = 62$  *hosts útiles por subred*

**iv. Dirección de la subred 710.**

- **Pasos:**

1. **Calcular tamaño de la subred** → El tamaño depende de la máscara, en este caso al tener máscara /26 sabemos que hay 64 hosts por subred, por lo tanto, el tamaño de la misma es 64.
2. **Calcular representación binaria de la dirección IP inicial** → A nosotros nos brindan una dirección de host específica que es 128.50.10.0, pero inicialmente las direcciones arrancan en 0, por lo tanto, la dirección IP inicial es 128.50.0.0 y la representación binaria de la misma es 10000000 . 00110010 . 00000000 . 00000000.
3. **Calcular representación binaria de la Máscara inicial** → Inicialmente tenemos una máscara /16 por lo tanto, en binario esta máscara sería 11111111 . 11111111 . 00000000 . 00000000.
4. **Calcular representación binaria de la Máscara de subred** → Sabemos que la máscara de subred es /26 por lo tanto, en binario esta máscara sería 11111111 . 11111111 . 11111111 . 11000000.
5. **Calcular qué bits se tomaron para ubicar las subredes** → Para la máscara inicial se usaban 2 octetos para red y 2 octetos para host, con la nueva máscara de subred se toman 2 octetos para red, 1 octeto y 2 bits para subred y 6 bits para host, por lo tanto los bits de subred son → 11111111 . 11111111 . 11111111 . 11000000.
6. **Calcular el número de subred requerido** → Se nos pide obtener la dirección de la subred 710 pero como se empieza a contar desde 0 tenemos que restar 1, por lo tanto, se nos está pidiendo calcular la subred 709.
7. **Calcular representación binaria de la subred requerida y dividir en octetos contando de izquierda a derecha** → 709 en binario es 10110001 . 01.
8. **Obtener la dirección** → Sabiendo qué bits se utilizan dentro de la máscara para el número de subred lo que tenemos que hacer es calcular un **OR** entre la **dirección IP inicial** y la **representación binaria de la subred requerida** ubicando los bits de la subred requerida en los bits específicos de la máscara que son para la subred, los bits restantes se completan con 0:

10000000 . 00110010 . 00000000 . 00000000	128.50.0.0
00000000 . 00000000 . 10110001 . 01000000	709
-----	OR
10000000 . 00110010 . 10110001 . 01000000	128.50.177.64

**v. Dirección de broadcast de la subred 710.**

- La dirección de broadcast de la subred 710 es 10000000 . 00110010 . 10110001 . 01111111 → 128.50.177.127.

**8. Si usted estuviese a cargo de la administración del bloque IP 195.200.45.0/24**

**a. ¿Qué máscara utilizaría si necesita definir al menos 9 subredes?**

- $2^h \geq 9 \rightarrow h = \log_2(9) \rightarrow h = 4$   
Necesitamos una máscara /28 → 11111111 . 11111111 . 11111111 . 11110000.

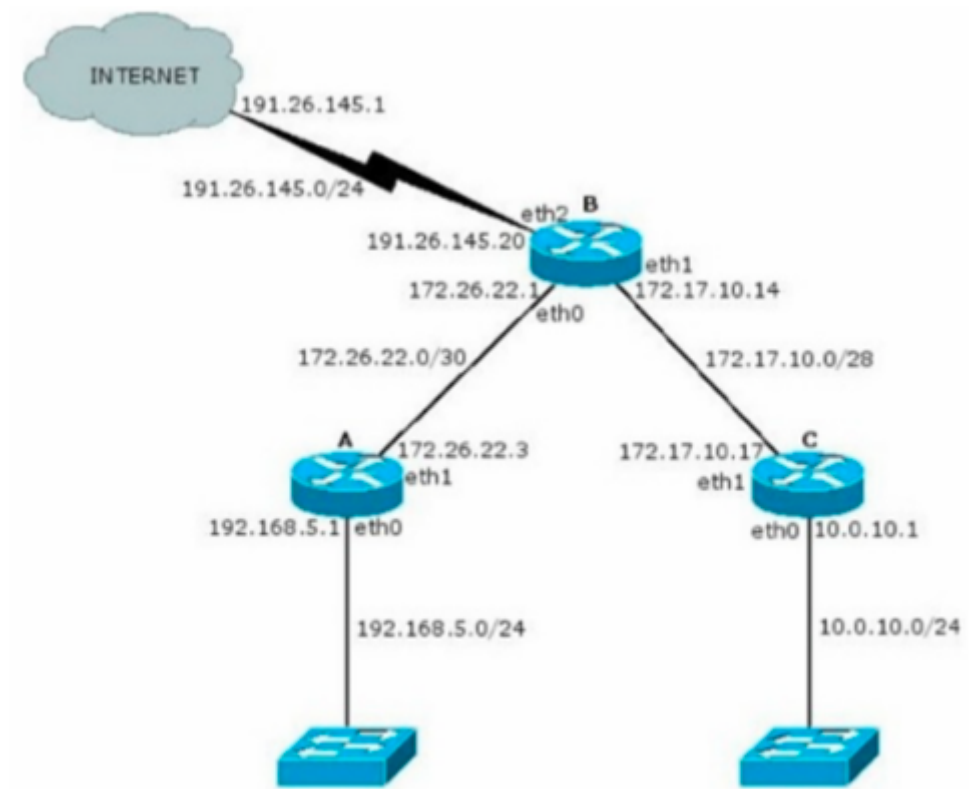
**b. Indique la dirección de subred de las primeras 9 subredes.**

- 11000011 . 11001000 . 00101101 . 00000000 → **195.200.45.0**  
11000011 . 11001000 . 00101101 . 00010000 → **195.200.45.16**  
11000011 . 11001000 . 00101101 . 00100000 → **195.200.45.32**  
11000011 . 11001000 . 00101101 . 00110000 → **195.200.45.48**  
11000011 . 11001000 . 00101101 . 01000000 → **195.200.45.64**  
11000011 . 11001000 . 00101101 . 01010000 → **195.200.45.80**  
11000011 . 11001000 . 00101101 . 01100000 → **195.200.45.96**  
11000011 . 11001000 . 00101101 . 01110000 → **195.200.45.112**  
11000011 . 11001000 . 00101101 . 10000000 → **195.200.45.128**

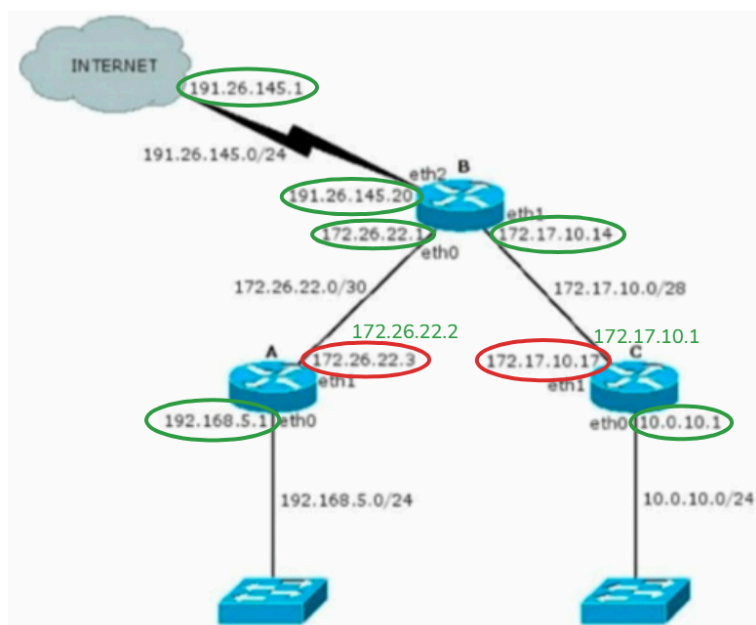
**c. Seleccione una e indique dirección de broadcast y rango de direcciones asignables en esa subred.**

- **Para la subred 195.200.45.0**
  - **Broadcast** → 195.200.45.15.
  - **Rango válido** → 195.200.45.01 ... 195.200.45.14.

**9. Dado el siguiente gráfico:**



- a. Verifique si es correcta la asignación de direcciones IP y, en caso de no serlo, modifique la misma para que lo sea.



○ Correcta  
○ Incorrecta - Solución

- b. ¿Cuántos bits se tomaron para hacer subredes en la red 10.0.10.0/24?  
¿Cuántas subredes se podrían generar?

- **Máscara Inicial para 10.0.10.0** → Al ser una IP de clase A la máscara inicial que se tomó es /8 255.0.0.0.  
**Nueva máscara aplicada** → Se aplicó una máscara /24, por lo tanto, se tomaron **16 bits para hacer subredes** =  $2^{16} = 65536$  subredes disponibles.
- c. **Para cada una de las redes utilizadas indique si son públicas o privadas.**
  - **10.0.10.0** → Privada, está comprendida en el rango: 10.0.0.0 ... 10.255.255.255.
  - **192.168.5.0** → Privada, está comprendida en el rango: 192.168.0.0 ... 192.168.255.255.
  - **172.17.10.0** → Privada, está comprendida en el rango: 172.16.0.0 ... 172.31.255.255.
  - **172.26.22.0** → Privada, está comprendida en el rango: 172.16.0.0 ... 172.31.255.255.
  - **191.26.145.0** → Pública.

## CIDR

### 10. ¿Qué es CIDR (Class Interdomain routing)? ¿Por qué resulta útil?

- **CIDR (Classless Inter-Domain Routing)** es una metodología de direccionamiento IP que permite una asignación más eficiente de direcciones IP, superando las limitaciones del sistema de clases tradicional (Clase A, B y C). A su vez, este mecanismo es una estrategia para **frenar ciertos problemas** que se manifestaron con el crecimiento de Internet, estos son:
  - Las clases A y B el 50% asignadas, clases C solo el 2%.
  - Las clases C:  $2^{21}$  redes aumentarían las tablas de ruteo notablemente.
  - Crecimiento de 1988 a 2000 de tablas de ruteo.
- **Optimiza la gestión del espacio de direcciones en Internet y reduce el tamaño de las tablas de enrutamiento mediante el agrupamiento.**
- **Elimina las clases** para no limitar las redes a tamaños específicos. En su lugar, utiliza una notación de prefijo (por ejemplo, /24) para indicar la máscara de subred.
- Consiste básicamente en permitir máscaras de subred de longitud variable (VLSM) para optimizar la asignación de direcciones IP y utilizar resumen de rutas para disminuir el tamaño de las tablas de enrutamiento.

### 11. ¿Cómo publicaría un router las siguientes redes si se aplica CIDR?

- a. 198.10.1.0/24
- b. 198.10.0.0/24
- c. 198.10.3.0/24
- d. 198.10.2.0/24



198.10.0.0 = 11000110 . 00001010 . 00000000 . 00000000  
198.10.1.0 = 11000110 . 00001010 . 00000001 . 00000000  
198.10.2.0 = 11000110 . 00001010 . 00000010 . 00000000  
198.10.3.0 = 11000110 . 00001010 . 00000011 . 00000000  
MÁSCARA = 11111111 . 11111111 . 11111100 . 00000000 255.255.252.0  
NÚMERO CIDR = 198.10.0.0/22

**12. Listar las redes involucradas en los siguientes bloques CIDR:**

- 200.56.168.0/21
- 195.24.0.0/13
- 195.24/13

200.56.168.0/21 → 11001000 . 00111000 . 10101000 . 00000000  
11001000 . 00111000 . 10101111 . 00000000 = 200.56.175.0/24  
RANGO DISPONIBLE → 200.56.168.0/24 ... 200.56.175.0/24

195.24.0.0/13 → 11000011 . 00011000 . 00000000 . 00000000  
11000011 . 00011111 . 11111111 . 00000000 = 195.31.255.0/24  
RANGO DISPONIBLE → 195.24.0.0/24 ... 195.31.255.0/24

195.24/13 → Es lo mismo que el de arriba.

**13. El bloque CIDR 128.0.0.0/2 o 128/2, ¿Equivale a listar todas las direcciones de red de clase B? ¿Cuál sería el bloque CIDR que agrupa todas las redes de clase A?**

128.0.0.0/2 → 10000000 . 00000000 . 00000000 . 00000000  
10111111 . 11111111 . 00000000 . 00000000 → 191.255.0.0  
DIRECCIONES DE RED DISPONIBLES →  $2^{14} = 16384$  → Equivale a la cantidad de redes de Clase B.  
BLOQUE CIDR PARA CLASE A → 0.0.0.0/1

## VLSM

**14. ¿Qué es y para qué se usa VLSM?**

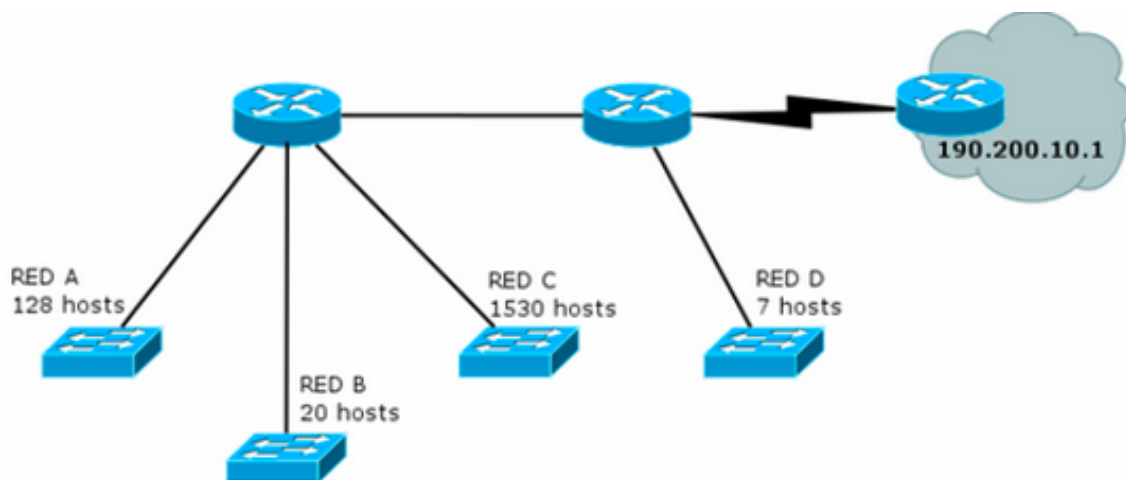
- En el **subnetting variable**, se utilizan máscaras de subred de **longitud variable** para crear subredes de diferentes tamaños, adaptadas a la cantidad de hosts en cada segmento de la red. Esto permite optimizar el uso del espacio de direcciones IP.
- Evita que haya un desperdicio de direcciones.

**15. Describa, con sus palabras, el mecanismo para dividir subredes utilizando VLSM.**

- **Mecanismo general de VLSM:**

1. Calcular la máscara para la/s subred/es de mayor cantidad de hosts.
2. De las subredes que obtenemos, asignamos todas las que se puedan con el menor desperdicio posible.
3. Si quedan segmentos de red sin una subred asignada volver a hacer el paso 1.

**16. Suponga que trabaja en una organización que tiene la red que se ve en el gráfico y debe armar el direccionamiento para la misma, minimizando el desperdicio de direcciones IP. Dicha organización posee la red 205.10.192.0/19, que es la que usted deberá utilizar.**



- a. ¿Es posible asignar las subredes correspondientes a la topología utilizando subnetting sin VLSM? Indique la cantidad de hosts que se desperdicia en cada subred.**

RED DE LA ORGANIZACIÓN → 205.10.192.0/19

CONVERSIÓN A BINARIO = 11001101 . 00001010 . 11000000 . 00000000

La red es de clase C y tiene máscara por default /24, por lo tanto, tiene  $2^8 - 2$  hosts disponibles → 254.

RED C → 1530 → No puede ser representada en subnetting fijo.

RED A → 128 → Si puede ser representada en subnetting fijo, se desperdician 126 hosts.

RED B → 20 → Si puede ser representada en subnetting fijo, se desperdician 234 hosts.

RED D → 7 → Si puede ser representada en subnetting fijo, se desperdician 247 hosts.

RED PEER-TO-PEER → 2 → Si puede ser representada en subnetting fijo, se desperdician 252 hosts.

- b. Asigne direcciones a todas las redes de la topología. Tome siempre en cada paso la primera dirección de red posible.**

- c. Para mantener el orden y el inventario de direcciones disponibles, haga un listado de todas las direcciones libres que le quedaron, agrupándolas utilizando CIDR.**
- d. Asigne direcciones IP a todas las interfaces de la topología que sea posible.**

RED DE LA ORGANIZACIÓN → 205.10.192.0/19

MÁSCARA ORIGINAL = 11111111 . 11111111 . 11100000 . 00000000 /19

CONVERSIÓN A BINARIO = 11001101 . 00001010 . 11000000 . 00000000

MÁSCARA DE RED C = 11111111 . 11111111 . 11111000 . 00000000 /21

DIRECCIÓN RED C = 11001101 . 00001010 . 11000000 . 00000000 205.10.192.0/21

PRÓXIMO A DIV. = 11001101 . 00001010 . 11001000 . 00000000

DIRECCIONES LIB. = 11001101 . 00001010 . 11010000 . 00000000

11001101 . 00001010 . 11011000 . 00000000

NÚMERO CIDR = 11001101 . 00001010 . 11010000 . 00000000 205.10.208.0/20

MÁSCARA DE RED A = 11111111 . 11111111 . 11111111 . 00000000 /24

DIRECCIÓN RED A = 11001101 . 00001010 . 11001000 . 00000000 205.10.200.0/24

PRÓXIMO A DIV. = 11001101 . 00001010 . 11001001 . 00000000

DIRECCIONES LIB. = 11001101 . 00001010 . 11001010 . 00000000

11001101 . 00001010 . 11001011 . 00000000

11001101 . 00001010 . 11001100 . 00000000

11001101 . 00001010 . 11001101 . 00000000

11001101 . 00001010 . 11001110 . 00000000

11001101 . 00001010 . 11001111 . 00000000

NÚMERO CIDR = 11001101 . 00001010 . 11001000 . 00000000 205.10.200.0/21

MÁSCARA DE RED B = 11111111 . 11111111 . 11111111 . 11100000 /27

DIRECCIÓN RED B = 11001101 . 00001010 . 11001001 . 00000000 205.10.201.0/27

PRÓXIMO A DIV. = 11001101 . 00001010 . 11001001 . 00100000

DIRECCIONES LIB. = 11001101 . 00001010 . 11001001 . 01000000

11001101 . 00001010 . 11001001 . 01100000

11001101 . 00001010 . 11001001 . 10000000

11001101 . 00001010 . 11001001 . 10100000

11001101 . 00001010 . 11001001 . 11000000

11001101 . 00001010 . 11001001 . 11100000

NÚMERO CIDR = 11001101 . 00001010 . 11001001 . 00000000 205.10.201.0/24

MÁSCARA DE RED D = 11111111 . 11111111 . 11111111 . 11110000 /28

DIRECCIÓN RED D = 11001101 . 00001010 . 11001001 . 00100000 205.10.201.32/28

PRÓXIMO A DIV. = 11001101 . 00001010 . 11001001 . 00110000

MÁSCARA DE RED P-P = 11111111 . 11111111 . 11111111 . 11111100 /30

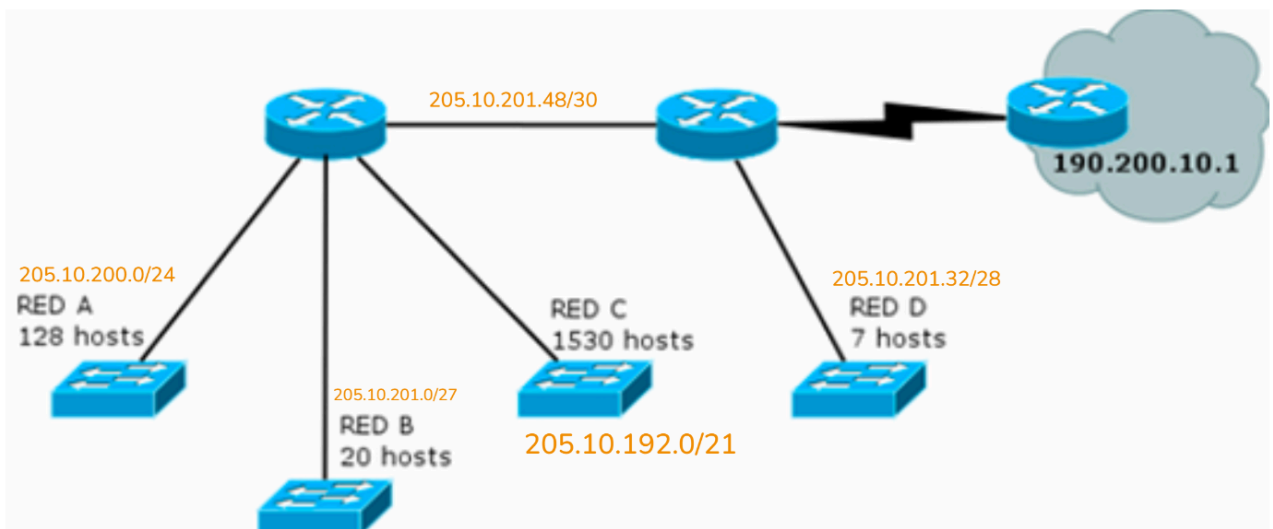
DIRECCIÓN RED P-P = 11001101 . 00001010 . 11001001 . 00110000 205.10.201.48/30

DIRECCIONES LIB. = 11001101 . 00001010 . 11001001 . 00110100

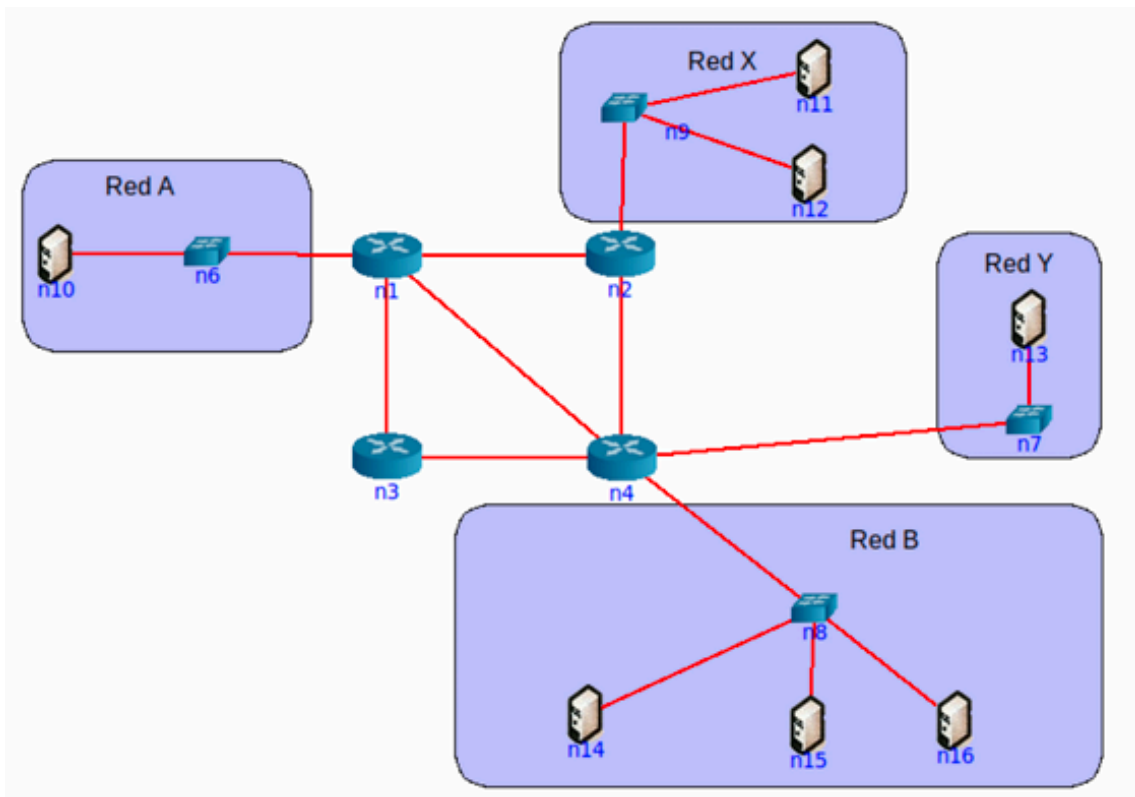
11001101 . 00001010 . 11001001 . 00111000

11001101 . 00001010 . 11001001 . 00111100

NÚMERO CIDR = 11001101 . 00001010 . 11001001 . 00110000 205.10.201.48/28

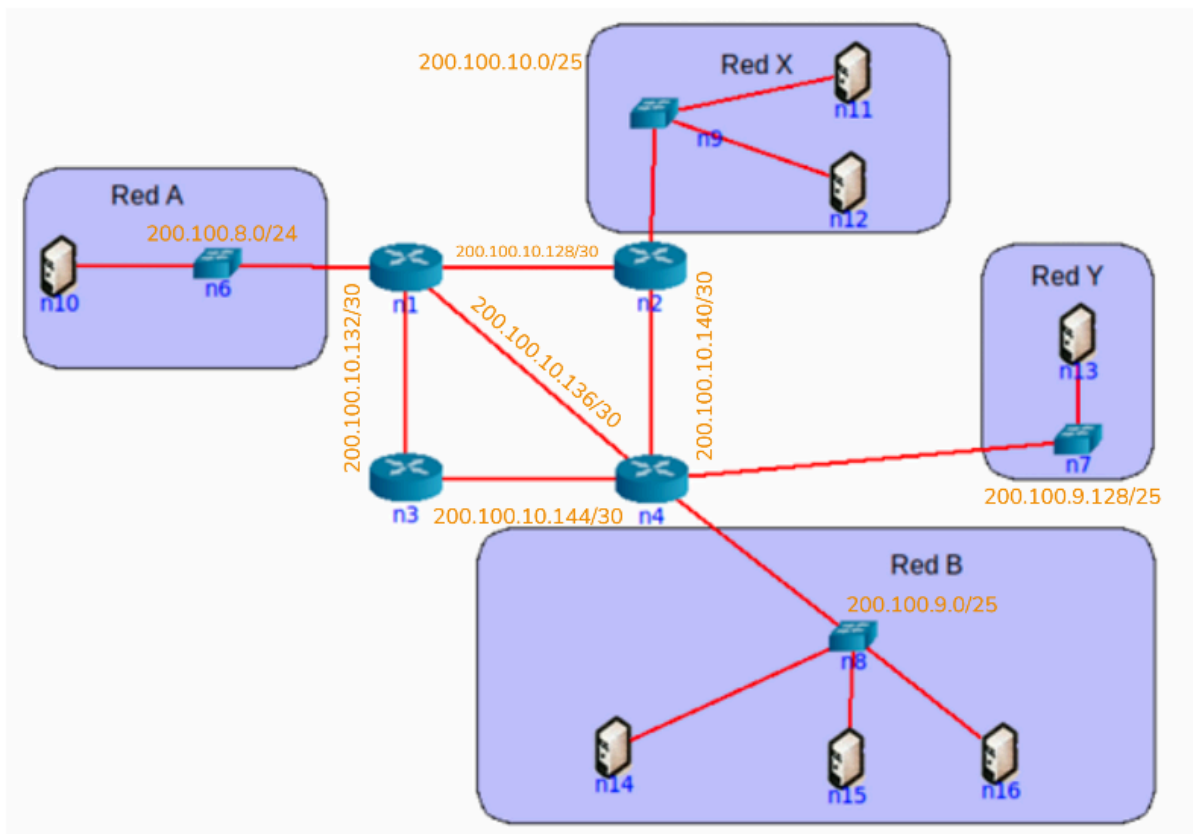


17. Utilizando la siguiente topología y el bloque asignado, arme el plan de direccionamiento IPv4 teniendo en cuenta las siguientes restricciones:



- Utilizar el bloque IPv4 200.100.8.0/22.
- La red A tiene 125 hosts y se espera un crecimiento máximo de 20 hosts.
- La red X tiene 63 hosts.
- La red B cuenta con 60 hosts.
- La red Y tiene 46 hosts y se espera un crecimiento máximo de 18 hosts.
- En cada red, se debe desperdiciar la menor cantidad de direcciones IP posibles. En este sentido, las redes utilizadas para conectar los routers deberán utilizar segmentos de red /30 de modo de desperdiciar la menor cantidad posible de direcciones IP.

18. Asigne direcciones IP en los equipos de la topología según el plan anterior.



IPv4 BLOQUE → 200.100.8.0/22

MÁSCARA ORIGINAL → 11111111 . 11111111 . 11111100 . 00000000 /22

IPv4 BLOQUE EN BIN. → 11001000 . 01100100 . 00001000 . 00000000

RED A (125 + 20 + 2 = 147):

MÁSCARA → 11111111 . 11111111 . 11111111 . 00000000 /24

DIRECCIÓN RED A → 11001000 . 01100100 . 00001000 . 00000000 200.100.8.0/24

PRÓXIMA A DIV. → 11001000 . 01100100 . 00001001 . 00000000

DIR. LIBRES → 11001000 . 01100100 . 00001010 . 00000000 → TOMADA  
11001000 . 01100100 . 00001011 . 00000000

RED B (69 + 2 = 71) →

MÁSCARA → 11111111 . 11111111 . 11111111 . 10000000 /25

DIRECCIÓN RED B → 11001000 . 01100100 . 00001001 . 00000000 200.100.9.0/25

PRÓXIMA A DIV. → 11001000 . 01100100 . 00001001 . 10000000

RED Y (46 + 18 + 2 = 66):

MÁSCARA → 11111111 . 11111111 . 11111111 . 10000000 /25

DIRECCIÓN RED Y → 11001000 . 01100100 . 00001001 . 10000000 200.100.9.128/25

PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 00000000 → TOMAMOS LA PRIMERA LIBRE DE A PORQUE NO HAY LIBRES

RED X (63 + 2 = 65):

MÁSCARA → 11111111 . 11111111 . 11111111 . 10000000 /25

DIRECCIÓN RED X → 11001000 . 01100100 . 00001010 . 00000000 200.100.10.0/25

PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 10000000

RED N1-N2 (2 + 2 = 4):

MÁSCARA → 11111111 . 11111111 . 11111111 . 11111100 /30

DIRECCIÓN RED N1-N2 → 11001000 . 01100100 . 00001010 . 10000000 200.100.10.128/30

PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 10000100

DIR. LIBRES → 11001000 . 01100100 . 00001010 . 10001000

11001000 . 01100100 . 00001010 . 10001100

11001000 . 01100100 . 00001010 . 10010000

11001000 . 01100100 . 00001010 . 10010100

11001000 . 01100100 . 00001010 . 10011000

...

11001000 . 01100100 . 00001010 . 11111100

RED N1-N3 (2 + 2 = 4):

MÁSCARA → 11111111 . 11111111 . 11111111 . 11111100 /30  
DIRECCIÓN RED N1-N3 → 11001000 . 01100100 . 00001010 . 10000100 200.100.10.132/30  
PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 10001000  
DIR. LIBRES → 11001000 . 01100100 . 00001010 . 10001100  
11001000 . 01100100 . 00001010 . 10010000  
11001000 . 01100100 . 00001010 . 10010100  
11001000 . 01100100 . 00001010 . 10011000  
...  
11001000 . 01100100 . 00001010 . 11111100

RED N1-N4 (2 + 2 = 4):

MÁSCARA → 11111111 . 11111111 . 11111111 . 11111100 /30  
DIRECCIÓN RED N1-N4 → 11001000 . 01100100 . 00001010 . 10001000 200.100.10.136/30  
PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 10001100  
DIR. LIBRES → 11001000 . 01100100 . 00001010 . 10010000  
11001000 . 01100100 . 00001010 . 10010100  
11001000 . 01100100 . 00001010 . 10011000  
...  
11001000 . 01100100 . 00001010 . 11111100

RED N2-N4 (2 + 2 = 4):

MÁSCARA → 11111111 . 11111111 . 11111111 . 11111100 /30  
DIRECCIÓN RED N2-N4 → 11001000 . 01100100 . 00001010 . 10001100 200.100.10.140/30  
PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 10010000  
DIR. LIBRES → 11001000 . 01100100 . 00001010 . 10010100  
11001000 . 01100100 . 00001010 . 10011000  
...  
11001000 . 01100100 . 00001010 . 11111100

RED N3-N4 (2 + 2 = 4):

MÁSCARA → 11111111 . 11111111 . 11111111 . 11111100 /30  
DIRECCIÓN RED N3-N4 → 11001000 . 01100100 . 00001010 . 10010000 200.100.10.144/30  
PRÓXIMA A DIV. → 11001000 . 01100100 . 00001010 . 10010100  
DIR. LIBRES → 11001000 . 01100100 . 00001010 . 10011000  
...  
11001000 . 01100100 . 00001010 . 11111100

## ICMP y Configuraciones IP

### 19. Describa qué es y para qué sirve el protocolo ICMP.

- Protocolo auxiliar (Helper) que funciona en conjunto con el Protocolo de Internet (IP) compensando la falta de mecanismos de control de errores en IP.
- No es un protocolo de transporte ya que no fue concebido para llevar datos de usuario, es de la capa 3 (Red).
- IP carece de control, el mismo es dado por un protocolo auxiliar.
- Se encapsula en IP.
- ICMP no agrega confiabilidad a IP. En cambio, proporciona una forma para que los hosts y enrutadores intercambien información vital sobre la red, principalmente informes de error y diagnósticos a modo de feedback.
- Podría ser prescindible en IPv4.
- **Funciones clave:**

- **Informes de Error:** ICMP informa a los hosts sobre varios errores encontrados durante el procesamiento de datagramas IP.
- **Diagnóstico de Red:** ICMP proporciona herramientas para evaluar la conectividad y el rendimiento de la red.

**a. Analice cómo funciona el comando ping.**

**i. Indique el tipo y código ICMP que usa el ping.**

- **Echo Request:** El host emisor envía un paquete ICMP de tipo 8 al destino con código 0. Este paquete contiene un mensaje de solicitud de eco. El código 0 especifica que es una solicitud estándar sin código específico.

**ii. Indique el tipo y código ICMP que usa la respuesta de un ping.**

- **Echo Reply:** El host destino responde con un paquete ICMP de tipo 0 al emisor con código 0. El código 0 especifica que es una respuesta estándar sin código específico.

**b. Analice cómo funcionan comandos como traceroute/tracert de Linux/Windows y cómo manipulan el campo TTL de los paquetes IP.**

- **traceroute** (en Linux/macOS) y **tracert** (en Windows) son herramientas de diagnóstico de red utilizadas para rastrear la ruta que toman los paquetes desde un origen hasta un destino. Su objetivo principal es identificar los nodos intermedios (routers) que atraviesan los paquetes en su camino hacia el destino, proporcionando información útil para diagnosticar problemas de conectividad.
- **Funcionamiento Básico:**
  - Ambos comandos envían paquetes al destino especificado con incrementos progresivos en el campo TTL (Time to Live) de los encabezados IP. La diferencia entre ellos radica principalmente en el protocolo que utilizan:
    - **traceroute en Linux:** Generalmente utiliza paquetes UDP.
    - **tracert en Windows:** Utiliza paquetes ICMP Echo Request (los mismos que ping).
- **Rol del campo TTL (Time to Live)**
  - **Concepto del TTL:**
    - El TTL es un campo en la cabecera IP que indica la cantidad máxima de saltos (hops) que un paquete puede atravesar antes de ser descartado. Se diseñó para evitar bucles infinitos en la red.
  - **Proceso de decremento:**



- Cada router que recibe un paquete reduce el TTL en 1. Si el TTL llega a 0, el router descarta el paquete y envía un mensaje ICMP Time Exceeded de vuelta al emisor.

c. Indique la cantidad de saltos realizados desde su computadora hasta el sitio [www.nasa.gov](http://www.nasa.gov). Analice:

```
> tracert -d www.nasa.gov

Traza a la dirección nasa-gov.go-vip.net [192.0.66.108]
sobre un máximo de 30 saltos:

 1    3 ms    3 ms    3 ms    192.168.0.1
 2    *      *      *      Tiempo de espera agotado para esta solicitud.
 3    *      *      *      Tiempo de espera agotado para esta solicitud.
 4    *      *      *      Tiempo de espera agotado para esta solicitud.
 5    *      *      *      Tiempo de espera agotado para esta solicitud.
 6   12 ms   16 ms    *      181.89.51.39
 7   13 ms   17 ms   15 ms   200.25.50.80
 8  153 ms  178 ms  157 ms  200.25.51.193
 9   44 ms   50 ms   42 ms  200.25.57.147
10   48 ms   45 ms   52 ms  192.0.66.108

Traza completa.
```

- i. **Cómo hacer para que no muestre el nombre del dominio asociado a la IP de cada salto.**
    - Para que no muestre el nombre del dominio asociado a la IP de cada salto, se puede usar la opción -d para evitar la resolución de nombres.
  - ii. **La razón de la aparición de \* en parte o toda la respuesta de un salto.**
    - Cuando aparece un asterisco (\*) en parte o en toda la respuesta de un salto, significa que ese enrutador o dispositivo de red no respondió a la solicitud de "traceroute" o "tracert". Puede ser una medida de seguridad, configuración o simplemente que el enrutador no responde a las solicitudes ICMP utilizadas por "traceroute" para rastrear la ruta.
- d. **Verifique el recorrido hacia los servidores de nombre del dominio unlp.edu.ar. En base al recorrido realizado, ¿podría confirmar cuál de ellos toma un camino distinto?**

```
> tracert -d unlp.edu.ar

Traza a la dirección unlp.edu.ar [163.10.0.135]
sobre un máximo de 30 saltos:

 1      3 ms      3 ms      3 ms  192.168.0.1
 2      *        *        *      Tiempo de espera agotado para esta solicitud.
 3      *        *        *      Tiempo de espera agotado para esta solicitud.
 4      *        *        *      Tiempo de espera agotado para esta solicitud.
 5      *        *        *      Tiempo de espera agotado para esta solicitud.
 6      *        15 ms     *      181.89.51.39
 7     16 ms     17 ms     14 ms  200.0.17.12
 8     18 ms     10 ms     17 ms  200.115.81.1
 9      *        16 ms     19 ms  163.10.199.203
10      *        *        *      Tiempo de espera agotado para esta solicitud.
11      *        *        *      Tiempo de espera agotado para esta solicitud.
12      *        *        *      Tiempo de espera agotado para esta solicitud.
13      *        *        *      Tiempo de espera agotado para esta solicitud.
14      *        *        *      Tiempo de espera agotado para esta solicitud.
15      *        *        *      Tiempo de espera agotado para esta solicitud.
16      *        *        *      Tiempo de espera agotado para esta solicitud.
17      *        *        *      Tiempo de espera agotado para esta solicitud.
18      *        *        *      Tiempo de espera agotado para esta solicitud.
19      *        *        *      Tiempo de espera agotado para esta solicitud.
20      *        *        *      Tiempo de espera agotado para esta solicitud.
21      *        *        *      Tiempo de espera agotado para esta solicitud.
22      *        *        *      Tiempo de espera agotado para esta solicitud.
23      *        *        *      Tiempo de espera agotado para esta solicitud.
24      *        *        *      Tiempo de espera agotado para esta solicitud.
25      *        *        *      Tiempo de espera agotado para esta solicitud.
26      *        *        *      Tiempo de espera agotado para esta solicitud.
27      *        *        *      Tiempo de espera agotado para esta solicitud.
28      *        *        *      Tiempo de espera agotado para esta solicitud.
29      *        *        *      Tiempo de espera agotado para esta solicitud.
30      *        *        *      Tiempo de espera agotado para esta solicitud.

Traza completa.
```

- No, no es posible determinar el camino que siguen los servidores de nombres del dominio unlp.edu.ar debido a que todas las respuestas muestran "Tiempo de espera agotado para esta solicitud.". Esto puede ser porque los servidores de nombres de dominio están configurados para no responder a traceroute o porque hay un problema en la red que impide que las respuestas lleguen de vuelta.

## 20. ¿Para que se usa el bloque 127.0.0.0/8? ¿Qué PC responde a los siguientes comandos?

- El bloque de direcciones IP 127.0.0.0/8 está reservado para el uso en la red de loopback. La dirección IP más comúnmente utilizada en este bloque es 127.0.0.1, que se conoce como "localhost". La dirección de loopback se utiliza para permitir que un dispositivo se comunique consigo mismo, lo que es útil en el diagnóstico y la prueba de aplicaciones y servicios de red sin la necesidad de acceder a una red real.

a. ping 127.0.0.1.

```
> 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
```

b. ping 127.0.54.43.

```
> ping 127.0.54.43

Haciendo ping a 127.0.54.43 con 32 bytes de datos:
Respuesta desde 127.0.54.43: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.54.43: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.54.43: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.54.43: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 127.0.54.43:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

21. Investigue para qué sirven los comandos `ifconfig` y `route`. ¿Qué comandos podría utilizar en su reemplazo? Inicie una topología con CORE, cree una máquina y utilice en ella los comandos anteriores para practicar sus diferentes opciones, mínimamente:

- Configurar y quitar una dirección IP en una interfaz.
- Ver la tabla de ruteo de la máquina.

## Función y reemplazos de `ifconfig` y `route`

### Comando `ifconfig`

- **Función:** Se utiliza para configurar interfaces de red en sistemas Linux. Permite asignar o cambiar direcciones IP, habilitar o deshabilitar interfaces, y mostrar información detallada sobre ellas.

- **Ejemplo:**  

```
ifconfig eth0 192.168.1.10 netmask 255.255.255.0 up
```
- **Reemplazo:** `ip` (paquete `iproute2`) ha reemplazado a `ifconfig`.
  - **Ejemplo equivalente:**  

```
ip addr add 192.168.1.10/24 dev eth0
ip link set eth0 up
```

## Comando **route**

- **Función:** Se utiliza para manipular la tabla de rutas, mostrando o modificando las rutas estáticas del sistema.
- **Ejemplo:**  

```
route add default gw 192.168.1.1
```
- **Reemplazo:** `ip route` es el reemplazo moderno.
  - **Ejemplo equivalente:**  

```
ip route add default via 192.168.1.1
```

## Práctica con CORE (Common Open Research Emulator)

1. **Iniciar CORE:**  
 Al abrir CORE, puedes crear una topología básica:
  - Agrega un nodo, que simulará una máquina.
  - Configurar interfaces y conexiones en este nodo.
2. **Configuración de IP:**
  - Abre la terminal del nodo.
- Configura una dirección IP con `ifconfig` o `ip`:  

```
ifconfig eth0 10.0.0.1 netmask 255.255.255.0 up
# o con ip
ip addr add 10.0.0.1/24 dev eth0
```
- Para eliminar la dirección IP:  

```
ifconfig eth0 down
# o con ip
ip addr del 10.0.0.1/24 dev eth0
```
3. **Ver la tabla de ruteo:**  

```
route -n
# o con ip
ip route show
```