

Arquitectura de Clases en IP



Arquitectura de Clases en IP

- En las primeras etapas del desarrollo del protocolo IP, las direcciones se interpretaban como 8 bits para la dirección de red y 24 para la dirección de host.

154 . 200 . 31 . 1

Arquitectura de Clases en IP

- En 1981 se revisó el direccionamiento y se introdujo la arquitectura de clases.
- Conocido como **Classful Addressing**
- El espacio de direcciones se dividió en clases
 - Clase A (126 redes, 16.777.216 hosts c/u)
 - Clase B (16.384 redes, 65.534 hosts c/u)
 - Clase C (2.097.152 redes, 256 hosts c/u)

Arquitectura de Clases en IP

Clase A:

- Prefijo /8
- Redes: 0.0.0.0/8 - 126.0.0.0/8
- Direcciones: 0.0.0.0 - 126.255.255.255
- Primeros bits: 0

10 . 20 . 0 . 7

Arquitectura de Clases en IP

Clase B:

- Prefijo /16
- Redes: 127.0.0.0/16 – 191.255.0.0/16
- Direcciones: 127.0.0.0 – 191.255.255.255
- Primeros bits: 10

154 . 13 . 22 . 3

Arquitectura de Clases en IP

Clase C:

- Prefijo /24
- Redes: 192.0.0.0/24 a 223.255.255.0/24
- Direcciones: 192.0.0.0 – 223.255.255.255
- Primeros bits: 110

200 . 4 . 67 . 3

Arquitectura de Clases en IP

- **Clase D (Multicast):**
 - 224.0.0.0 – 239.255.255.255
 - Primeros bits: 1110
- **Clase E (Experimental):**
 - 240.0.0.0 – 255.255.255.255
 - Primeros bits: 1111

Arquitectura de Clases en IP

- En 1993, se introduce **Classless Inter-Domain Routing (CIDR)**
 - Máscaras de subred de longitud variable.
 - **Notación CIDR**

200.4.67.3/23

Subnetting

Subnets

- Una subnet es una subdivisión **lógica** de una red.
- Permiten agrupar un conjunto de direcciones en un rango.
- ¿Cómo se describen?
 - Ejemplo: 192.168.1.0/24

¿Por qué existen las subnets?

- El ***masking*** reduce la lógica que tienen que manejar los routers.
- Separar el tráfico en subredes optimiza el uso de la red (broadcast domains)
- Reduce congestión de red
- Aumenta la seguridad (impl ACLs)
- Facilita la administración

Ejercicio

¿Qué buscamos al resolver ejercicios?

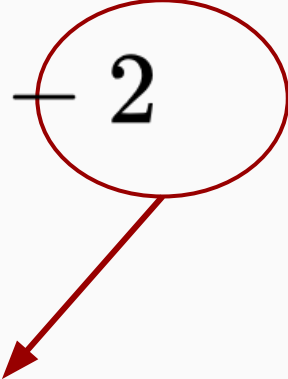
Para una dada subred, buscamos **maximizar la cantidad de direcciones ocupadas** (cada dirección inactiva desperdicia \$\$)

Entonces, resolver un ejercicio de subnetting es resolver un ejercicio de optimización de recursos

Metodología

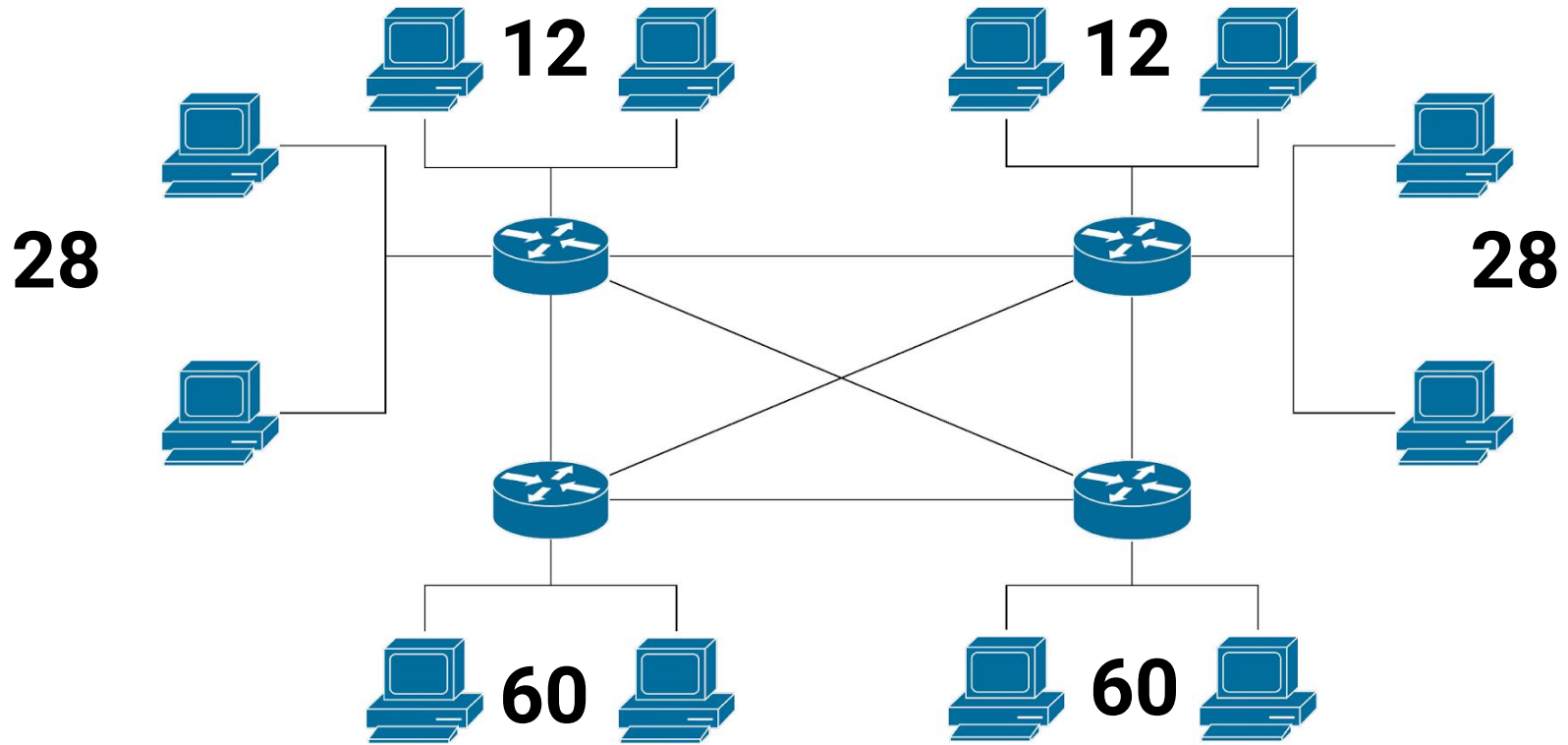
- Todas las subredes tiene una **dirección de red** y una **dirección de broadcast**
- **No se puede** asignar la dirección de red o la de broadcast a un host
- Todos los dispositivos conectados a la red tienen una dirección IP (**los routers también!**)

Metodología

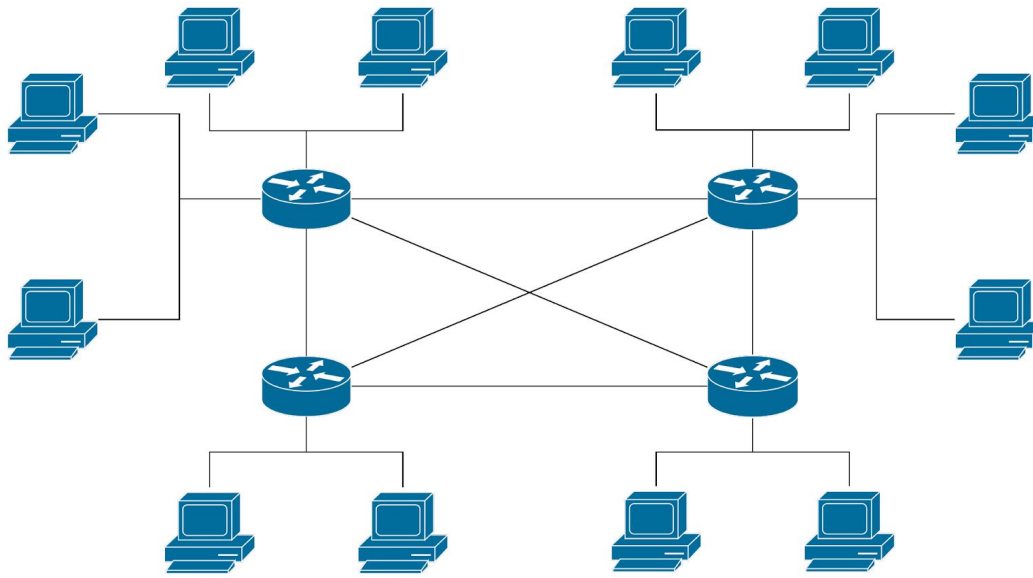
$$\#de\ hosts = 2^{32 - mascara} - 2$$


La direcciones de red y
broadcast no se
pueden usar!

Enunciado

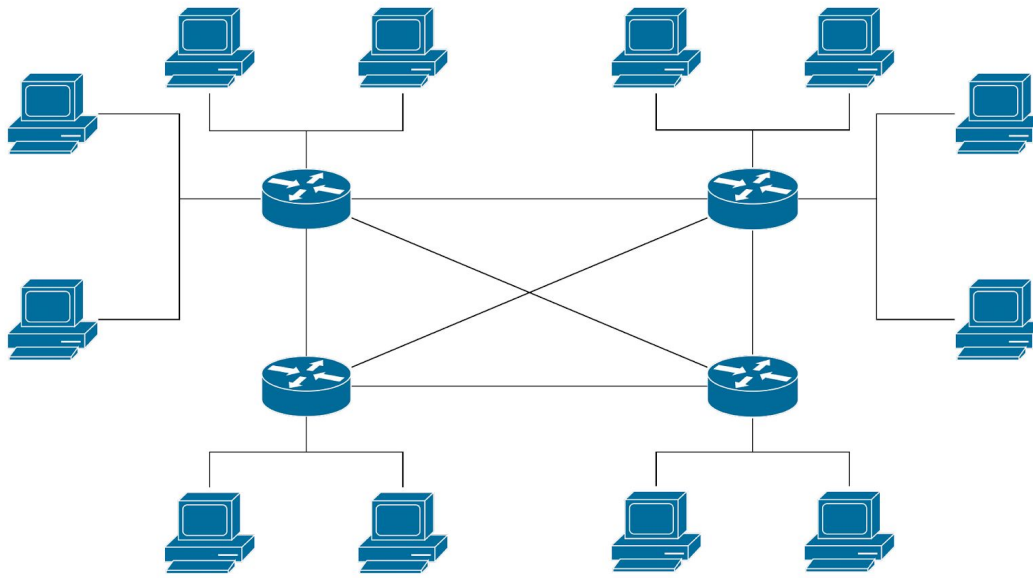


Disponibile: 201.10.3.0/24



Análisis a priori:

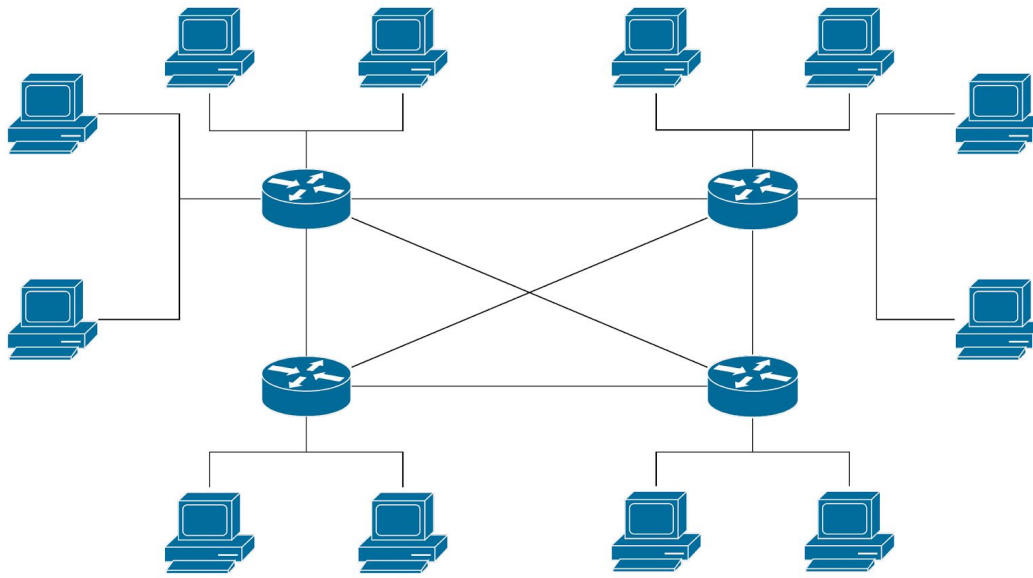
- ~ 6 subredes
- 4 routers
- 6 enlaces entre routers



Análisis a priori:

- ~ 6 subredes
- 4 routers
- 6 enlaces entre routers

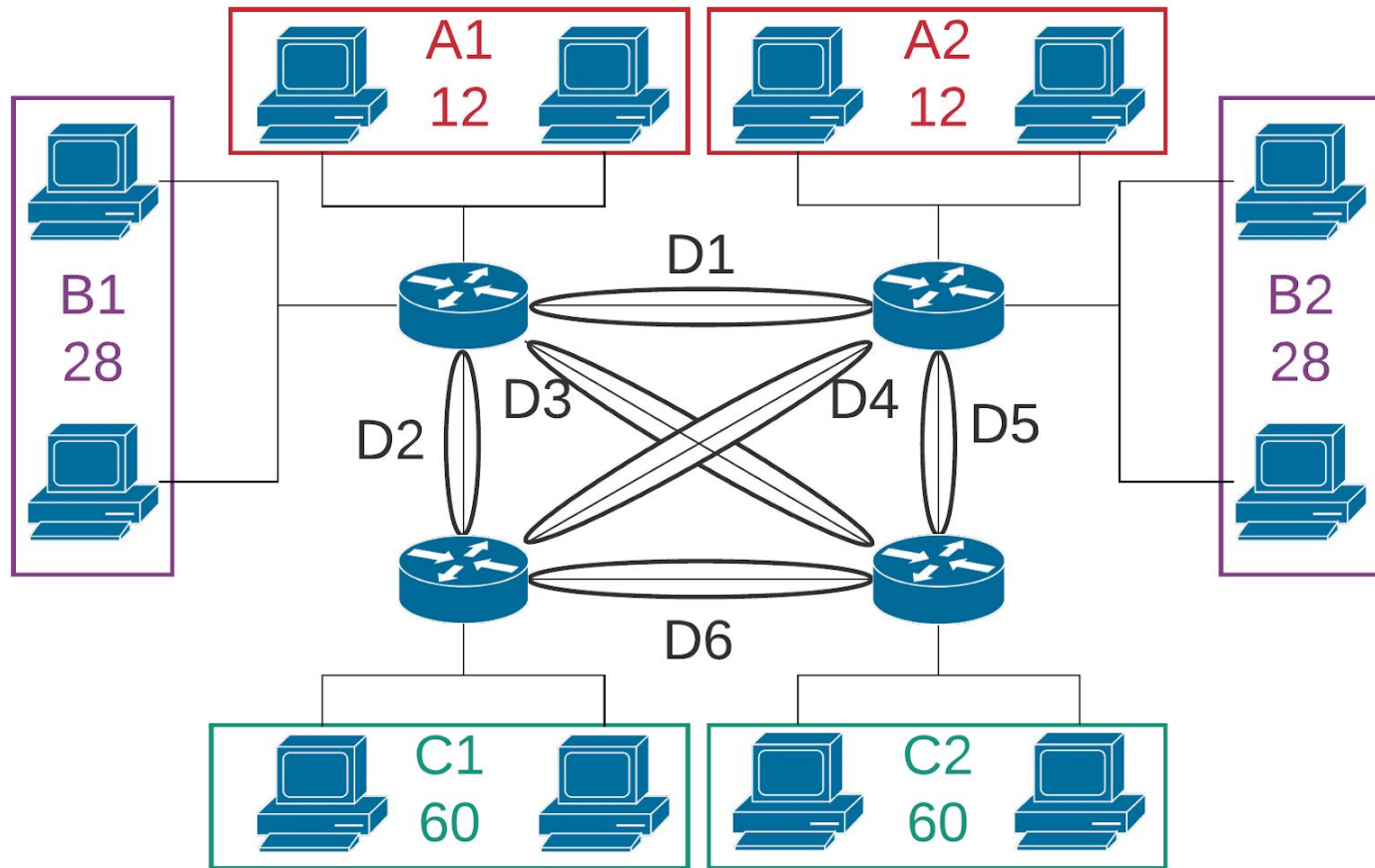
Cuidado! Los enlaces “punto a punto” entre routers necesitan tener una subred

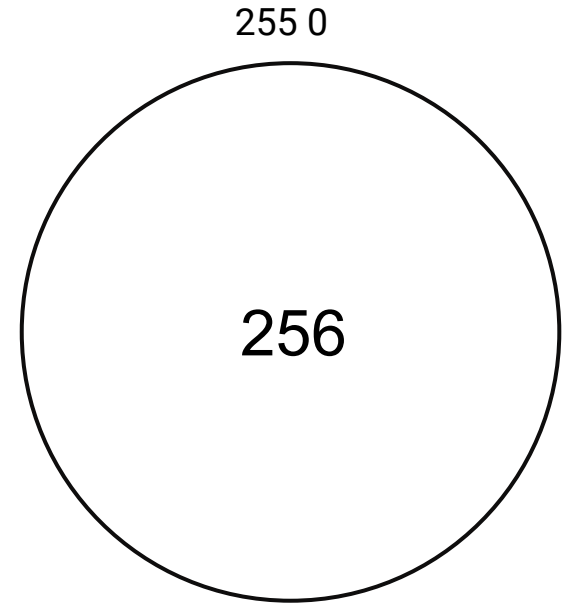
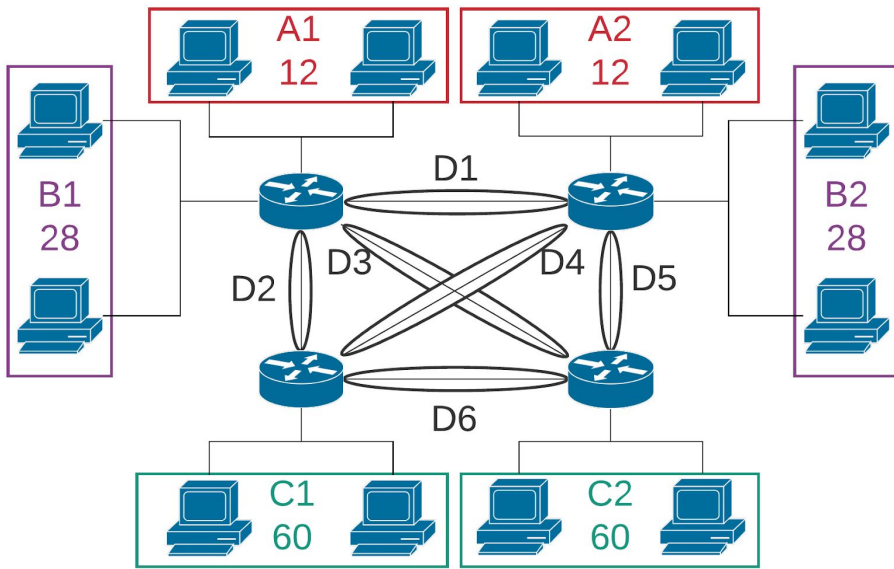


Análisis a priori:

- **12 subredes**
- 4 routers
- 6 enlaces entre routers

Cuidado! Los enlaces “punto a punto” entre routers necesitan tener una subred

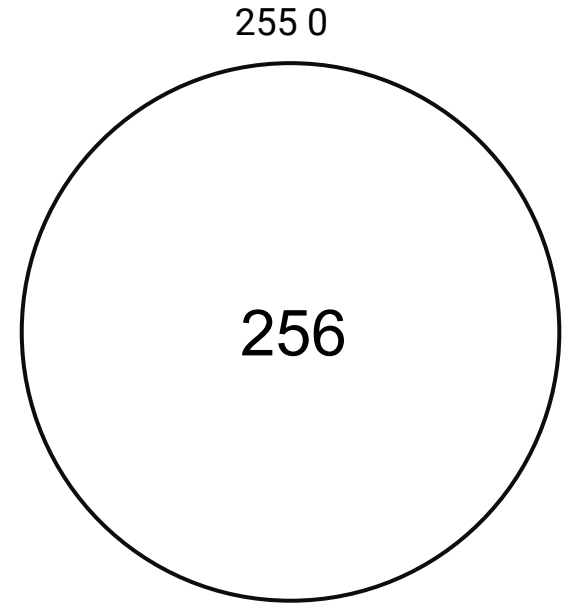
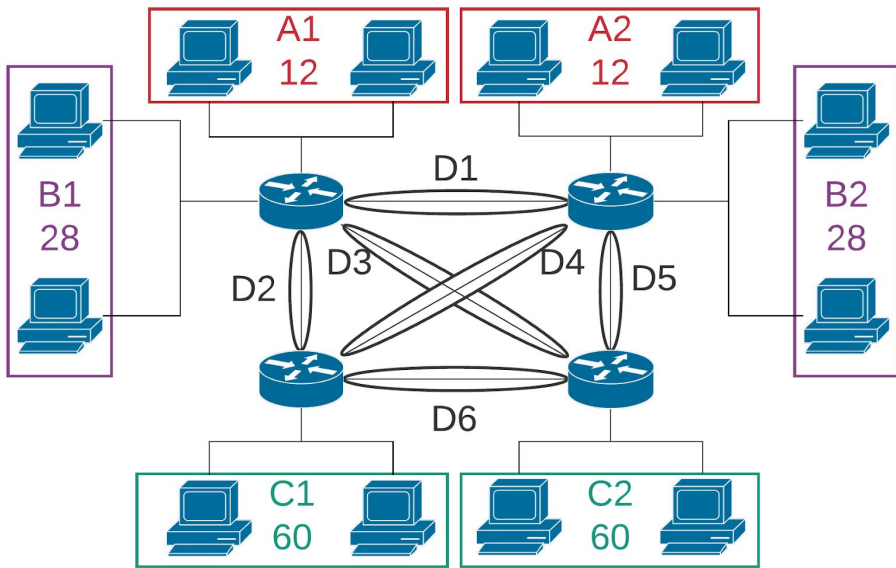




201.10.3.0/24

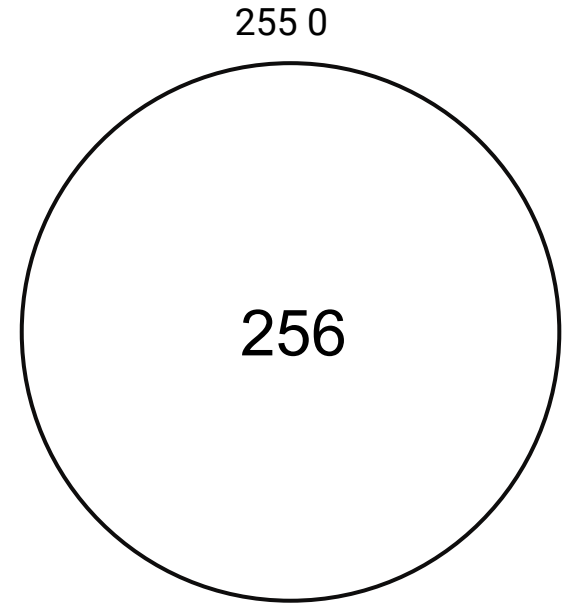
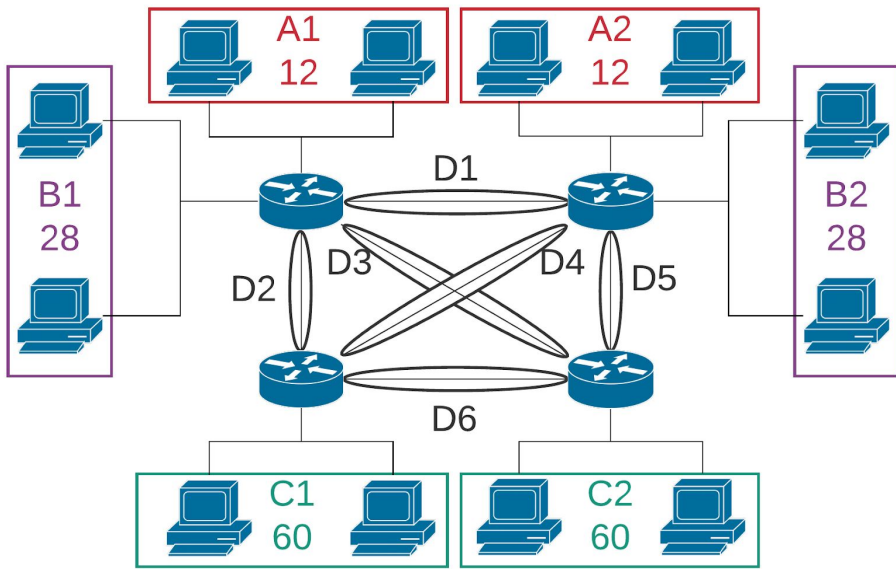
Delimitando las subredes

- Averiguar cuántos hosts tiene c/u (no olvidar el router!)
- Averiguar el espacio de direcciones a subdividir
- Recordar el objetivo: asignar la mínima cantidad de IPs a cada subred



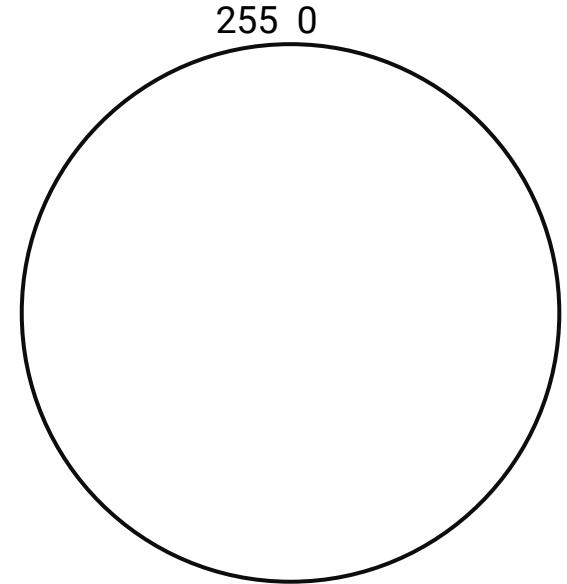
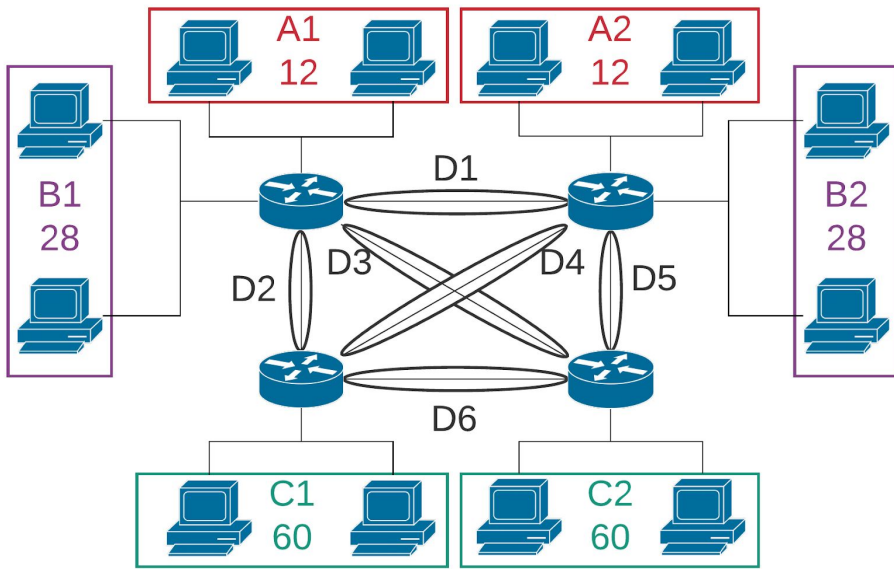
201.10.3.0/24

- ¿Cuál es la/s red/es con más hosts?



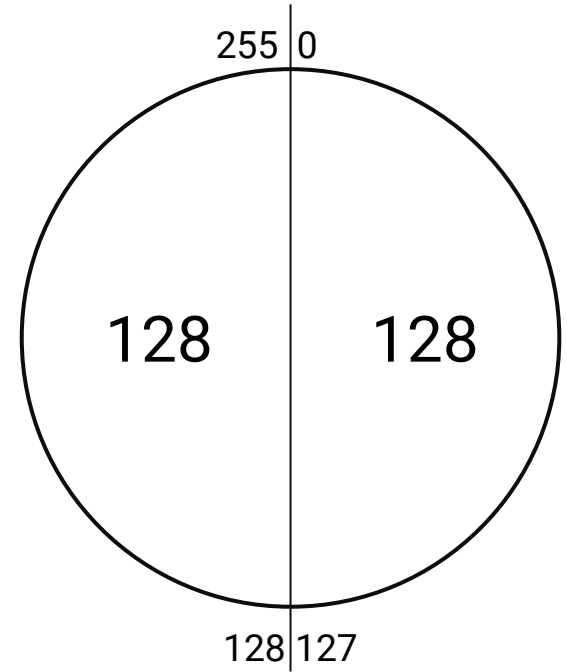
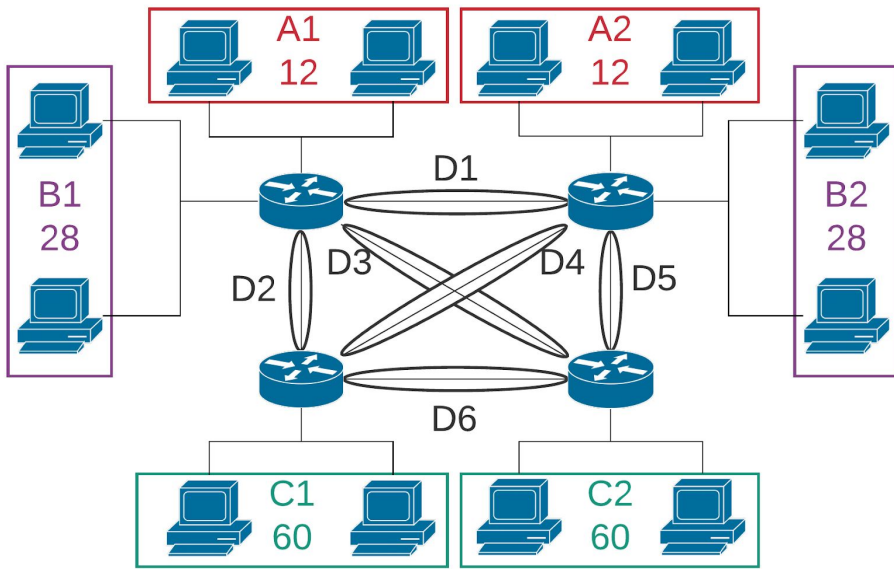
201.10.3.0/24

- ¿Cuál es la/s red/es con más hosts? -> **C1 y C2**



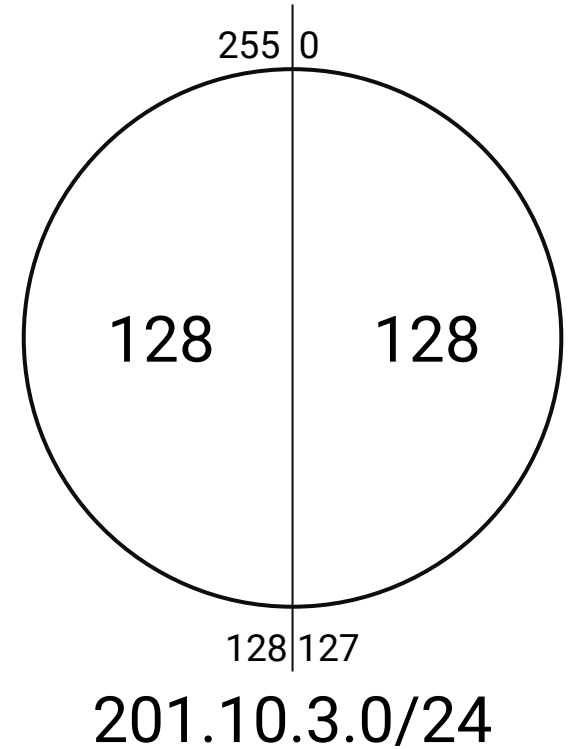
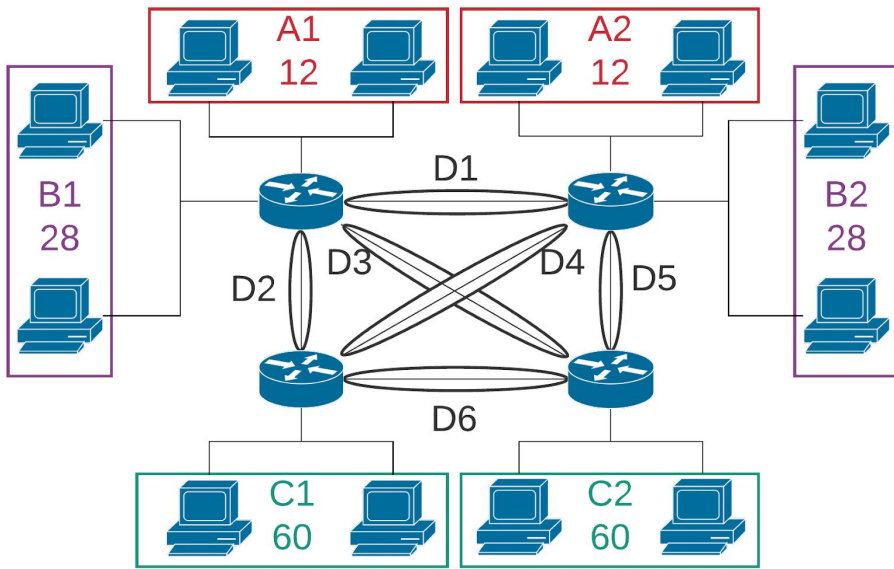
201.10.3.0/24

- ¿Cuál es la/s red/es con más hosts? -> **C1 y C2 (60 hosts)**
- Tengo una red con 256 IPs. 256 >> a 60 -> divido la red en 2

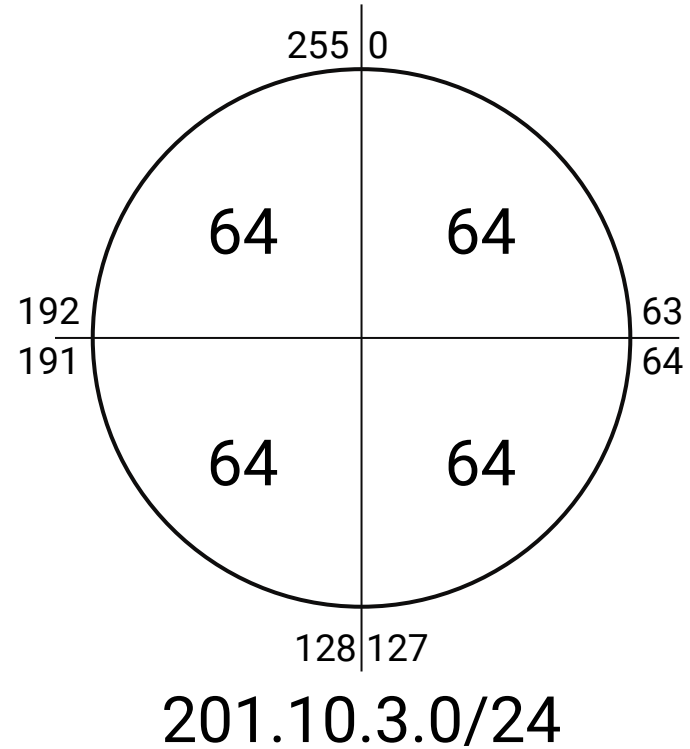
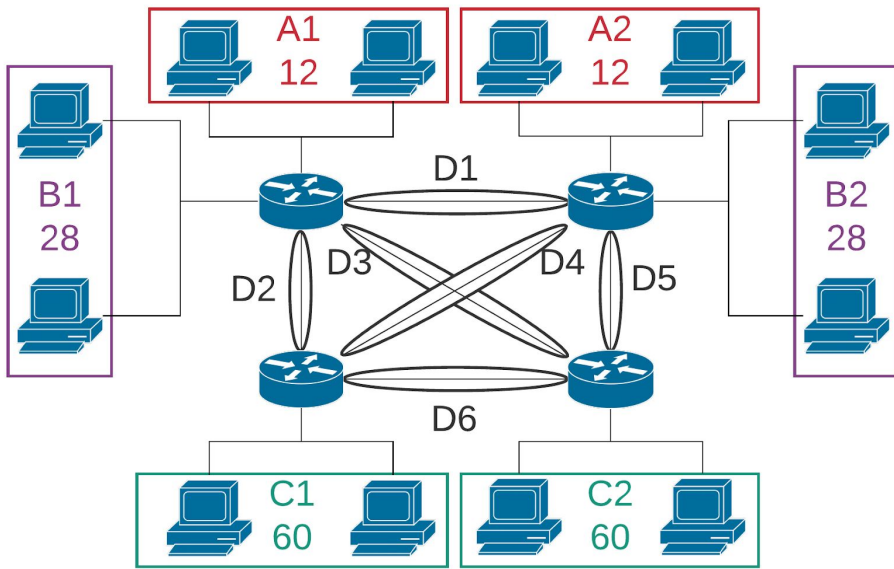


201.10.3.0/24

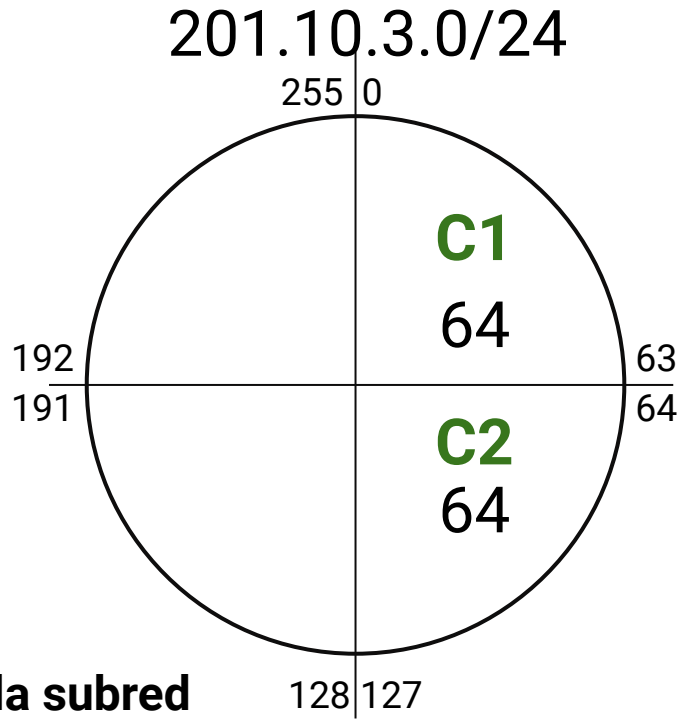
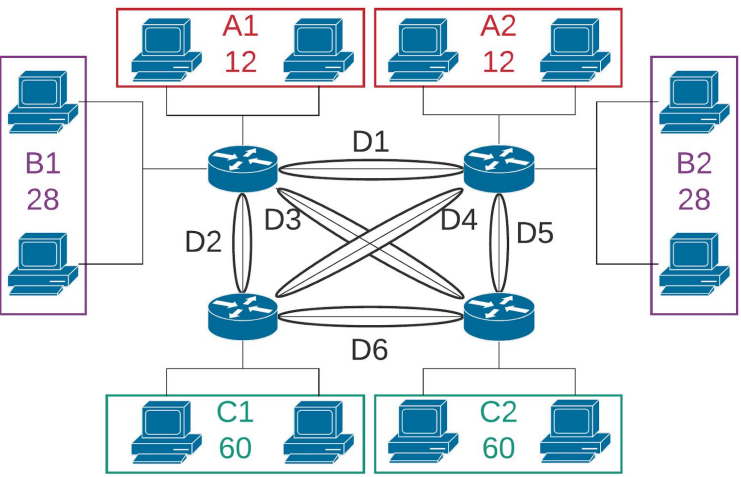
- ¿Cuál es la/s red/es con más hosts? -> **C1 y C2 (60 hosts + 1 router = 61)**
- Tengo una red con 256 IPs. 256 >> a 61 -> divido la red en 2
- Divido la red en 2 -> 2 redes /25



- ¿Cuál es la/s red/es con más hosts? -> **C1 y C2 (60 hosts)**
- Tengo una red con 256 IPs. $256 >> a\ 61$ -> divido la red en 2
- Divido la red en 2 -> 2 redes /25
- Todavía son muy grandes ($128 > 61$)



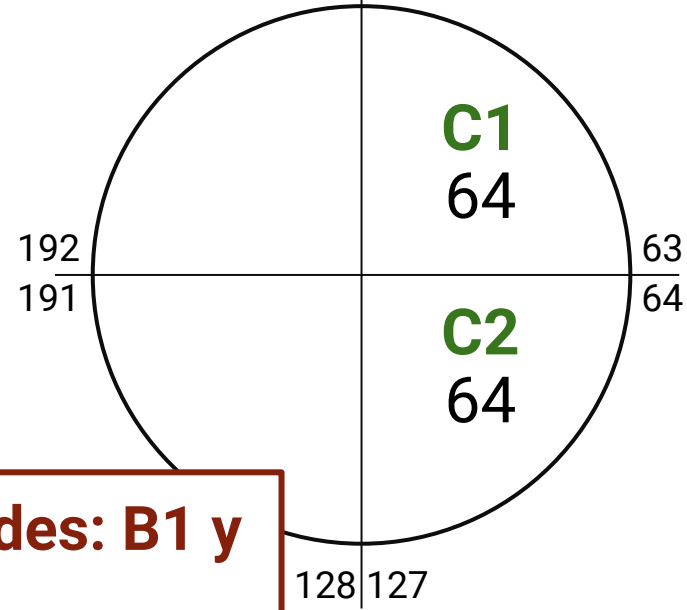
- Vuelvo a subdividir
- Ahora 4 tenemos redes /26 de 64 hosts
- Si dividiera de nuevo tendría redes de 32. Como **32 < 61** en este momento debo ubicar a **C1 y C2**



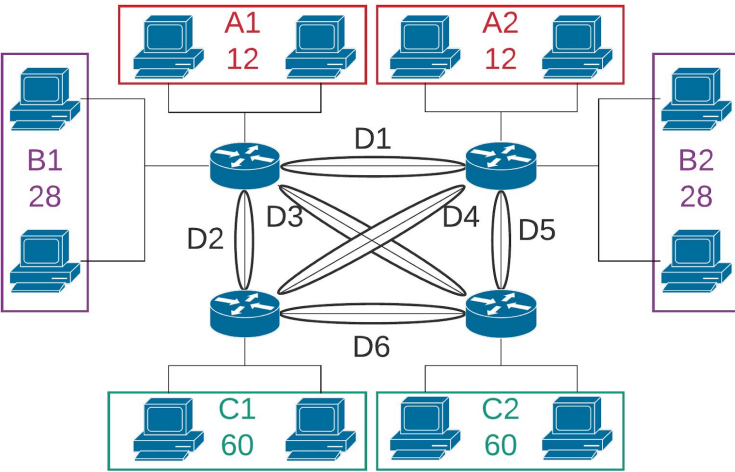
Subred	Dirección de la subred	Subred	Dirección de la subred
C1	201.10.3.0 / 26	D1	
C2	201.10.3.64 / 26	D2	
B1		D3	
B2		D4	
A1		D5	
A2		D6	

201.10.3.0/24

255 0

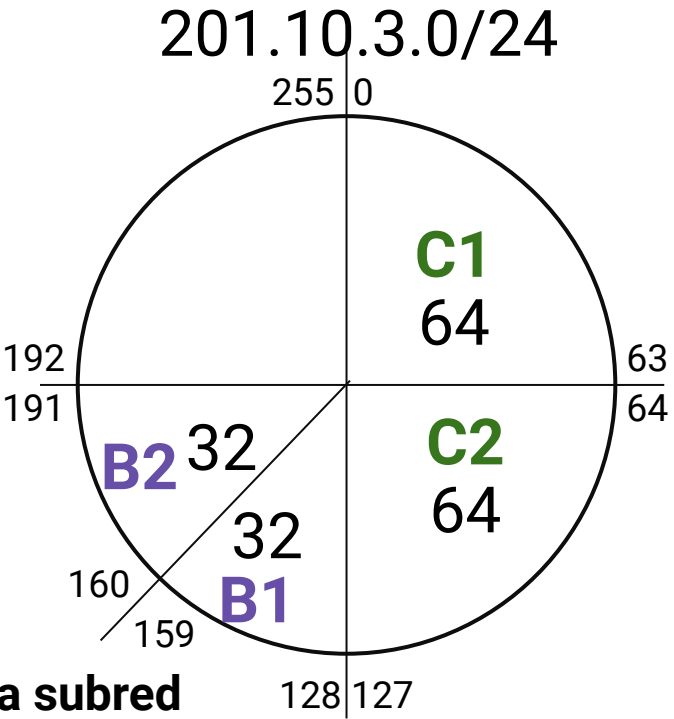
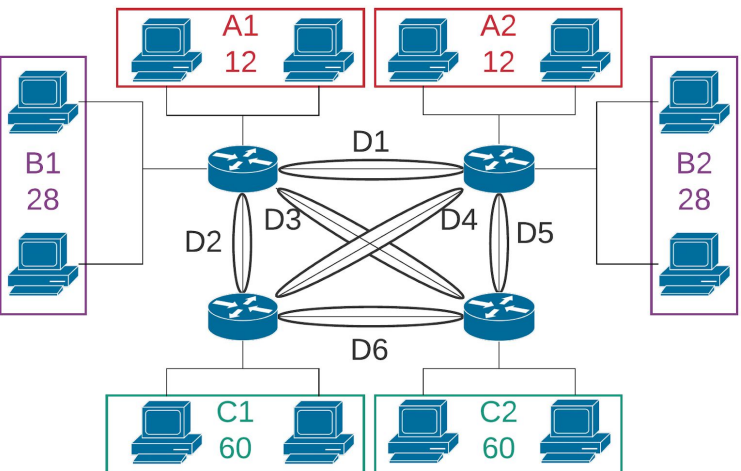


- **Siguiente redes más grandes: B1 y B2 (28)**
- **Tenemos redes de 64. Si partimos de nuevo, nos quedan de 32. Como $29^* < 30$, podemos dividir**



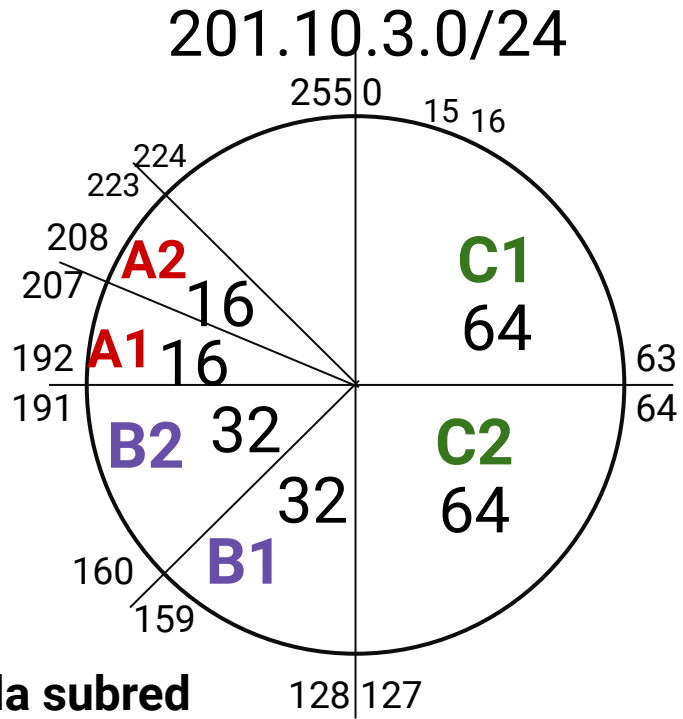
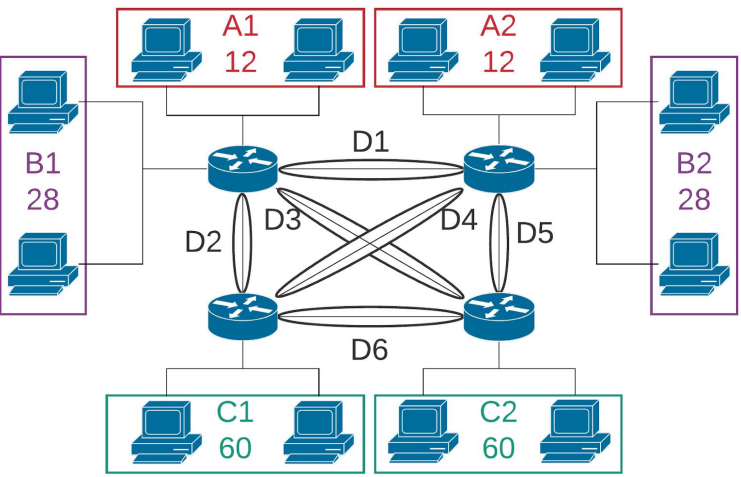
Subred	Direcci
C1	201.10
C2	201.10
B1	
B2	
A1	
A2	

D3
D6



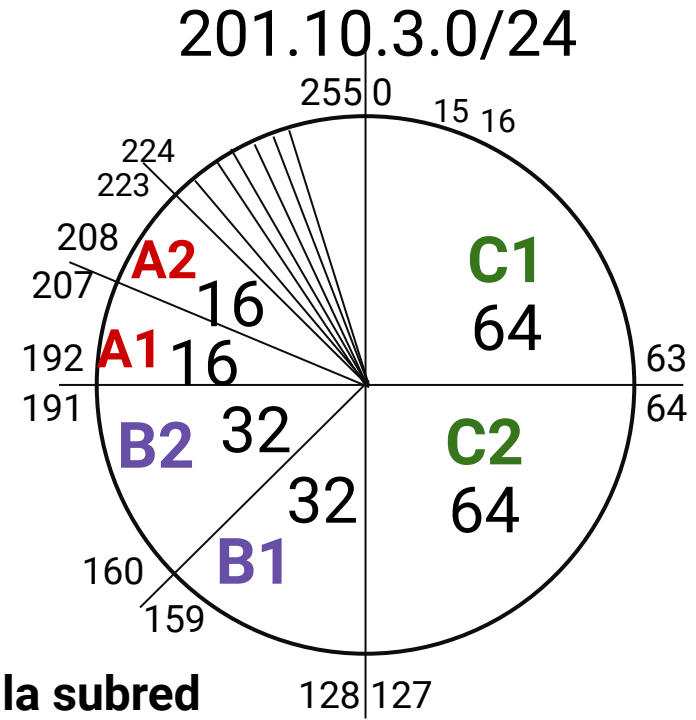
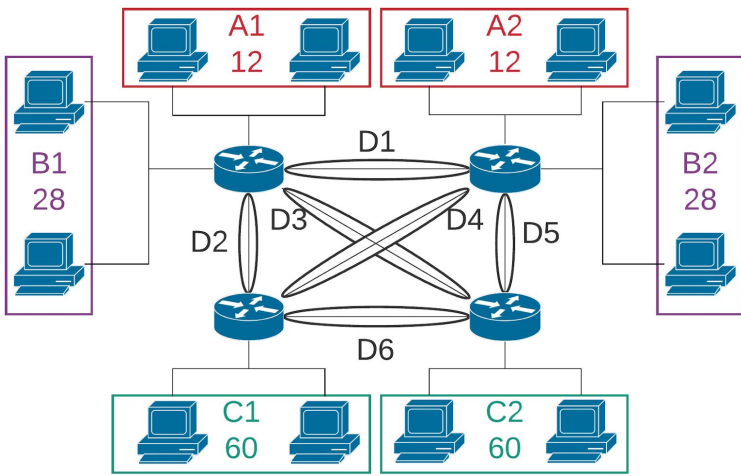
Subred	Dirección de la subred	Subred
C1	201.10.3.0 / 26	D1
C2	201.10.3.64 / 26	D2
B1	201.10.3.128 / 27	D3
B2	201.10.3.160 / 27	D4
A1		D5
A2		D6

Dirección de la subred

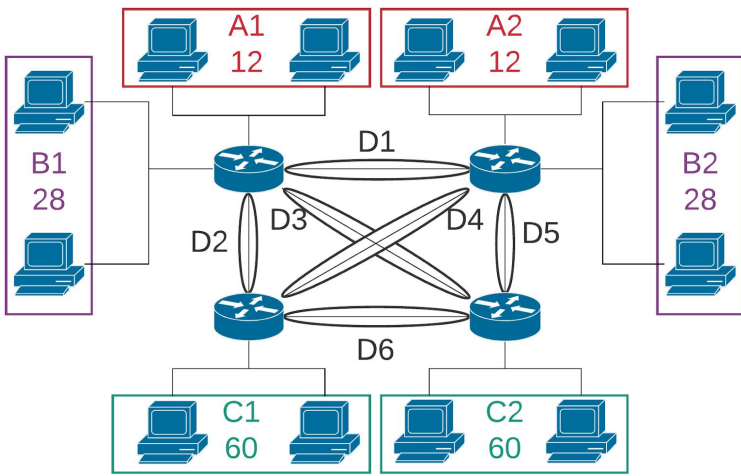


Subred	Dirección de la subred	Subred
C1	201.10.3.0 / 26	D1
C2	201.10.3.64 / 26	D2
B1	201.10.3.128 / 27	D3
B2	201.10.3.160 / 27	D4
A1	201.10.3.192/ 28	D5
A2	201.10.3.208/ 28	D6

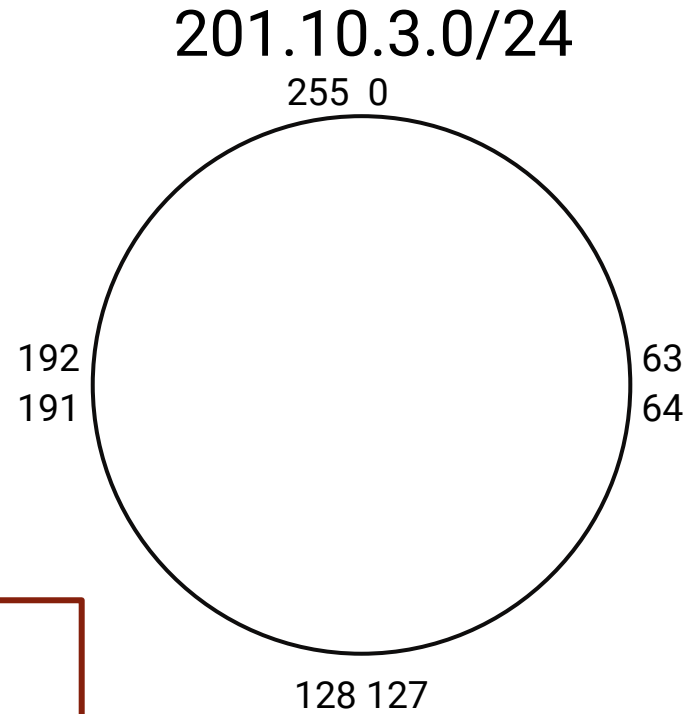
Dirección de la subred

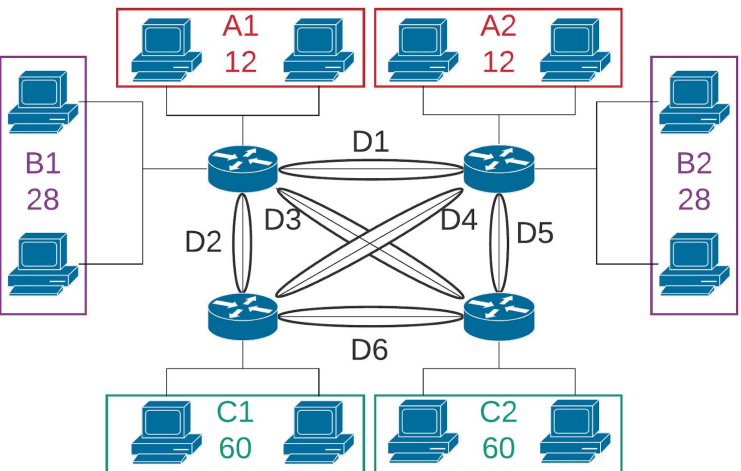


Subred	Dirección de la subred	Subred	Dirección de la subred
C1	201.10.3.0 / 26	D1	201.10.3.224 / 30
C2	201.10.3.64 / 26	D2	201.10.3.228 / 30
B1	201.10.3.128 / 27	D3	201.10.3.232 / 30
B2	201.10.3.160 / 27	D4	201.10.3.236 / 30
A1	201.10.3.192 / 28	D5	201.10.3.240 / 30
A2	201.10.3.208 / 28	D6	201.10.3.244 / 30

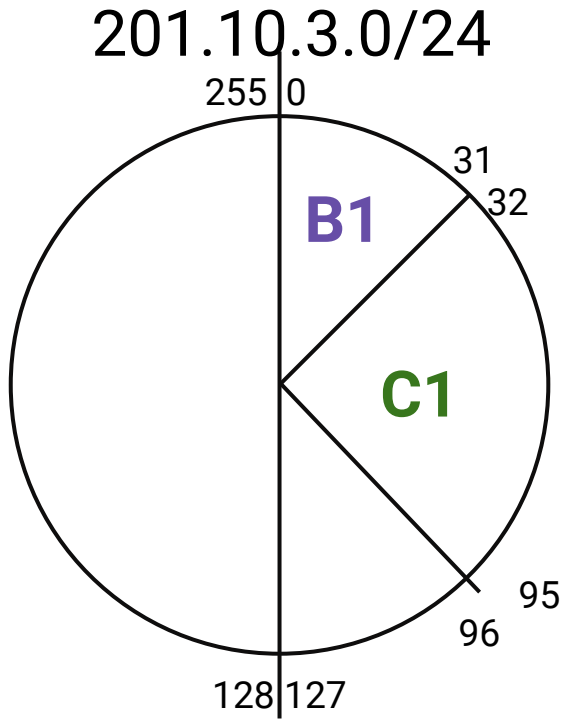


Ojo! Es muy importante arrancar por la red con más hosts





Subred	Dirección de la subred
B1	201.10.3.0/27
C1	201.10.3.32/26




Pero pero pero...

- Supuestamente la red C1 va de 201.10.3.32 a 201.10.3.95
- Calculemos la red aplicando masking y veamos que pasa


Pero pero pero...

Resolvamos la máscara

- 201.10.3.32/26 implica quedarnos con los primeros 26 bits como subred
- Los tres primeros octetos quedan igual (son los primeros $8 \times 3 = 24$ bits)
- Veamos el último
 - $32 =$ 
Red Host

Pero pero pero...

Resolvamos la máscara

- 201.10.3.32/26 implica quedarnos con los primeros 26 bits (son los primeros 255)
- Los tres primeros octetos son los primeros 24 bits
- Veamos el último
 - 32 = 

Pero pero pero...

Resolvamos la máscara

- 20 **Corolario:** empezar siempre por la red con mayor cantidad de hosts. Ante la duda, aplicar la máscara a cualquier dirección dentro de la red. Tiene que dar la dirección de red!
- Lo
- los
- Veamos el ejemplo

○ $32 = \underbrace{00}_{\text{Red}} \underbrace{1000000}_{\text{Host}}$

Aclaración

- Subnet zero y all ones networks: 'redes prohibidas'
 - RFC 950: ambigüedad de redes
 - No se podían usar las direcciones de red y broadcast de la clase en las redes
 - Hace años no se obliga a cumplirla

Referencias

- RFC 950: Internet Standard Subnetting Procedure
<https://tools.ietf.org/html/rfc950>
- RFC 4632: Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan <https://tools.ietf.org/html/rfc4632>