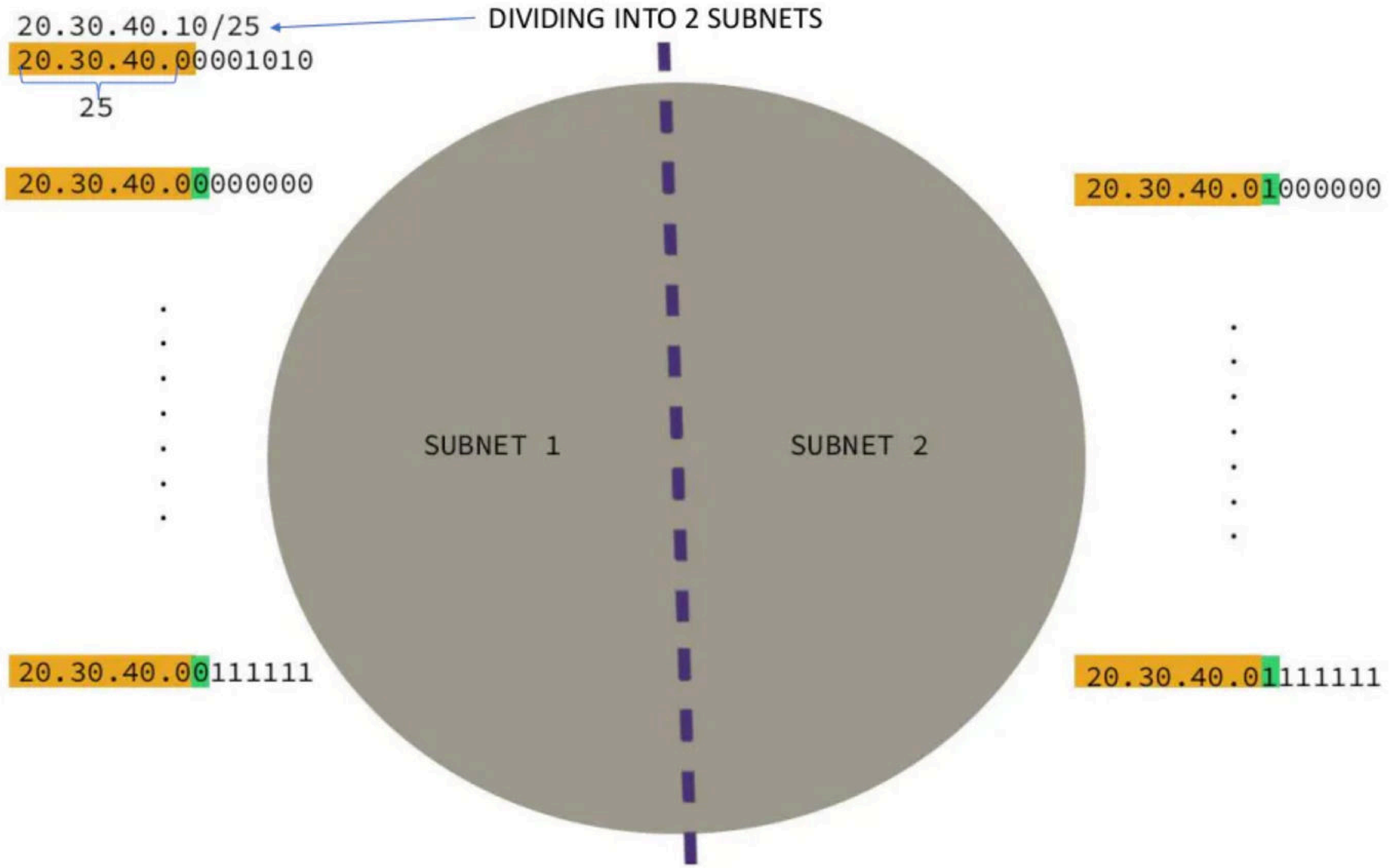
Doubt Clearing Session

Complete Course on Computer Networks - Part I

Ravindrababu RAVULA • Lesson 11 • Jan 29, 2021

Computer Networks

Subnetting in CIDR, VLSM in CIDR

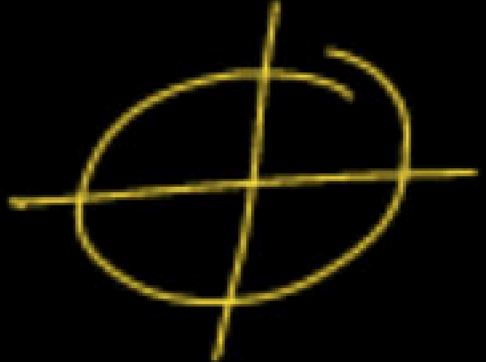

20.30.40.10/25
DIVIDING INTO 2 SUBNETS
20.30.40.00001010
25
20.30.40.00000000
20.30.40.01000000
SUBNET 1
SUBNET 2
20.30.40.00111111
20.30.40.01111111

① $\boxed{20.30.40.10/25}$ — 1/4, 1/4, 1/4, 1/4

20.30.40.0 | 0001010 — HID - 7

.0
.127

$2^7 = 128$, $\frac{128}{4} = 2^5 = 32$ ✓

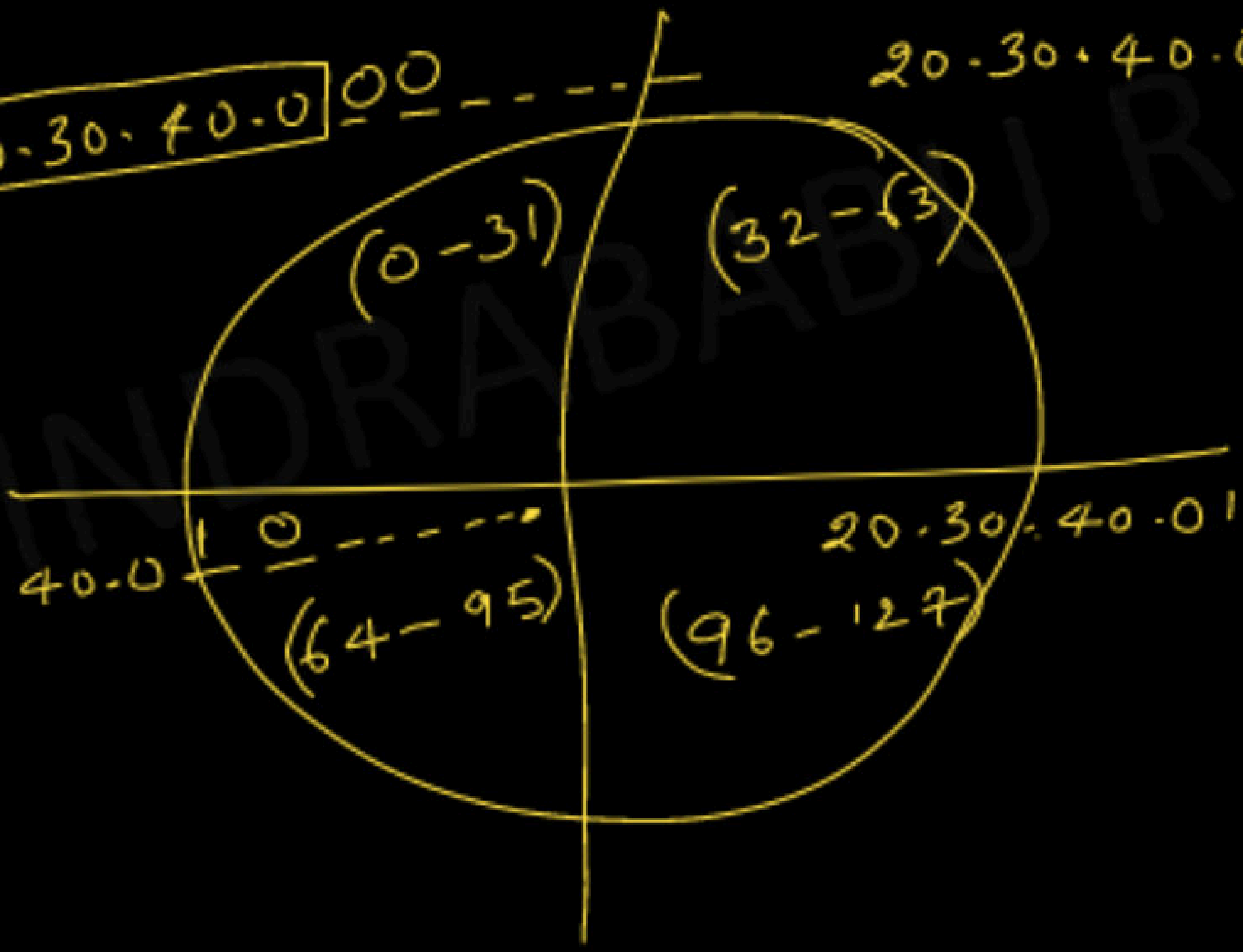


A - (0-31) /27 $2^5 = 32$
B - (32-63)
C - (64-95)
D - (95-127)

2^5 HID = 5
NID = 27.

② 20.30.40.0 00 - - - - -

20.30.40.0 0 1 - - - - - ✓ ① ②



20.30.40.0 1 0 - - - - -

20.30.40.0 11 - - - - -

20.30.40.10/25 — [1/2 ; 1/2 — [1/4 ; 1/4

255.255.255.128

(VLSM) (0-127) - $\frac{128}{2}$ = 64, $\frac{64}{2}$ → 32, 32

(0-63) (64-95) (96-127)

20.30.40.0 0 - - - - -

20.30.40.0/26

SM: 255.255.255.192

20.30.40.0 1 0 - - - - -

20.30.40.64/27

SM: 255.255.255.224

20.30.40.011 - - - - -

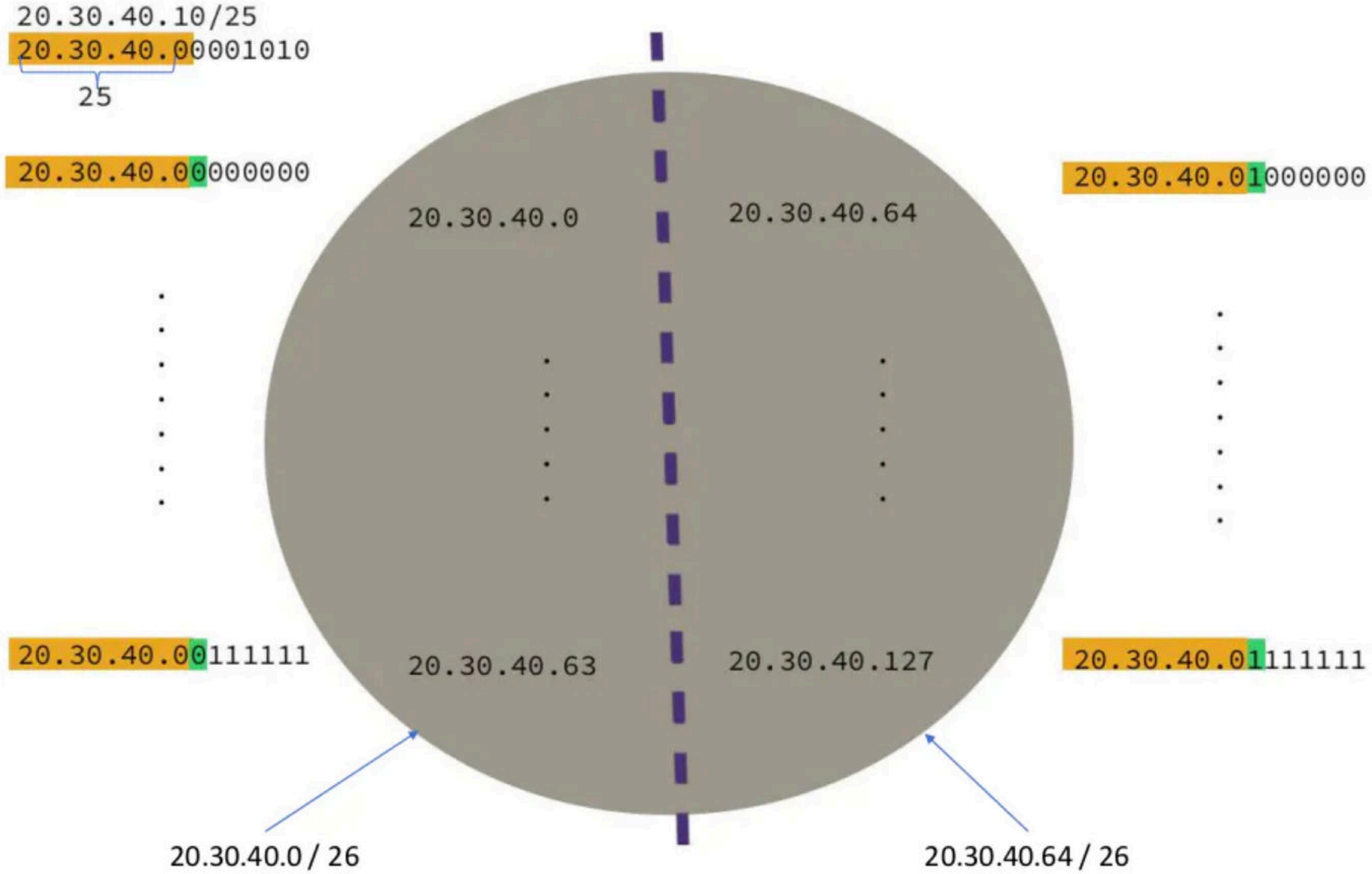
20.30.40.96/27

SM: "

20.30.40.0 0 0 - - - - -

0 1

(0-31) (32-63) (64-127)



20.30.40.10/25
20.30.40.00001010
25
20.30.40.00000000
20.30.40.00111111
20.30.40.0
20.30.40.63
20.30.40.64
20.30.40.127
20.30.40.01000000
20.30.40.01111111
20.30.40.0 / 26
20.30.40.64 / 26

10.10.10.10/12 — 1/2, 1/2 HID.

4

10.0000 - - - - - - - - - - - - - - - - 10.7.253.255

10.0.0.0 to 10.7.255.255

10.8.0.0 to 10.15.255.255

10.0.0.0

10.15.255.255

$$\frac{\cancel{16}^{8} \times 256 \times 256}{\cancel{2}}$$

$$\frac{16}{4} \times 256 \times 256$$

4

(0 - 3)

(4 - 7)

(8 - 11)

(12 - 15)

10.0000 0 | 10.0000 1

12 + 1

10.0.0.0/13 to 10.7.255.255/13 | 10.8.0.0/13 to 10.15.255.255/13

0 - 3

$\boxed{10.0.0.0/14 \text{ to } 10.3.255.255/14}$

4 - 7

$\boxed{10.4.0.0/14 \text{ to } 10.7.255.255/14}$

12 = NID.

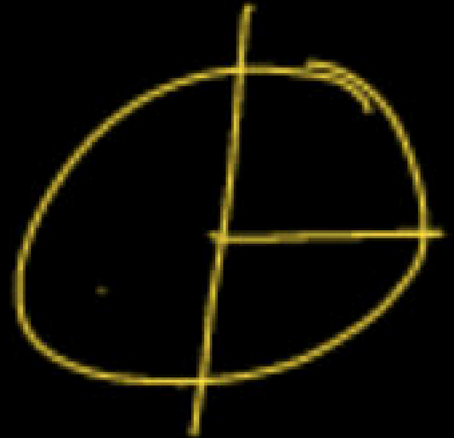
$$\underset{\text{NID}}{12} + \underset{\text{SID}}{2} = 14$$

00, 01, 10, 11

All 0's → NID

All 1's HID

10.10.10.10/12



range - ?

10.0.0.0 to 10.15.255.255

(0-7)
(8-11)
(12-15)

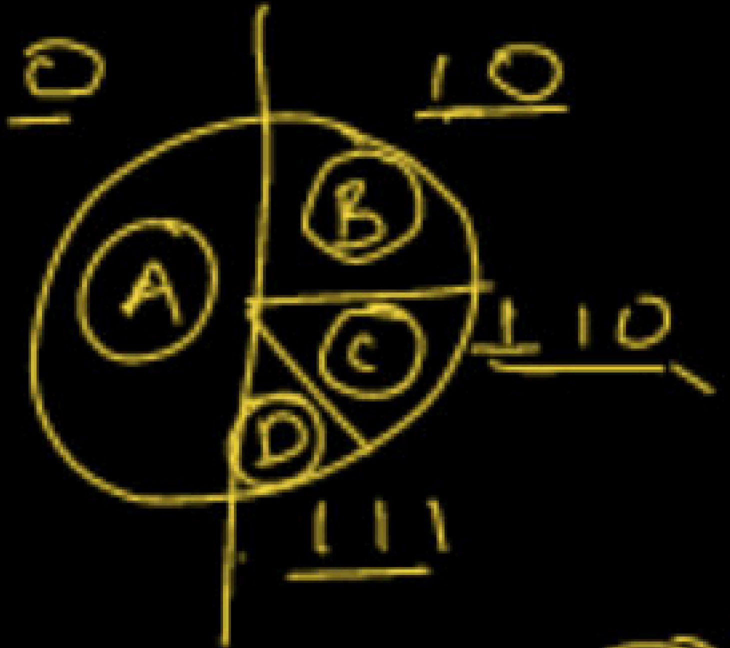
12+2 = 14

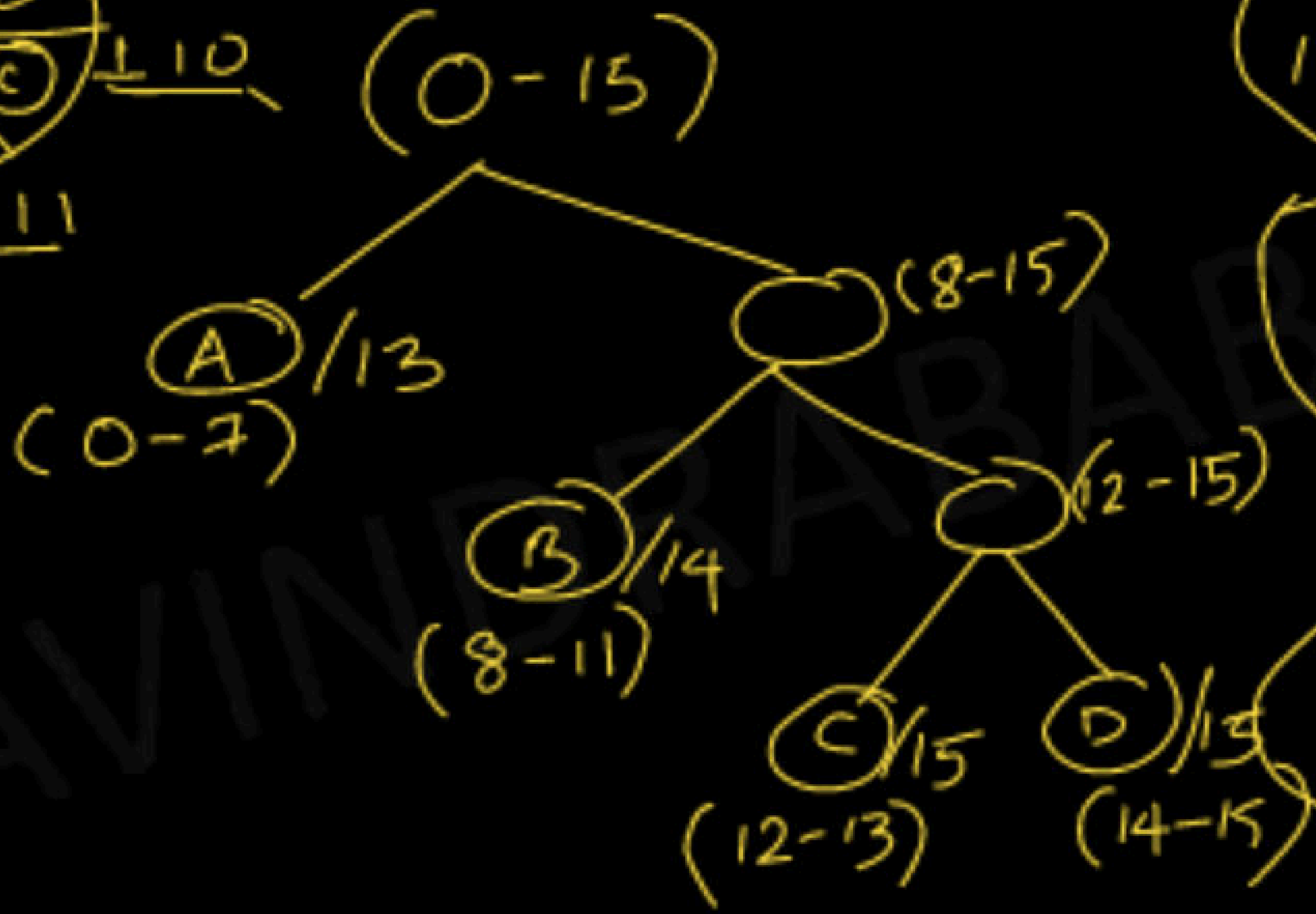
10.0.0.0/13 to 10.7.255.255/13

10.8.0.0/14 to 10.11.255.255/14

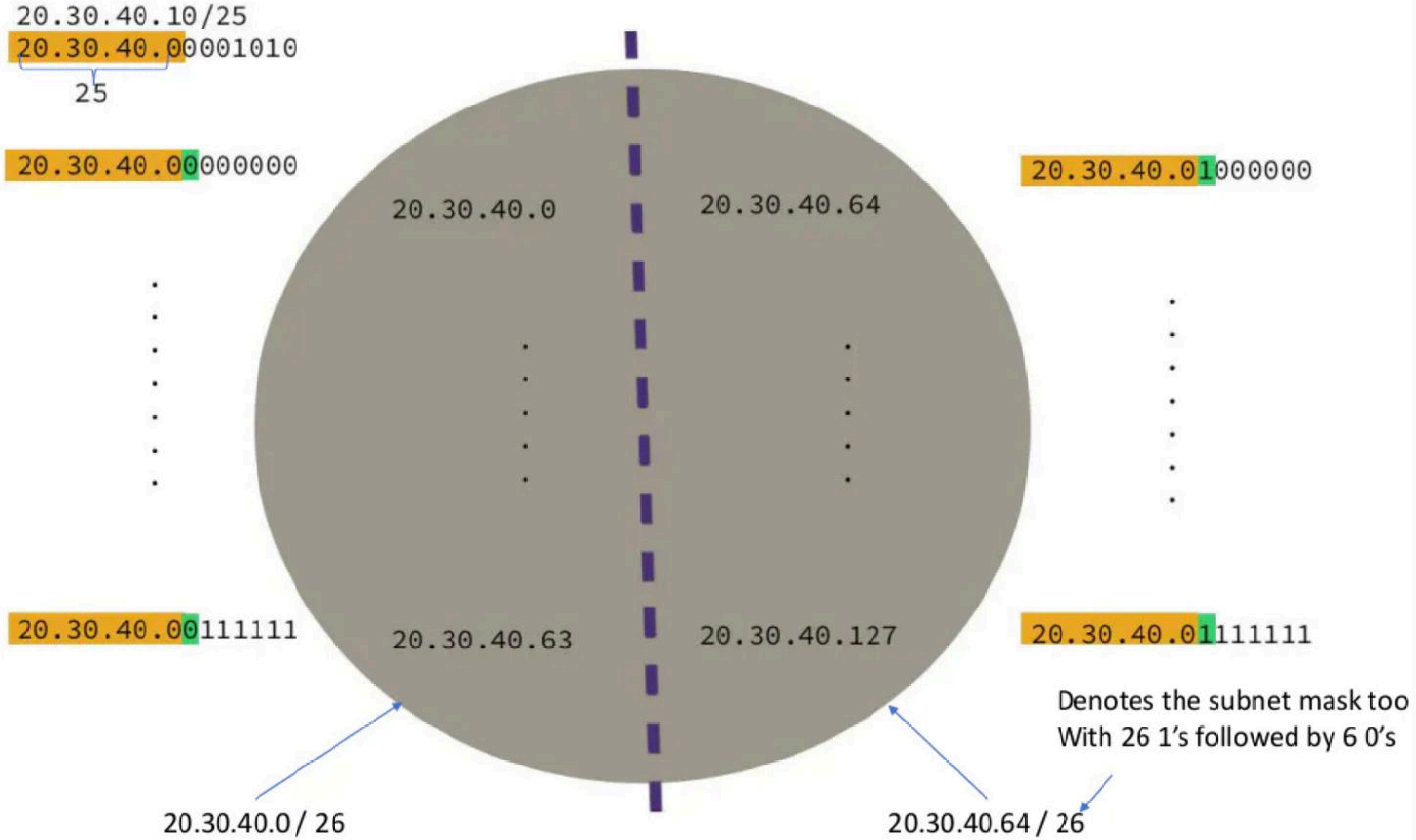
10.12.0.0/14 to 10.15.255.255/14

/0
/1
/2





12

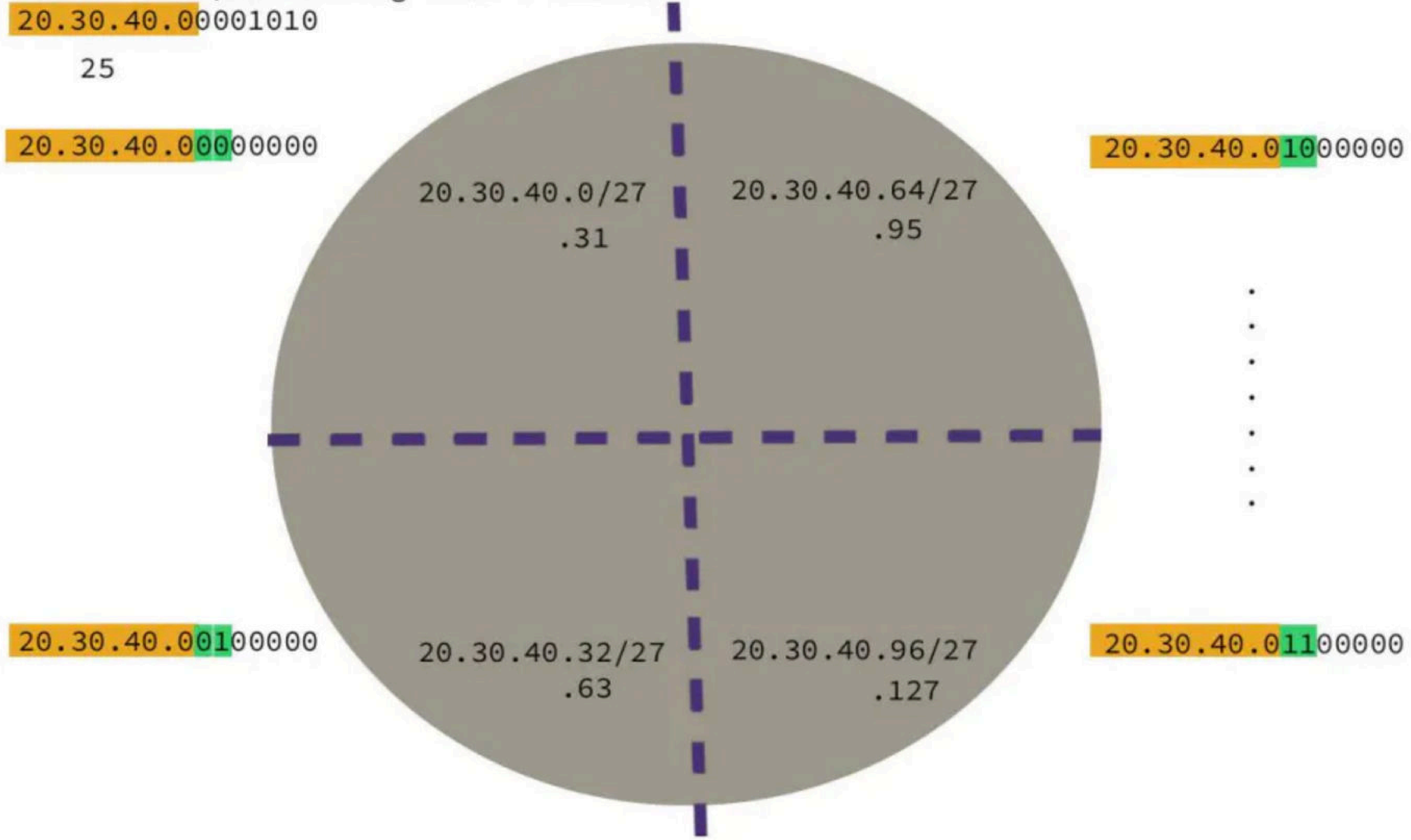


20.30.40.10/25
20.30.40.00001010
25
20.30.40.00000000
20.30.40.0
20.30.40.64
20.30.40.01000000
20.30.40.00111111
20.30.40.63
20.30.40.127
20.30.40.01111111
Denotes the subnet mask too
With 26 1's followed by 6 0's
20.30.40.0 / 26
20.30.40.64 / 26

20.30.40.10/25 ←Diving into 4 subnets

20.30.40.00001010

25



FROM THE PREVIOUS 2 EXAMPLES

DIVING THE NETWORK INTO 2 SUBNETS:

/25 -> 2^7

/26 -> 2^6 /26 -> 2^6

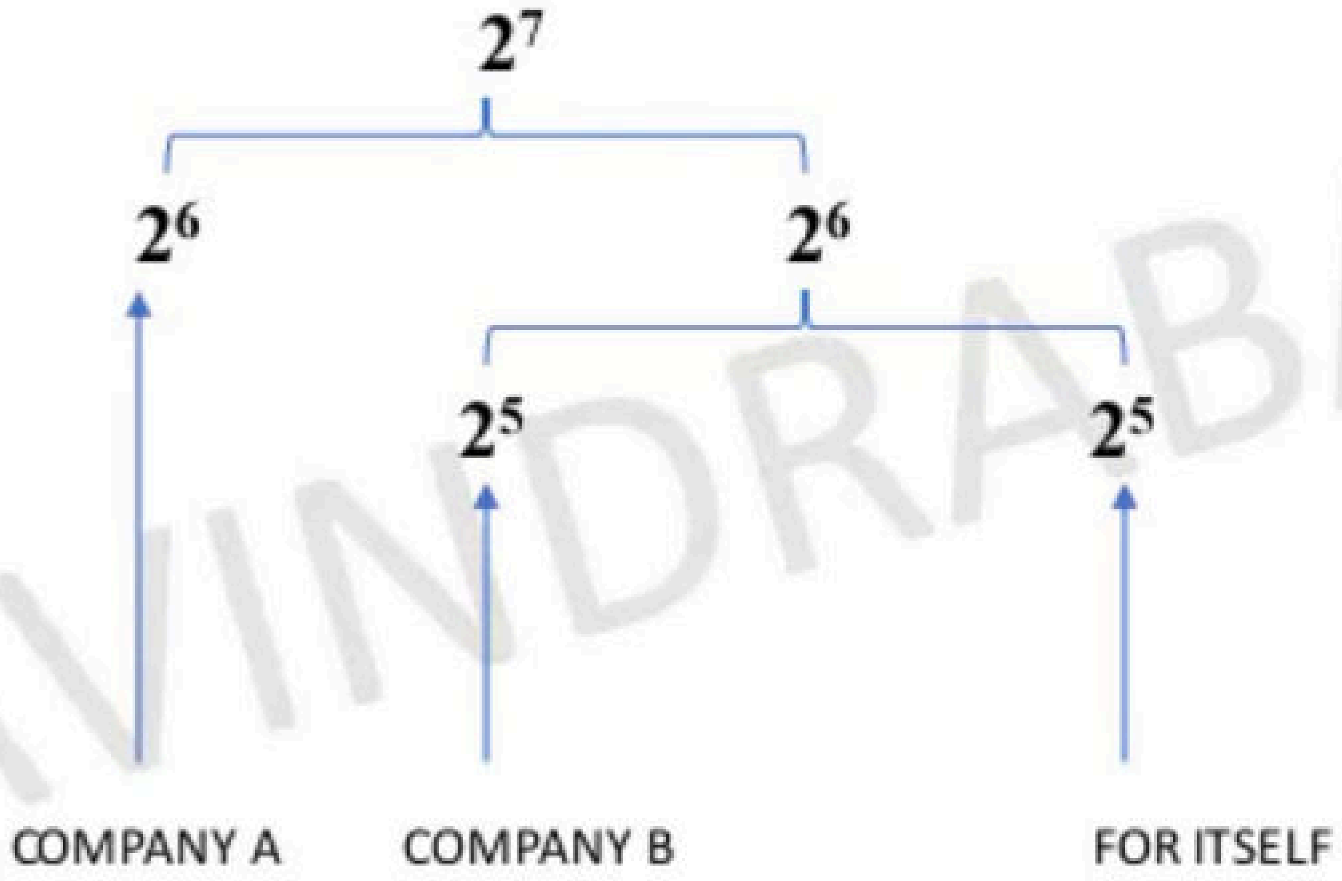
DIVING THE NETWORK INTO 4 SUBNETS:

/25 -> 2^7

/27 -> 2^5 /27 -> 2^5 /27 -> 2^5 /27 -> 2^5

VLSM IN CIDR BLOCKS

LET US ASSUME A SCENARIO IN WHICH AN ISP HAS A BLOCK 20.30.40.10/25 AND THEY WISH TO DIVIDE IT INTO 3 SUBNETS IN A FOLLOWING WAY:



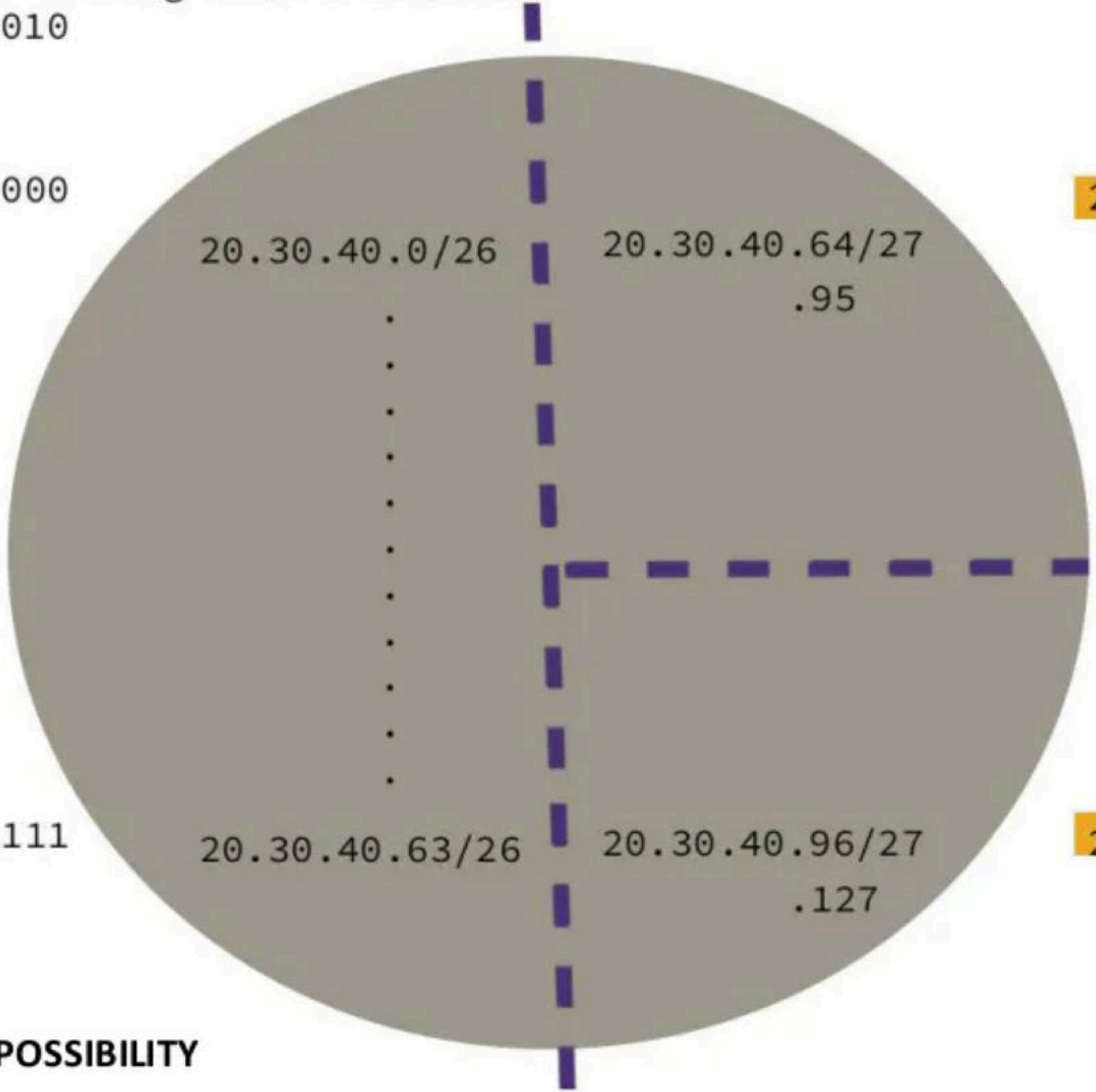
20.30.40.10/25 Diving into 3 subnets

20.30.40.00001010

25

20.30.40.00000000

20.30.40.00111111

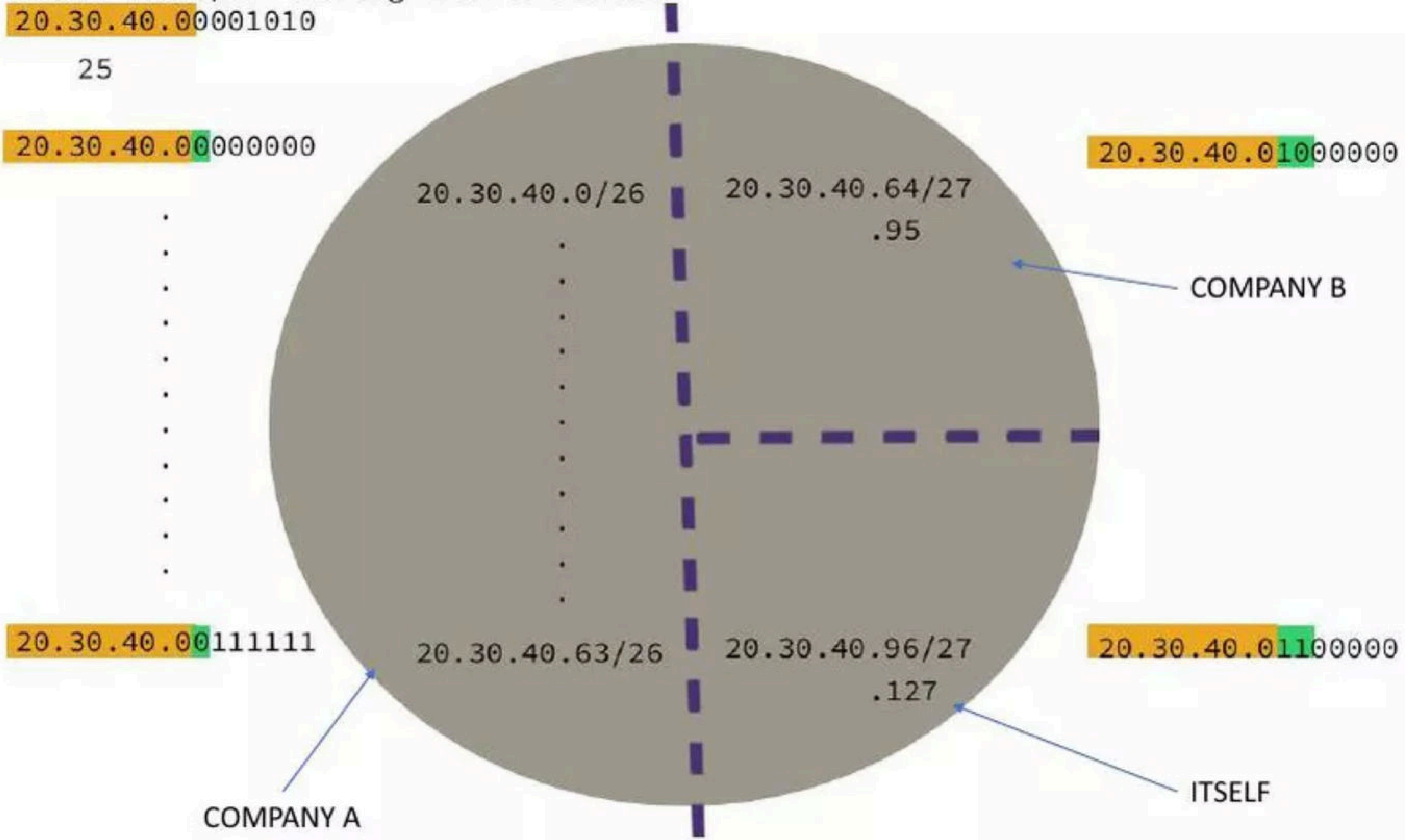


20.30.40.01000000

20.30.40.01100000

THIS COULD BE ONE POSSIBILITY

20.30.40.10/25 Diving into 3 subnets



20.30.40.10/25 ← Diving into 3 subnets

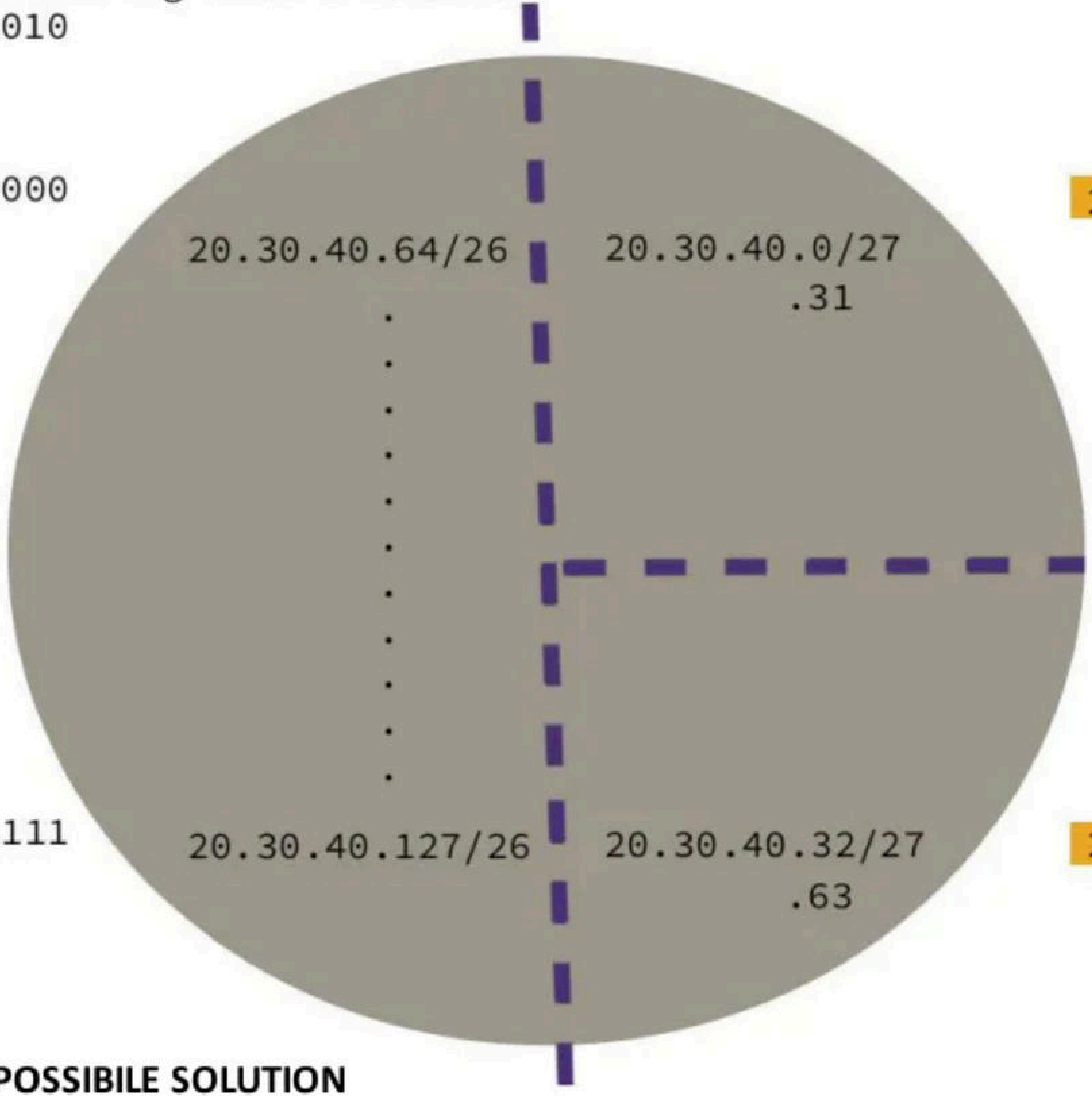
20.30.40.00001010

25

20.30.40.01000000

.
.
.
.
.
.
.
.
.
.
.

20.30.40.01111111



20.30.40.00000000

20.30.40.00100000

THIS COULD BE ANOTHER POSSIBILE SOLUTION

20.30.40.10/25 Diving into 3 subnets

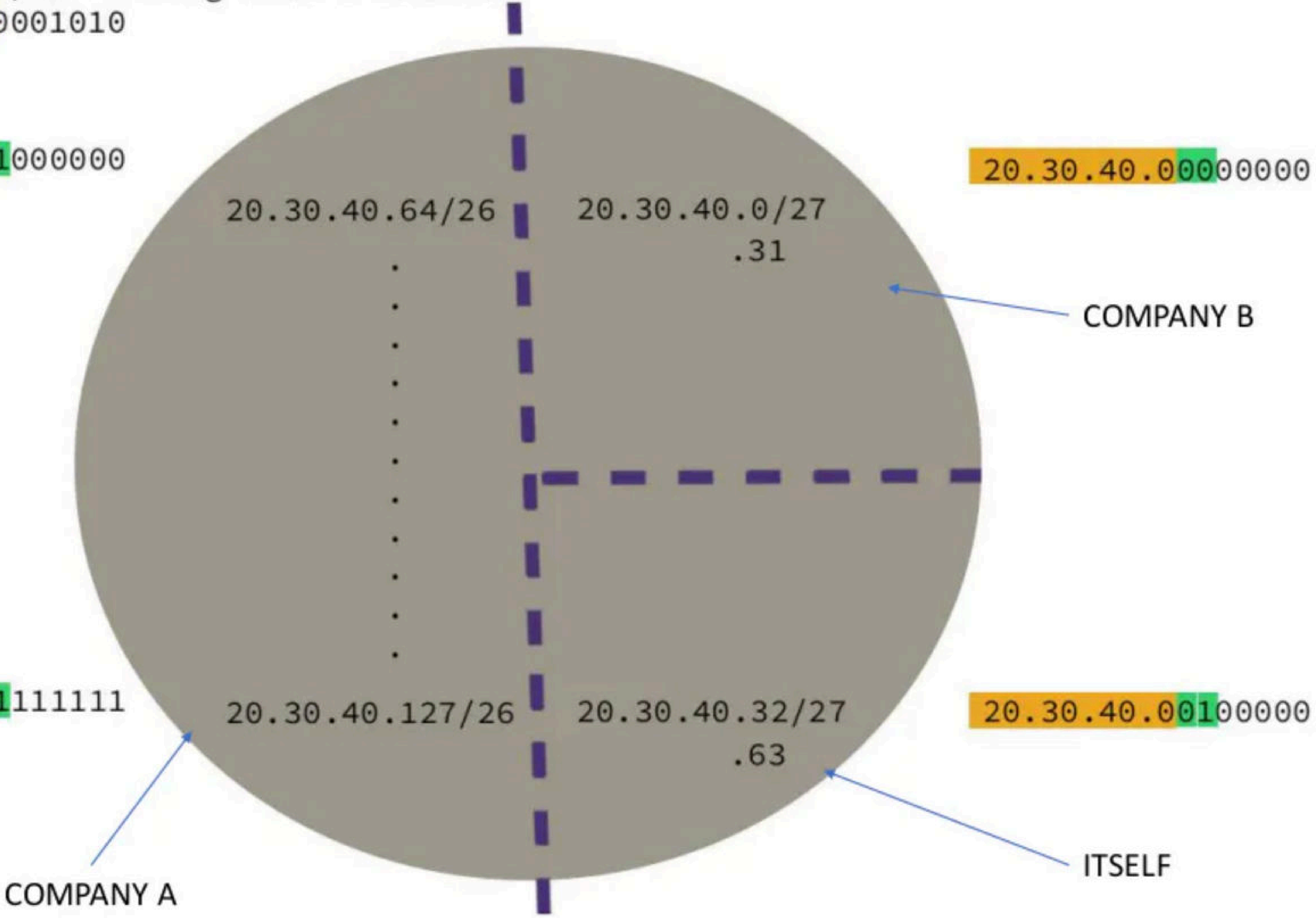
20.30.40.00001010

25

20.30.40.01000000

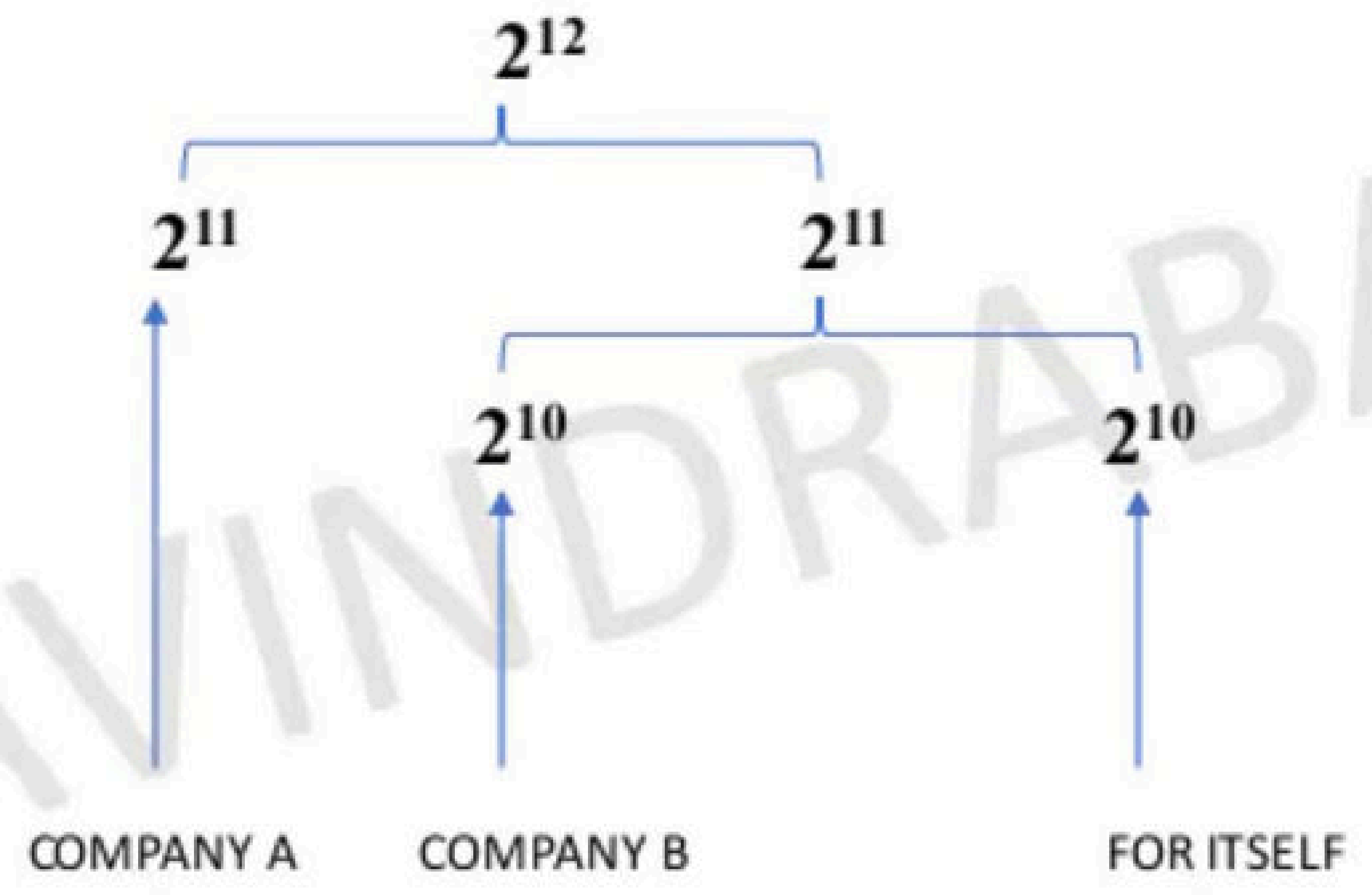
.
.
.
.
.
.
.
.
.
.
.

20.30.40.01111111



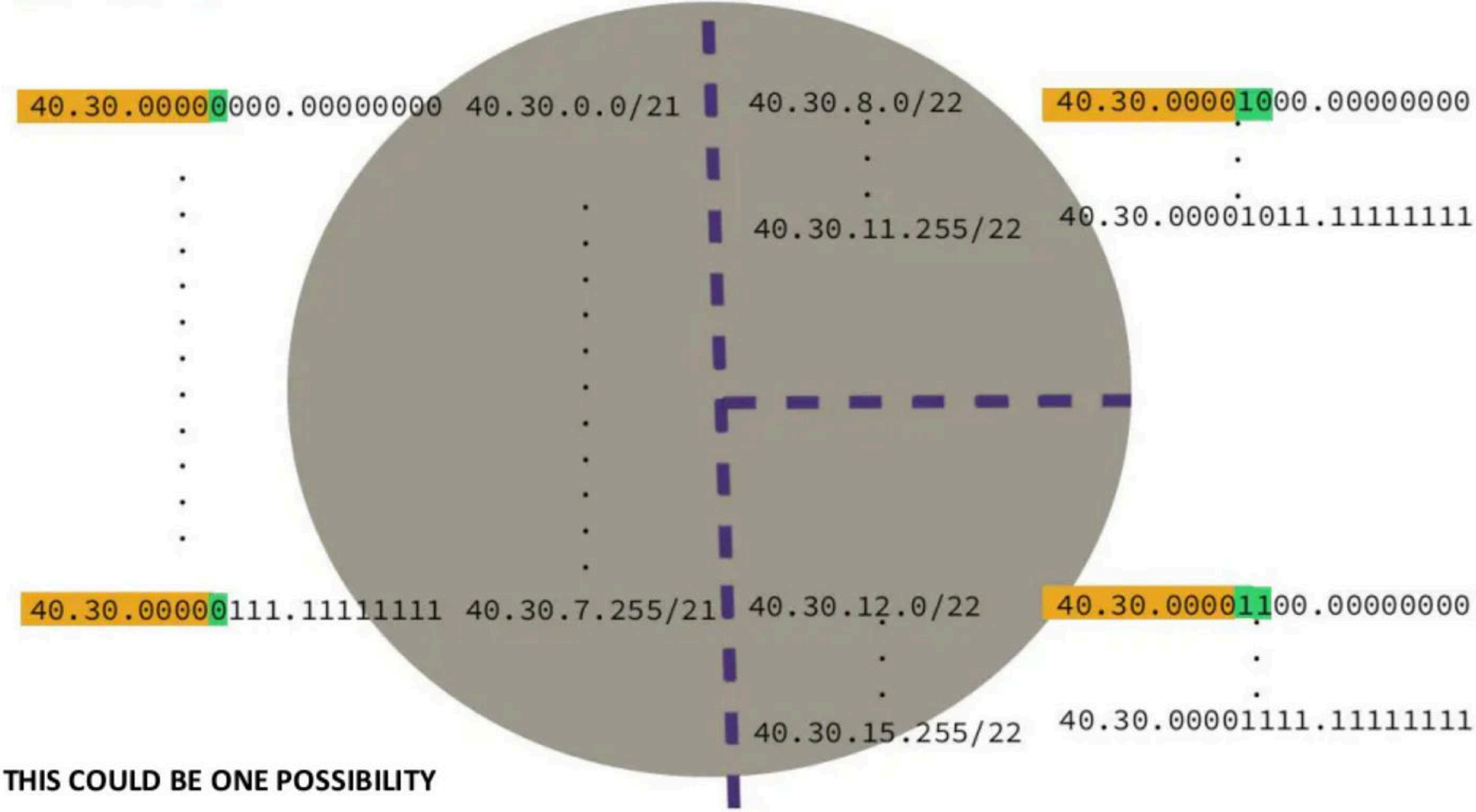
LET'S CHECK OUT ONE MORE EXAMPLE

LET US ASSUME A SCENARIO IN WHICH AN ISP HAS A BLOCK 40.30.10.10/20 AND THEY WISH TO DIVIDE IT INTO 3 SUBNETS IN A FOLLOWING WAY:



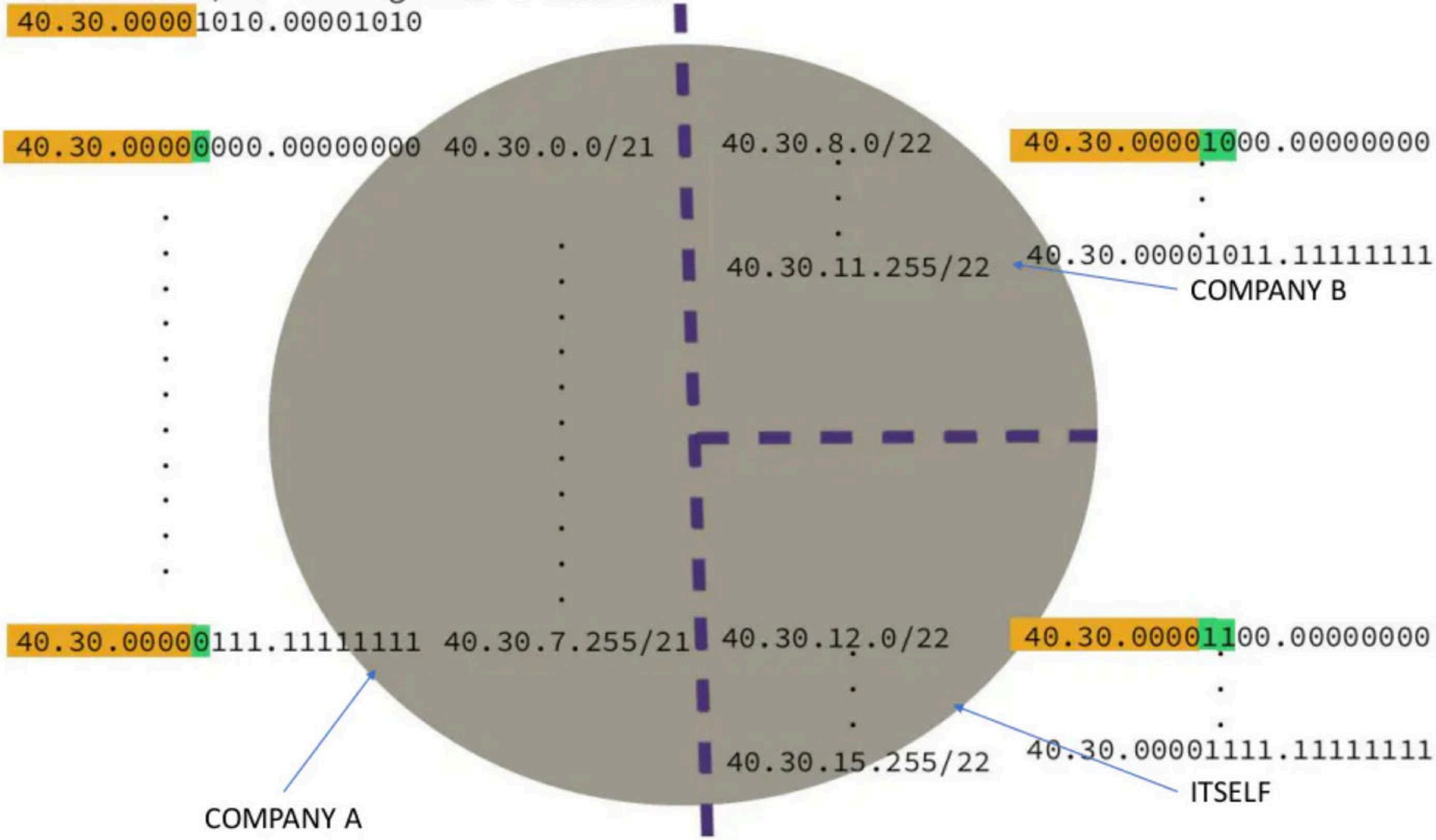
40.30.10.10/20 ← Diving into 3 subnets

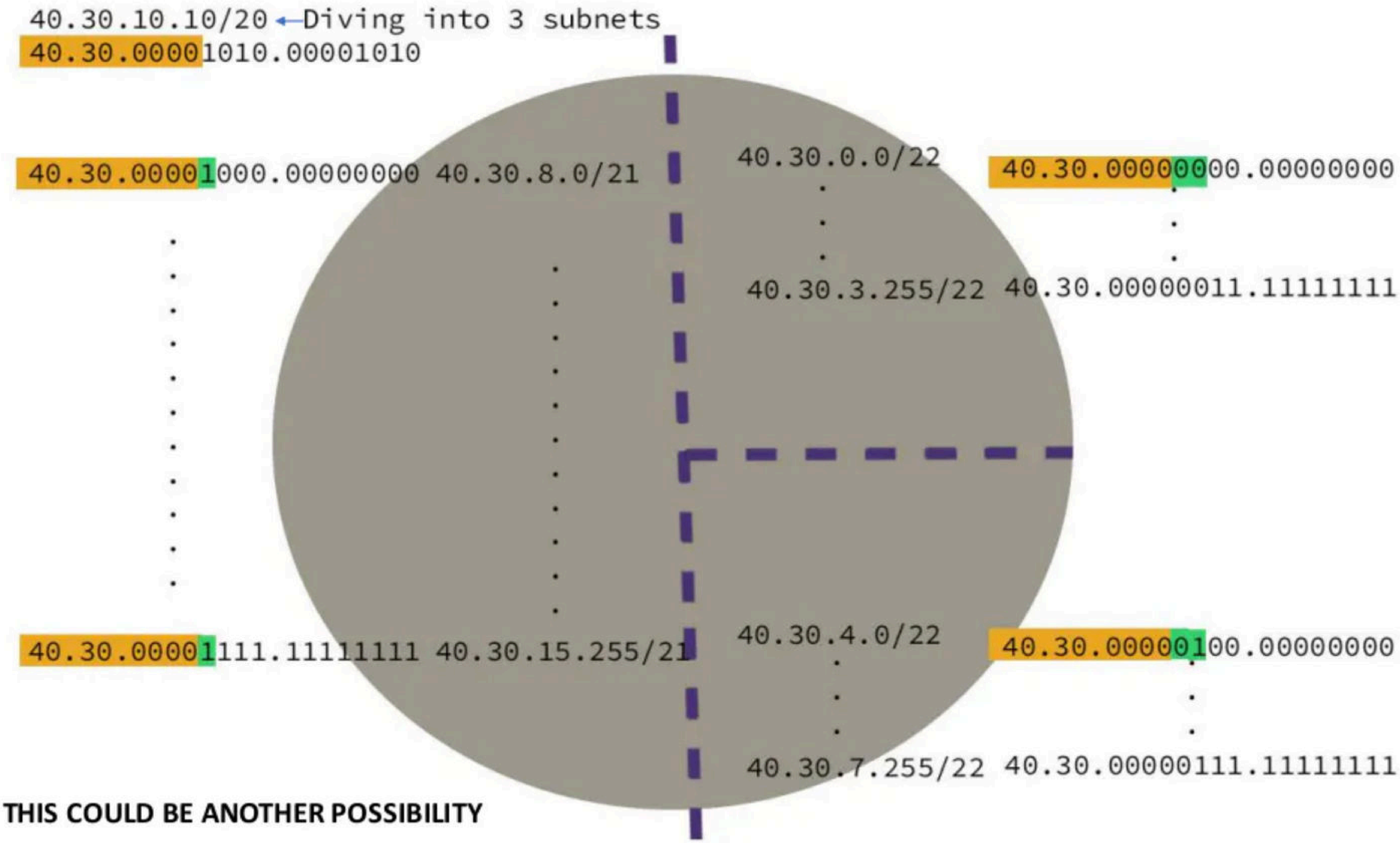
40.30.00001010.00001010



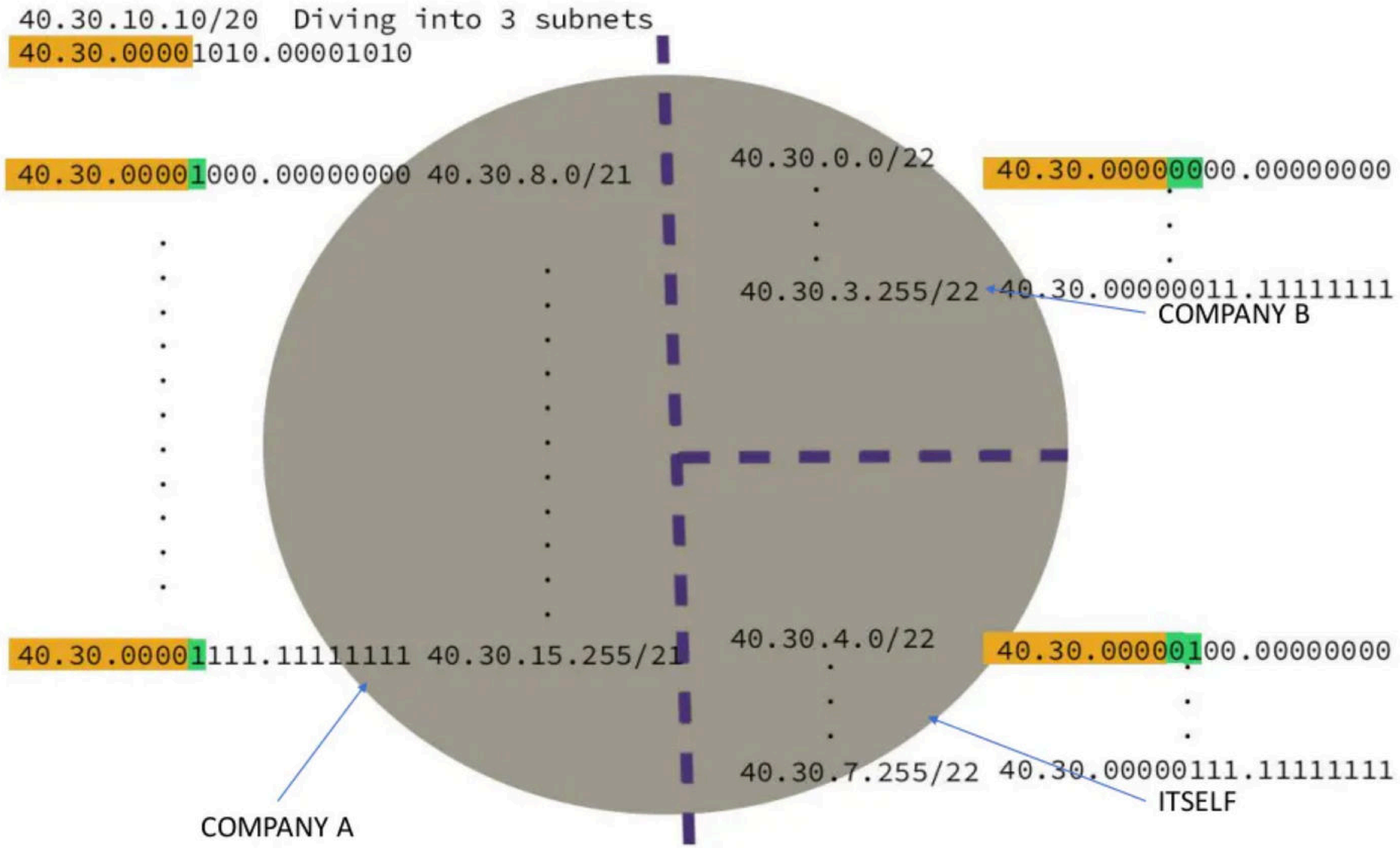
THIS COULD BE ONE POSSIBILITY

40.30.10.10/20 Diving into 3 subnets





THIS COULD BE ANOTHER POSSIBILITY



40.30.10.10/20 Diving into 3 subnets
40.30.00001010.00001010
40.30.00001000.00000000 40.30.8.0/21
40.30.00001111.11111111 40.30.15.255/21
COMPANY A
40.30.0.0/22
40.30.3.255/22
40.30.00000000.00000000
40.30.00000011.11111111
COMPANY B
40.30.4.0/22
40.30.7.255/22
40.30.00000100.00000000
40.30.00000111.11111111
ITSELF

Computer Networks

Supernetting OR Aggregation

In subnetting, a single network is divided into multiple smaller subnetworks.

In Supernetting/Aggregation, multiple networks are combined into a bigger network termed as a Super network or Supernet.

Rules for Aggregation :

1.All the Networks should be contiguous.

2.The block size of every networks should be equal and must be in form of 2^n.

3.First Network id should be exactly divisible by whole size of supernet.

Points to be checked before aggregation :

1.All the Networks should be contiguous.

2.The block size of every networks should be equal and must be in form of 2^n.

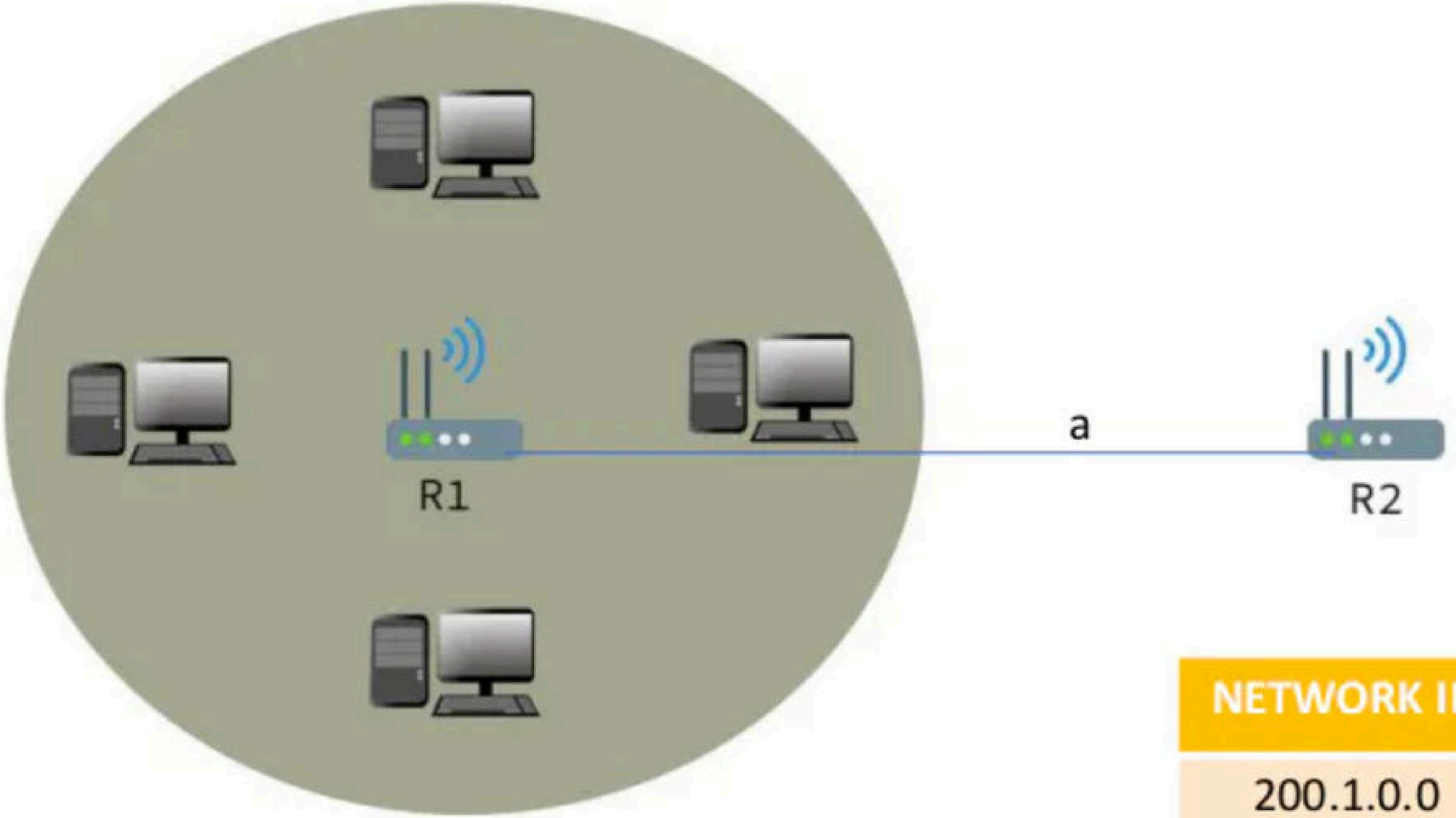
3.First Network id should be exactly divisible by whole size of supernet.

Example:

200.1.0.0 / 24
200.1.1.0 / 24
200.1.2.0 / 24
200.1.3.0 / 24

Total size of the supernet = 4 X $2^8 = 2^{10}$
Diving the First Network id with 2^{10}
Has 0 in last 10 bits. Therefore Divisible.

All are satisfied, hence we can move forward with aggregation.



Instead of having 4 entries
We want just one entry in R2
having a single supernet id

ROUTING TABLE AT R2

NETWORK ID	SUBNET MASK	INTERFACE
200.1.0.0	255.255.255.0	a
200.1.1.0	255.255.255.0	a
200.1.2.0	255.255.255.0	a
200.1.3.0	255.255.255.0	a

Finding the Supernet mask:

It is 32 bit number

In Supernetting, the Number of 1's represent the Fixed part
And Number of 0's represent the Variable part

200.1.00000000.00000000
200.1.00000001.00000000
200.1.00000010.00000000
200.1.00000011.00000000

Fixed Variable

Replacing the fixed part with 1's and variable with 0's
We get the supernet mask,

255.255.252.0

Finding the Supernet mask:

It is 32 bit number

In Supernetting, the Number of 1's represent the Fixed part
And Number of 0's represent the Variable part

Fixed	Variable
200.1.000000	00.00000000
200.1.000000	01.00000000
200.1.000000	10.00000000
200.1.000000	11.00000000

Replacing the fixed part with 1's and variable with 0's
We get the supernet mask,

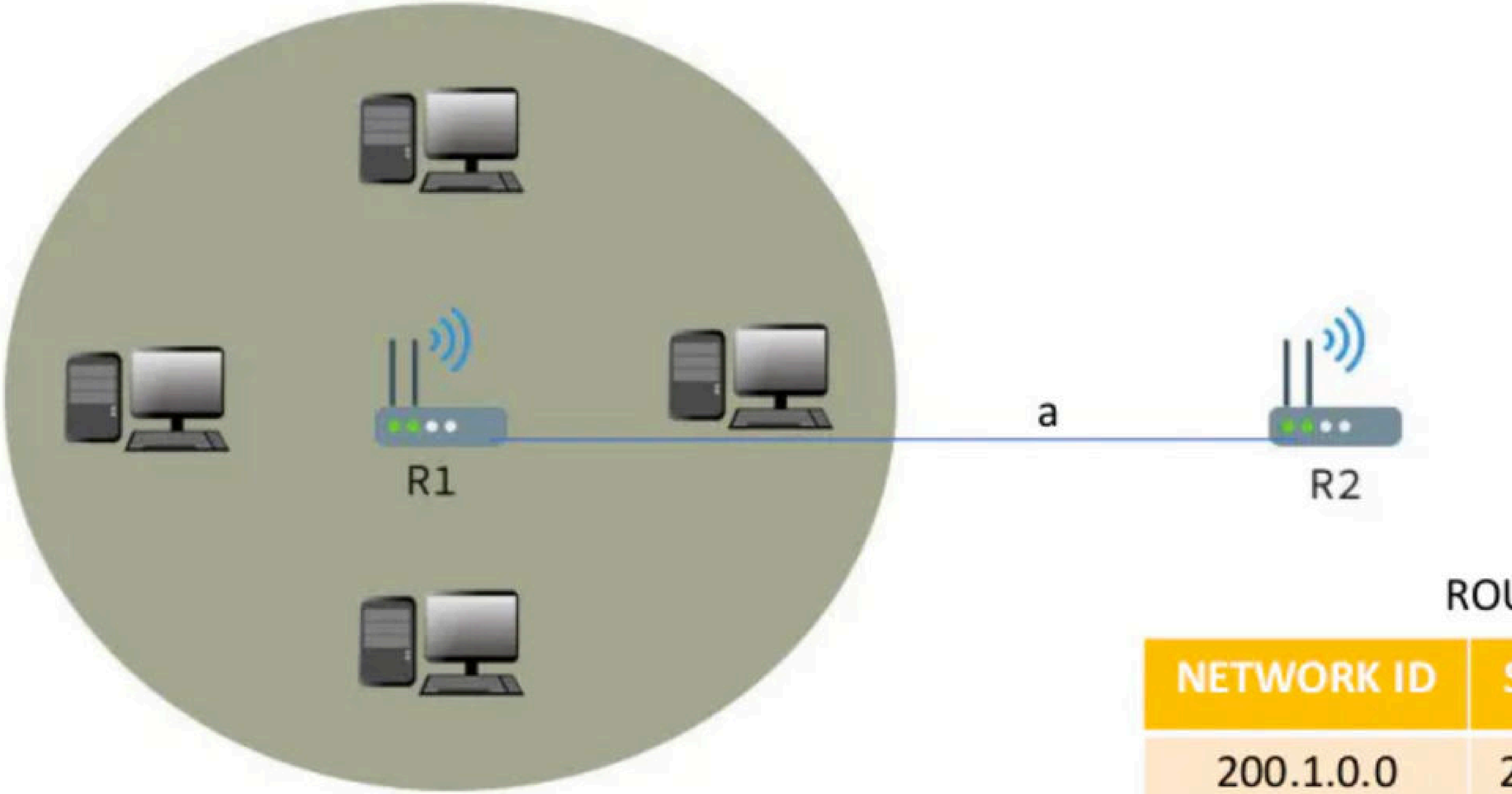
255.255.252.0

SHORTCUT TO FIND THE SUPERNET ID AND SUPERNET MASK

If the network id's follow the Rules then,
The first IP address is always the Supernet id.

Add the network size of all the networks given
The total denotes the host if part of the subnet mask .
From the example:
Network size of all = $2^{8} + 2^{8} + 2^{8} + 2^{8} = 2^{10}$
Therefore the subnet mask contains
10 bits in host id part.
And the network id part will contain 22 bits.

That is what is the subnet mask



Revised Routing table

ROUTING TABLE AT R2

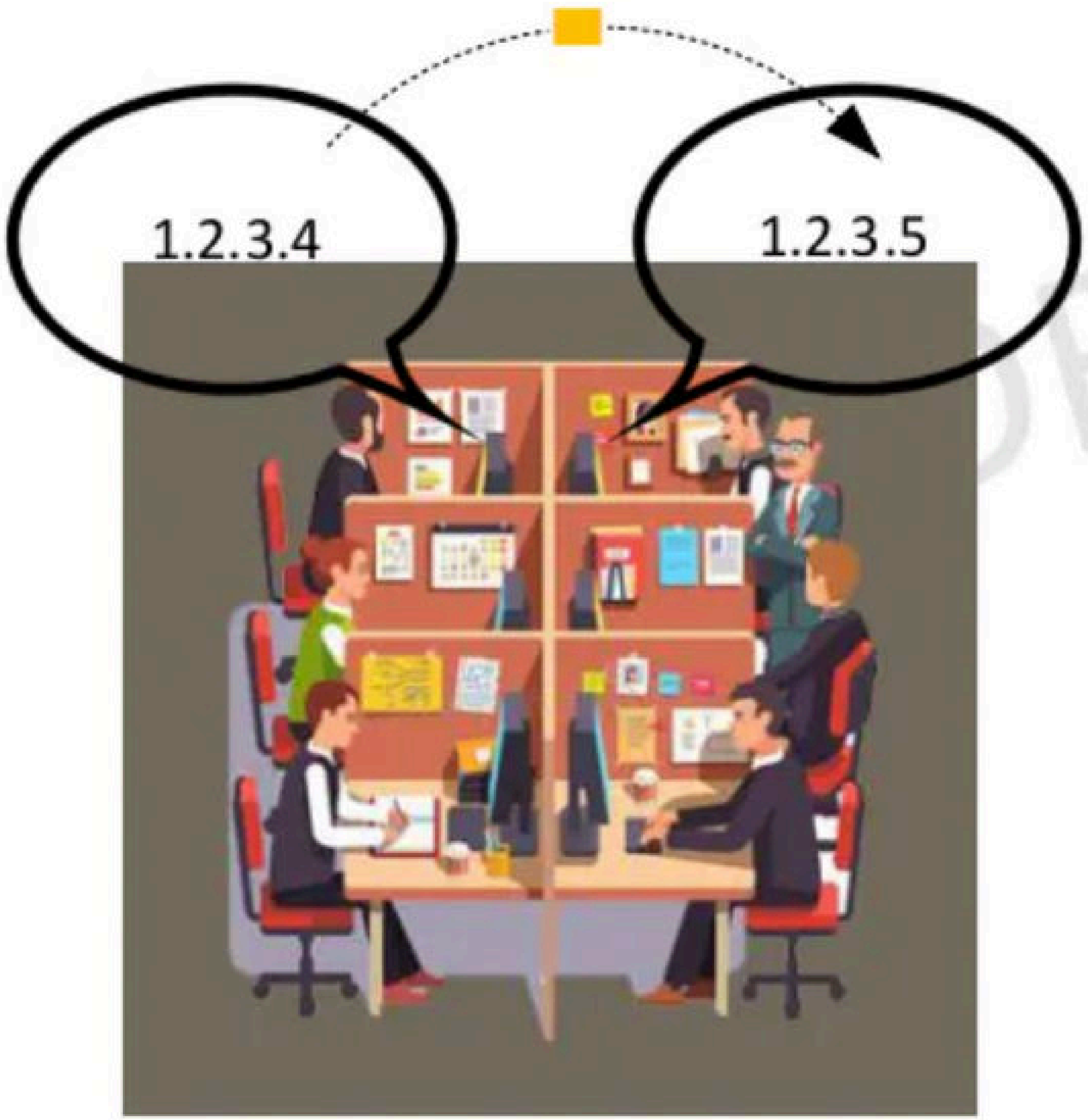
NETWORK ID	SUBNET MASK	INTERFACE
200.1.0.0	255.255.252.0	a

Note : In case of R1, we need all the 4 entries

Computer Networks

Private IP addresses

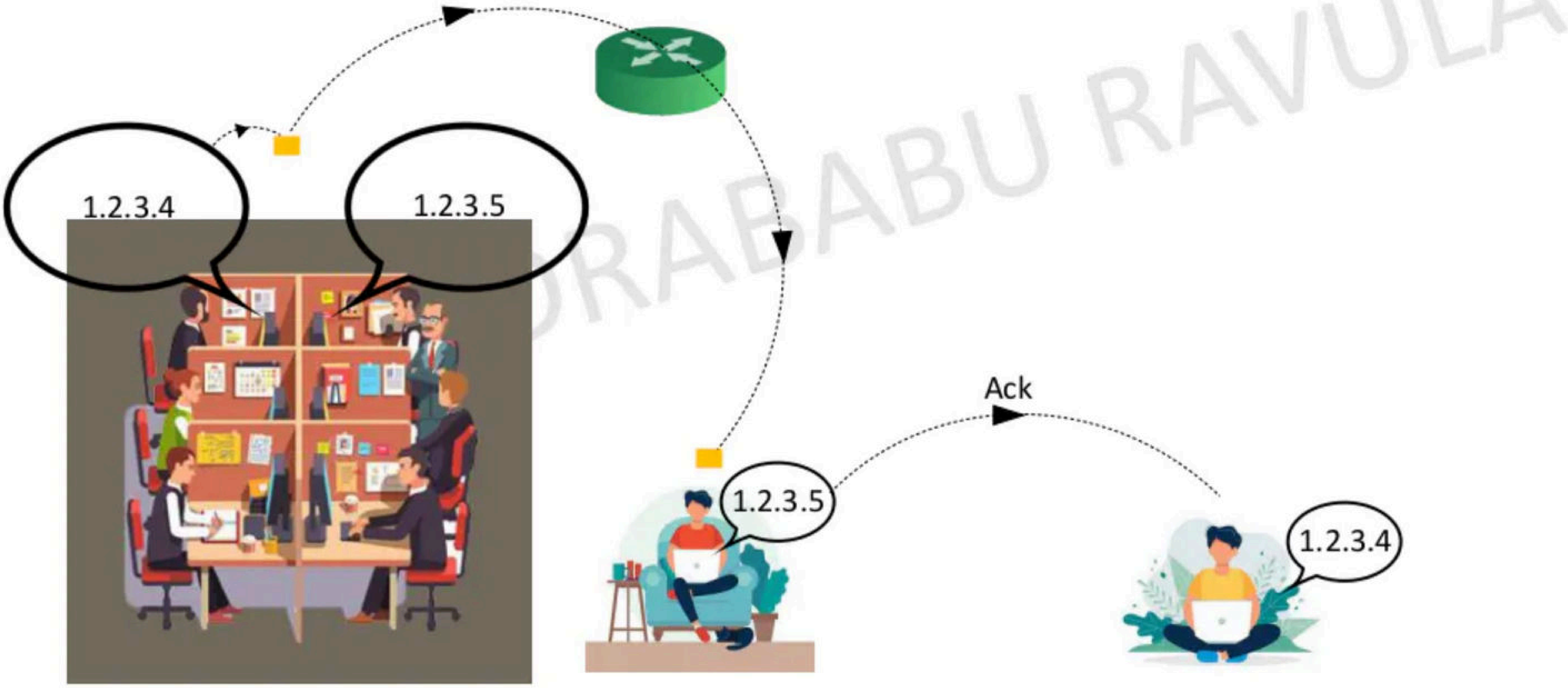
Imagine there is a software company, in which there are 6 PCs and the company has 2 criteria:
1.)No internet access / No communication with the outside world
2.)All the employees can communicate between each other



For this you can set a TCP/IP network and
Assign any IP address to the hosts

This allocation remains safe only till the packet is sent within the network

Suppose internet was accessed and the packet sent
from 1.2.3.4 To 1.2.3.5 escaped onto the internet



IANA has come to a solution to the problem and introduced **Private IP address**
So if one wishes to use Ips only within the Private Network, Private IPs should be assigned and not the Public IPs

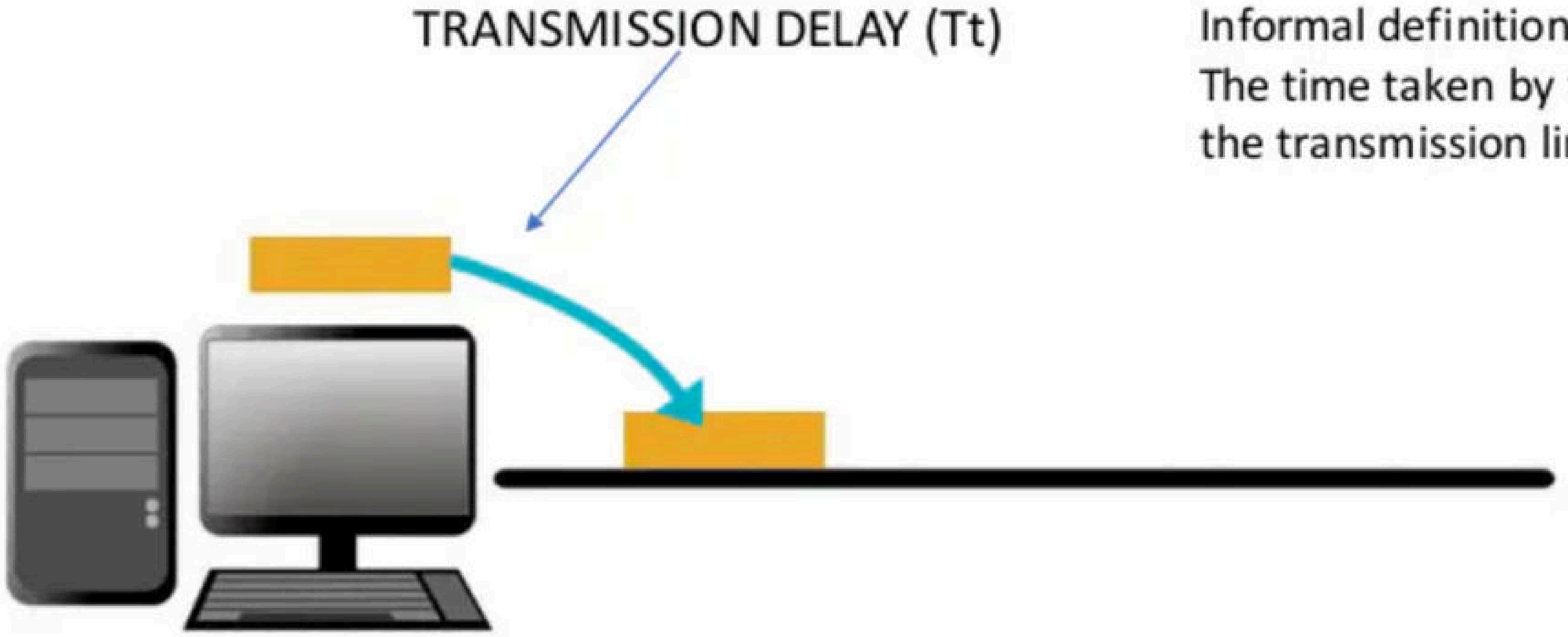
PRIVATE IP RANGE

- 10.0.0.0 to 10.255.255.255
- 172.16.0.0 to 172.31.255.255
- 192.168.0.0 to 192.168.255.255

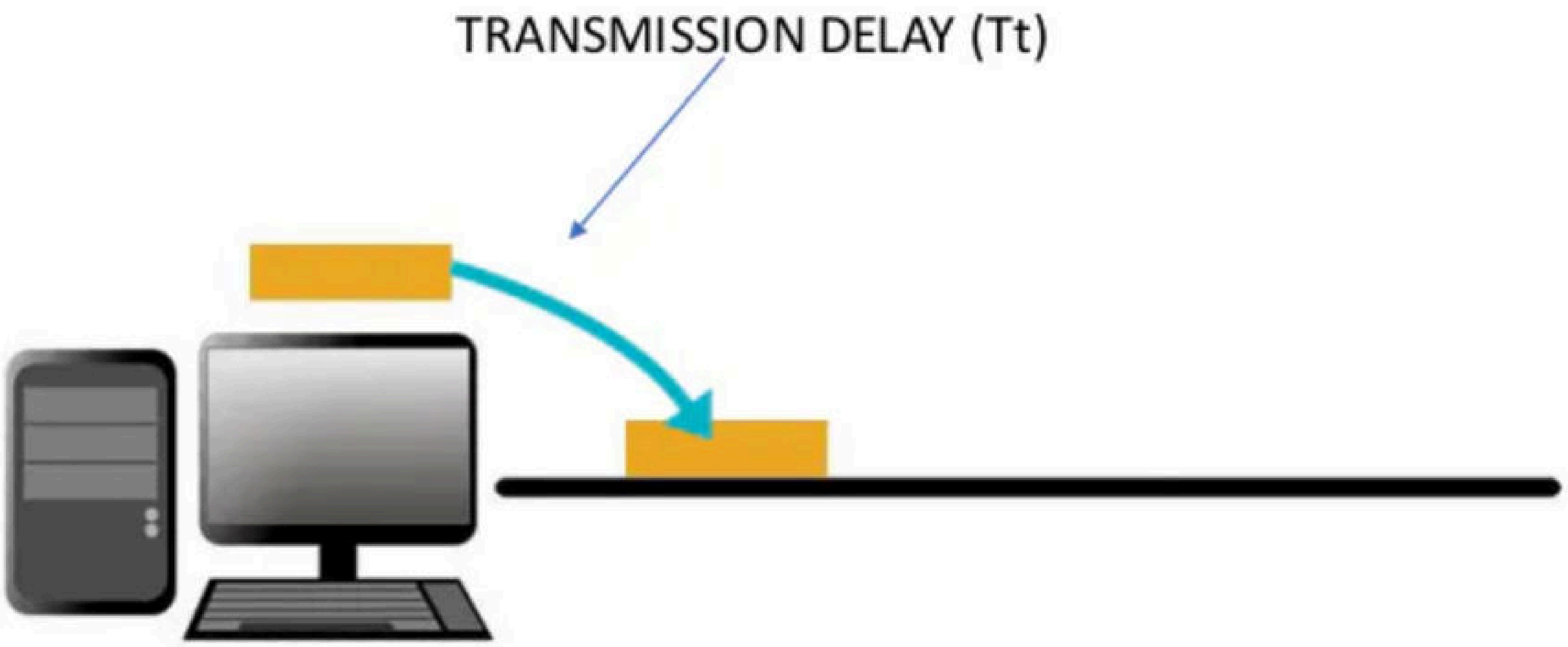
So, due to this what happens is even if the packet goes outside the private network the router discards it because It was send from a Private IP.

Computer Networks

Delays in Computer Networks



Informal definition:
The time taken by the host to put the data packet into the transmission link is called Transmission Delay



If Bandwidth is 1 bps
And data to be sent is 10 bits
The Tt would be 10 sec

So, in general we can say that,

$$Tt = \frac{\text{Length of data (bits)}}{\text{Bandwidth(bps)}} = \frac{L}{B} \text{ sec}$$

Consider these 2 examples

Example 1 : If L = 1000 bits and BW = 1 Kbps

Then

$$Tt = \frac{L}{B} = \frac{1000}{1000} = 1 \text{ sec}$$

Example 2 : If L = 1 KB and BW = 1 Kbps

Then

$$Tt = \frac{L}{B} = \frac{1024}{1000} = 1.024 \text{ sec}$$

Consider these 2 examples

Example 1 : If L = 1000 bits and BW = 1 Kbps
Then

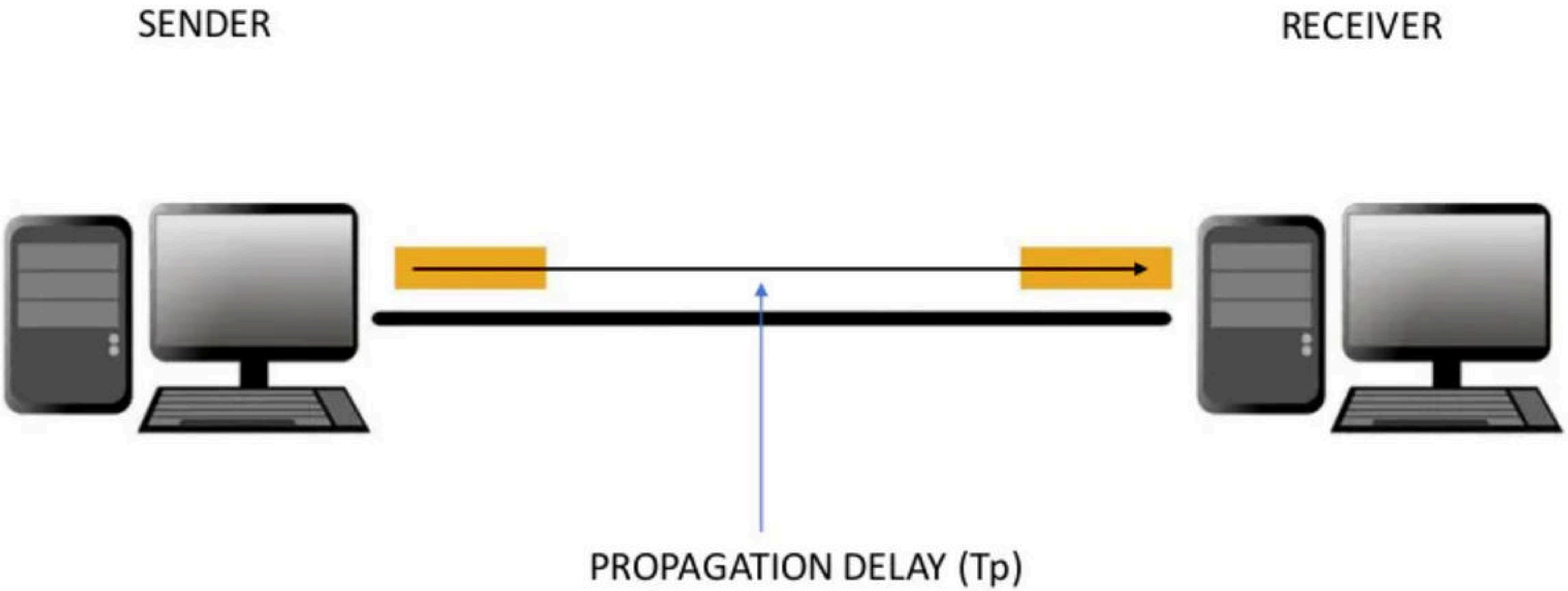
$$Tt = \frac{L}{B} = \frac{1000}{1000} = 1 \text{ sec}$$

Example 2 : If L = 1 KB and BW = 1 Kbps
Then

$$Tt = \frac{L}{B} = \frac{1024}{1000} = 1.024 \text{ sec}$$

REMEMBER

Whenever data has **K**, K= 1024	Whenever Bandwidth has **K**, K= 1000 = 10^3
Whenever data has **M**, M=1024 X 1024	Whenever Bandwidth has **M**, M= 10^6
Whenever data has **G**, G=1024 X 1024 X 1024	Whenever Bandwidth has **G**, G= 10^9



Time taken for one bit to travel from sender to receiver end of the link is called as propagation delay.

$$Tp = \frac{\text{distance of the link (d)}}{\text{Velocity(v)}} \text{ seconds}$$

Generally the link nowadays is optical fibre and
We know that the speed at which the light travels
is 3 X 10^8 m/s.

In optical fibre the light travels with 70% of the original speed of light(3 X 10^8 m/s)
=>v = 0.7 X 3 X 10^8 m/s = 2.1 X 10^8 m/s

Generally the velocity will be given in the question but, in questions where velocity is not given, we need to take it as 2.1 X 10^8 m/s

Example: d = 2.1 km and v = 2.1 X 10^8 m/s

$$Tp = \frac{\text{distance of the link (d)}}{\text{Velocity(v)}}$$

$$Tp = \frac{2.1}{2.1 \times 10^8}$$

$$= \frac{2.1 \times 1000}{2.1 \times 10^8}$$

$$Tp = 10^{-5} \text{ sec}$$

Converting 10^{-5} sec to µsec

We know that µ is 10^{-6}

Step 1 :Divide and multiply by 10^{-6}

$$10^{-5} \times \frac{10^{-6}}{10^{-6}}$$

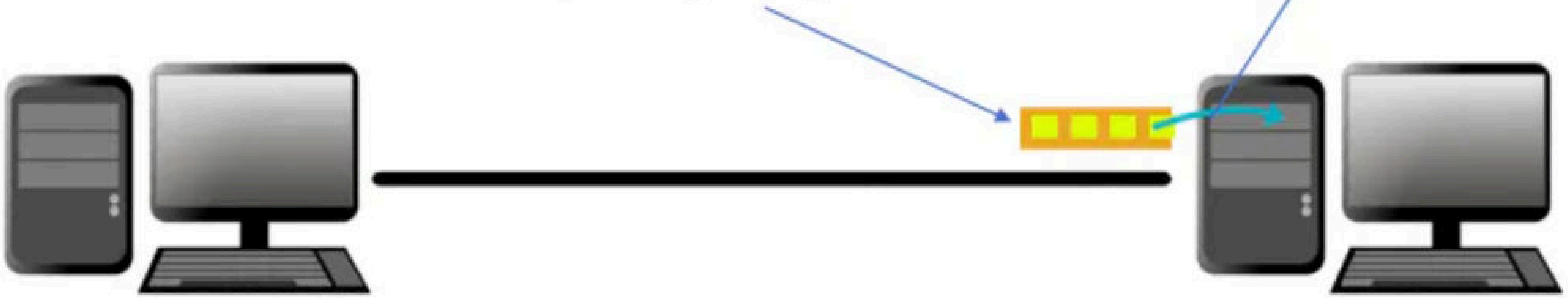
Step 2 :Combine

$$10^{-5} \times \frac{10^{-6}}{10^{-6}} = 1 \times 10^{-6}$$

= 1 µsec

Time taken by the processor to process the data packet is called as **processing delay**

Time spent by the data packet waiting in the queue before it is taken for execution is called as **queuing delay**.



NOTE :THESE TWO DELAYS DEPEND UPON THE TYPE OF PROCESSOR USED AND CANNOT BE EVALUATED AND THEREFORE ARE CONSIDERED AS ZERO IN ALMOST ALL NUMERICALS