Subnetting

Figure 19.13 Netid and hostid

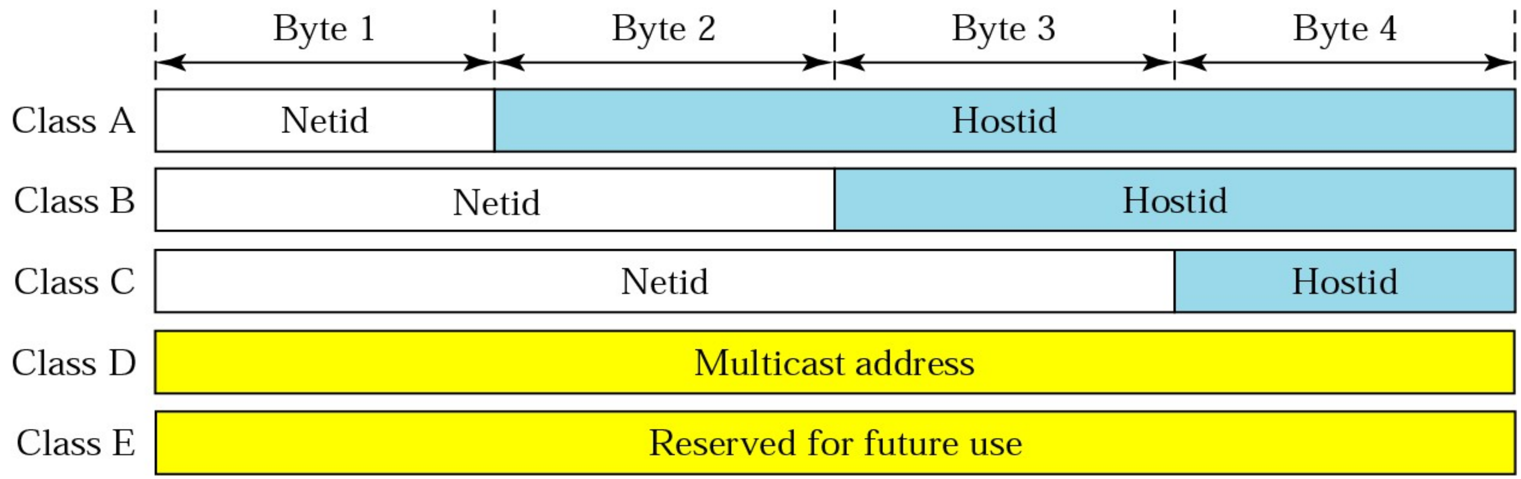


Table 19.1 Default masks

Class	*In Binary*	*In Dotted-Decimal*	*Using Slash*
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 111111111 11111111 00000000	255.255.255.0	/24

SUBNETTING:
Dividing the large network into multiple small NETWORK

192.168.10.0/24

- Class C IP
- /24-network id(24 1’s)
- Default subnet mask is 255.255.255.0
- 11111111.11111111.11111111.00000000
- 1-network id; 0- host id

192.168.10.0/25

- Class C IP
- /25-network id(25 1’s).
- 11111111.11111111.11111111.10000000(borrow 1 from the next octet)
- Default subnet mask is 255.255.255.128
- 1-network id; 0- host id

Problem 1

- Subnet the ip 216.21.5.0 address into 30 hosts in each subnet

Network Ranges (Subnets)

216.21.5.0 – 216.21.5.31
216.21.5.32 – 216.21.5.63
216.21.5.64 – 216.21.5.95
216.21.5.96 – 216.21.5.127
216.21.5.128 – 216.21.5.159
and so on_

192.168.10.0/25

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
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	0	0	0	0

HOW TO FIND THE NUMBER OF NETWORK

2^n (Here, n indicated total number of bits borrowed from host)

$2^1 = 2$ (You can create only two networks)

192.168.10.0/25

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
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	0	0	0	0

HOW TO FIND THE NUMBER OF NETWORK

2^n (Here, n indicated total number of bits borrowed from host)

$2^1 = 2$ (You can create only two networks)

HOW TO FIND THE NUMBER OF IP ADDRESS ON EACH NETWORK

2^n (Here, n indicated total number of host bits)

$2^7 = 128$ (On each network you will have 128 IP Address)

192.168.10.0/25

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
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	0	0	0	0

HOW TO FIND THE NUMBER OF NETWORK

2^n (Here, n indicated total number of bits borrowed from host)

$2^1 = 2$ (You can create only two networks)

HOW TO FIND THE NUMBER OF IP ADDRESS ON EACH NETWORK

2^n (Here, n indicated total number of host bits)

$2^7 = 128$ (On each network you will have 128 IP Address)

HOW TO FIND THE NUMBER OF HOSTS IN EACH NETWORK

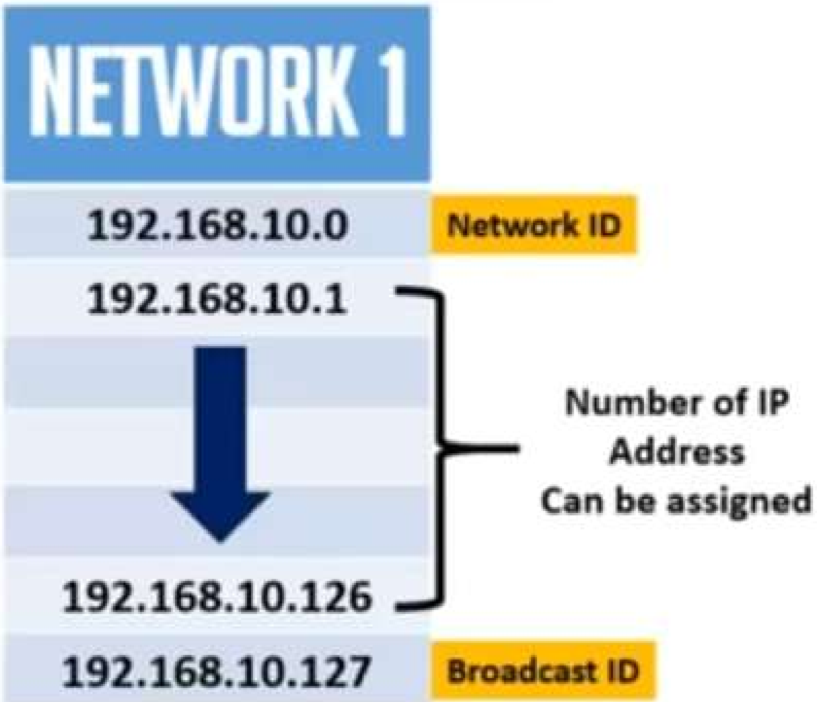
$2^n - 2$ (Here, n indicated total number of remaining host bits)

$2^7 - 2 = 126$ (You will have total 126 Host IP Address on each network)

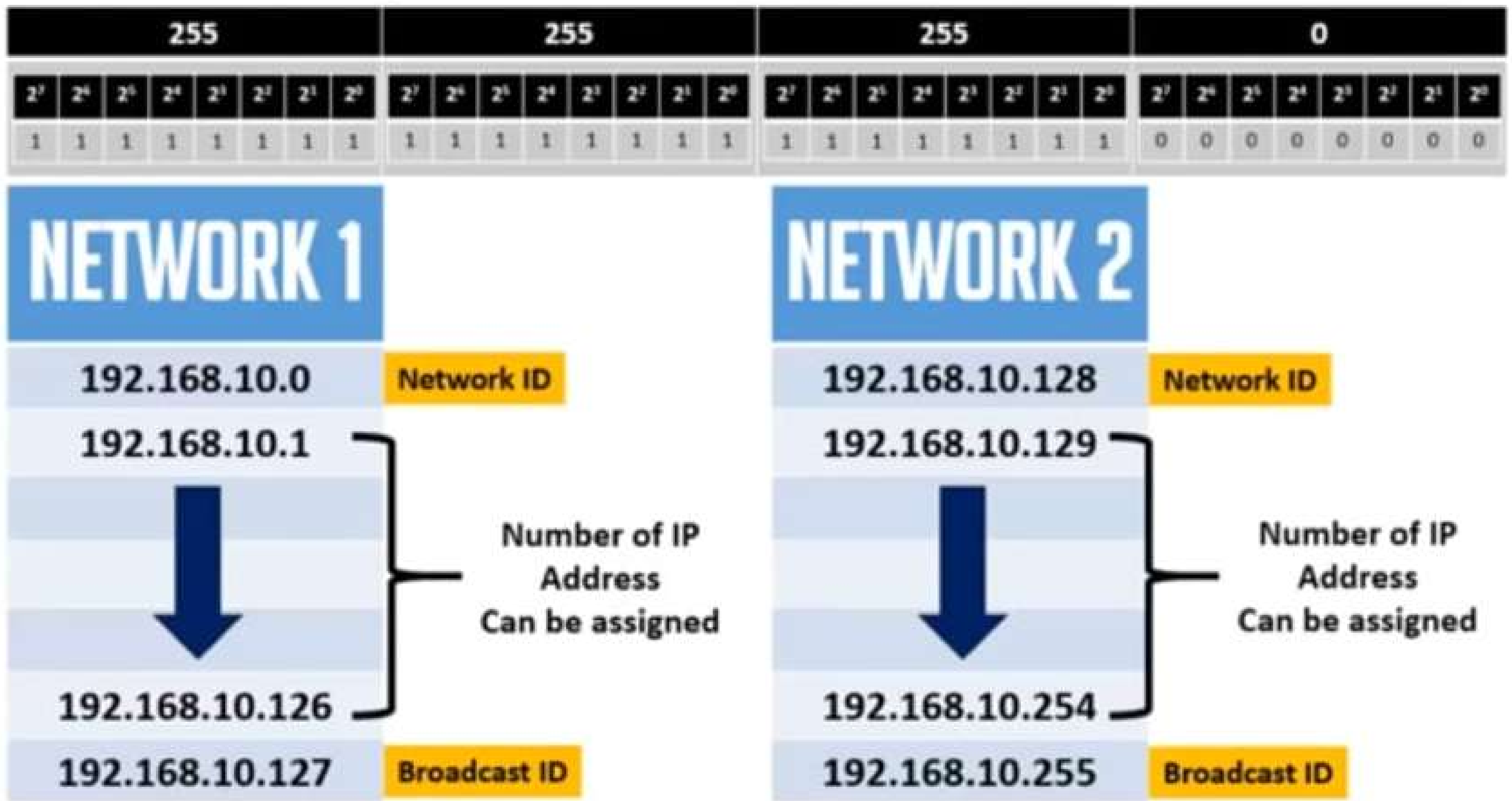
NOTE: In every network, the first ip address is reserved for the network id and the last ip address is reserved for broadcast id

192.168.10.0/25

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
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	0	0	0	0



192.168.10.0/25



192.168.10.0/26

- Find the no of possible networks
- Find the no of IP address
- Find the no of host on each network

192.168.10.0/26

255							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	1	1	1	1	1	1

255							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	1	1	1	1	1	1

255							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	1	1	1	1	1	1

0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	0	0	0	0	0	0

Total No of Networks: 4

Total No of IP Address on each network: 64

Total No of Host on each network: 64-2=62

192.168.10.0/26

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
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	0	0	0	0

Total No of Networks: 4

Total No of IP Address on each network: 64

Total No of Host on each network: 64-2=62

NETWORK NO	NETWORK ID	
1	192.168.10.0	0
2	192.168.10.64	0+64=64
3	192.168.10.128	64+64=128
4	192.168.10.192	128+64=192

192.168.10.0/26

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
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	0	0	0	0

Total No of Networks: 4

Total No of IP Address on each network: 64

Total No of Host on each network: 64-2=62

NETWORK NO	NETWORK ID	HOST ADDRESS	BROADCAST ID
1	192.168.10.0	192.168.10.1 - 192.168.10.62	192.168.10.63
2	192.168.10.64	192.168.10.65 – 192.168.10.126	192.168.10.127
3	192.168.10.128	192.168.10.129 – 192.168.10.190	192.168.10.191
4	192.168.10.192	192.168.10.193 -192.168.10.254	192.168.10.255

Fixed Length subnet masking
FLSM

What is a fixed-length subnet mask (FLSM)?

- A fixed-length subnet mask (FLSM) refers to a type of enterprise or provider networking where a block of IP addresses is divided into multiple subnets of equal length, i.e. an equal number of IP addresses. FLSM streamlines packet routing within the subnets of a proprietary network.
- A subnet can be a geographically defined local area network (LAN). Alternatively, it may define security boundaries, departmental boundaries, multicast zones or hardware security parameters. One benefit is to create locally significant subnet identification addresses
- Subnetting means dividing a network into multiple smaller subnetworks by using a subnetting mask. In FLSM, the number of IP addresses is the same in each subnet. Another way of saying this is that the same number of IP addresses are allocated to each subnet. Therefore, the subnet mask used will be the same for all the subnets.

What is a Variable-length subnet mask (VLSM)?

- Varrying based on required hosts

Problem 1:

An ISP has given a block 10.0.50.0/24 and organization have 16 departments in which they wish to connect the machines. Find the subnet mask, subnet address of all 16 networks and 1st and last IP of machines in each subnetwork.

Solution :-

Given Network Address :- 10.0.50.0/24.

Since /24 → Subnet mask = 255.255.255.0.

Total departments needed = 16 = 2^{4}

∴ For utilization $2^4 = 16$ ⇒ 4 bits in host Id has to be borrowed.

∴ 24+4 → (/28) → New CIDR value.

The first three octet remains const.

∴ 11111111 . 11111111 . 11111111 . $\boxed{0000}$0000 – 1st subnet Add

11111111 . 11111111 . 11111111 . $\boxed{1111}$0000 – 16^{th} Subnet Add

First Subnet block :-

$10 . 0 . 50 . 0/28 \Rightarrow 10.0.50.\boxed{0000}0000$

1st - Subnetwork Address ← 10 . 0 . 50 . 0

Possible Address ⇐ 10 . 0 . 50 . 1 – 10 . 0 . 50 . 14

1st - Broadcast Address ← 10 . 0 . 50 . 15

$10 . 0 . 50 . \boxed{0000}1111$

Second Subnet block :-

$\boxed{0001}0000$

10 · 0 · 50 · 16 / 28 ⇒ 10 · 0 · 50 · $\boxed{0001}0000$

2nd Subnet work Address ⇐ 10 · 0 · 50 · 16

$\boxed{\text{Possible Address to Hosts}}$ ⇐ 10 · 0 · 50 · 17 … 10 · 0 · 50 · 30

2nd Broadcast Address ← 10 · 0 · 50 · 31

10 · 0 · 50 · $\boxed{0001}1111$

0010 0000

3rd Subnet Addr = 10.0.50.32 / 28

0011

4th Subnet Address = 10.0.50.48/28

0100

5th Subnet Address = 10.0.50.64 /28

0101

6th Subnet Address = 10.0.50.80/ 28

7th Subnet Address = 10.0.50.96/ 28

8th Subnet Addr = 10.0.50.112/28

9th - Subnet Address — 10.0.50.128/28

10th - SN Add — 10.0.50.144/28

11th → SN Add — 10.0.50.160/28

12th → SN Add — 10.0.50.176/28

13th → SN Add — 10.0.50.192/28

14th → SN Add — 10.0.50.208/28

15th → SN Add — 10.0.50.224/28

16th → SN Add — 10.0.50.240/28

1111 0000

Problem 2

An organization is granted the block 16.0.0.0/8. The administrator wants to create 500 fixed length subnets:

a. Find the subnet mask

b. Find the number of addresses in each subnet

c. Find 1st and last address in 1st Subnet

d. Find 1st and last address in last subnet (500th Subnet)

Subnet Calculator

Number of bits	Number of Subnets
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
11	2048
12	4096

a. Find the subnet mask

An organization is granted the block 16.0.0.0/8

Given Mask: /8 *11111111 00000000 00000000 00000000*

The administrator wants to create 500 fixed length subnets

So, Number of bits required from host id part is : 9 (2^9 =512 ~500)

Subnet mask is: 8 + 9 =17

Hence, Subnet mask is /17 or *11111111 11111111 10000000 00000000*

b. Find the number of addresses in each subnet

Subnet mask is /17

Number of bits for Host are: 32-17=15

Hence, Number of addresses in each subnet= 2^{15} =32768

1st Subnet:

1st address: 16.0.0.0 *00010000 00000000 00000000 00000000*

Find 1st and last address in 1st Subnet

1st Subnet:

1st address: 16.0.0.0   *00010000 00000000 00000000 00000000*

last address: ***16.0.127.255***   *00010000 00000000 0**1111111** **11111111***

For 500th subnet

d. Find 1st and last address in last subnet (500th Subnet)

500th Subnet:

1	1	1	1	1	0	0	1	1
256	128	64	32	16	8.	4	2	1

Subnet mask: *11111111 11111111 10000000 00000000*

500th subnet address: *00010000 11111001 10000000 00000000*

1st address: *16.* *249.* *128.* *0*

last address: 16. 249. 255. 255

500th Subnet address: 16.249.128.0/17

1st address: 16.249.128.0

Last Host Address: 16.249.255.254

Problem 3

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

a. The first group has 64 customers; each needs 256 addresses.
b. The second group has 128 customers; each needs 128 addresses
c. The third group has 128 customers; each needs 64 addresses.

Number of Host (Theoretically), practically $2^n - 2$

Number of bits (Suffix)	Number of Host
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
11	2048
12	4096

Given block is: 190.100.0.0/16

The first group has 64 customers; each needs 256 addresses.

Number of bits (Suffix)	Number of Host
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	**256**
9	512
10	1024
11	2048
12	4096

Given block is: 190.100.0.0/16

The first group has 64 customers; each needs 256 addresses.

Mask is 32-8=24

Customer-1: 190.100.0.0/24 to 190.100.0.255/24

Customer-2: 190.100.1.0/24 to 190.100.1.255/24

Customer-64: 190.100.63.0/24 to 190.100.63.255/24

Number of bits (Suffix)	Number of Host
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	**256**
9	512
10	1024
11	2048
12	4096

Given block is: 190.100.0.0/16

The second group has 128 customers; each needs 128 addresses.

Mask is 32-7=25

Available after Group-1 last customer:

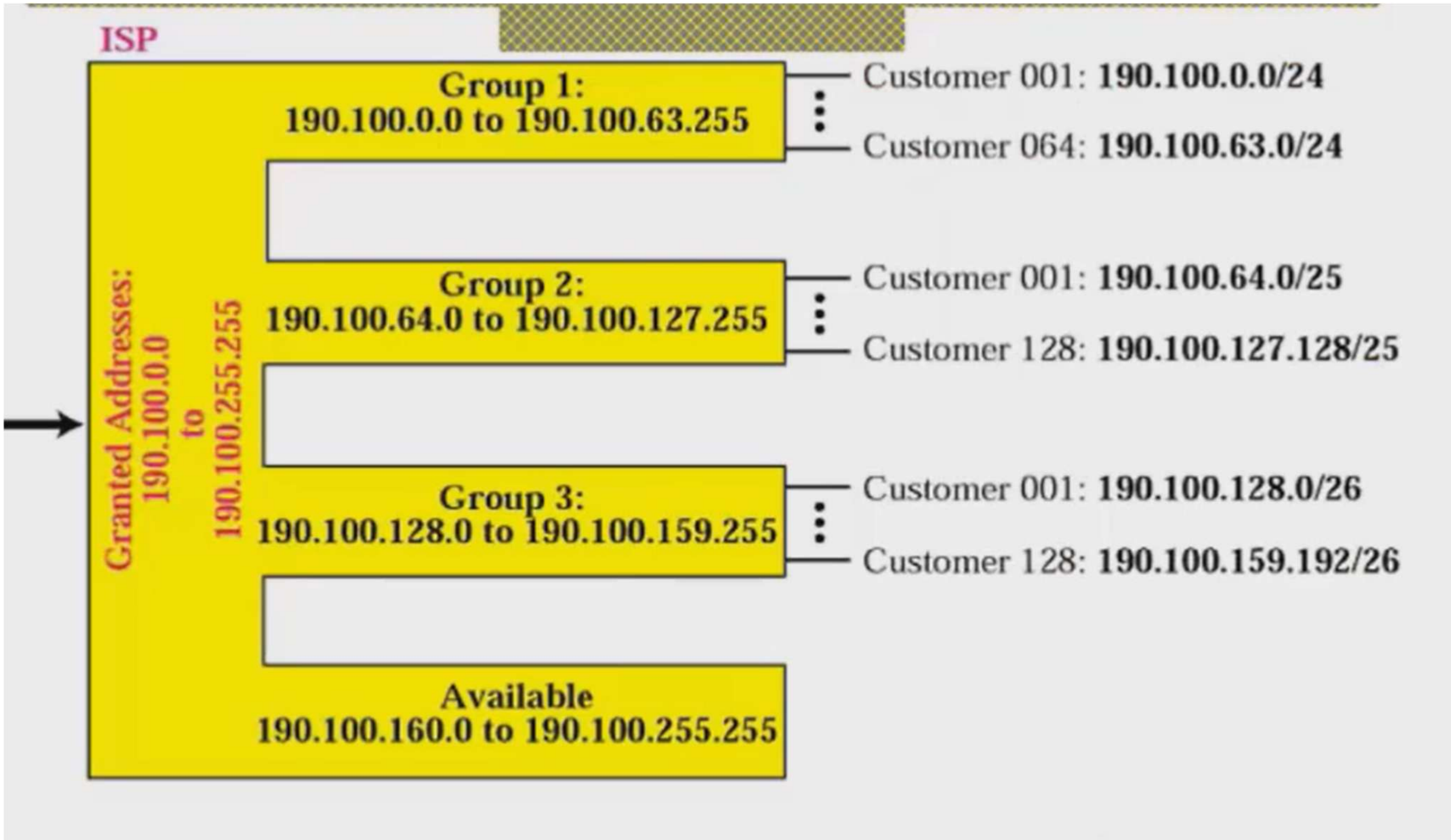
Customer-1: 190.100.64.0/25 to 190.100.64.127/25

Customer-2: 190.100.64.128/25 to 190.100.64.255/25

Customer-3: 190.100.65.0/25 to 190.100.65.127/25

Customer-128: 190.100.127.128/25 to 190.100.127.255/25

Number of bits (Suffix)	Number of Host
1	2
2	4
3	8
4	16
5	32
6	64
7	**128**
8	256
9	512
10	1024
11	2048
12	4096


ISP
Granted Addresses:
190.100.0.0
to
190.100.255.255
Group 1:
190.100.0.0 to 190.100.63.255
Customer 001: 190.100.0.0/24
Customer 064: 190.100.63.0/24
Group 2:
190.100.64.0 to 190.100.127.255
Customer 001: 190.100.64.0/25
Customer 128: 190.100.127.128/25
Group 3:
190.100.128.0 to 190.100.159.255
Customer 001: 190.100.128.0/26
Customer 128: 190.100.159.192/26
Available
190.100.160.0 to 190.100.255.255

- Consider a block of address with starting address 112.78.0.0/16. Find the first and last addresses for each group, if the number and size of the subnets required by them are as given in the input.

- Group-1:
- 256 subnets, each needs 128 addresses

- Group-2:
- 1024 subnets, each requiring 4 addresses

- Group-3:
- 128 subnets, each consisting of 16 addresses

```
Group-1:
First Address : 112.78.0.0 /17
Last Address : 112.78.127.255 /17

Group-2:
First Address : 112.78.128.0 /20
Last Address : 112.78.143.255 / 20

Group-3:
First Address : 112.78.144.0 /21
Last Address : 112.78.151.255 /21
```