AWS NOTES

VPC-Virtual Private Cloud

TOPIC

SUBNETTING

IP Addresses & Its Classes, Network Classes,
Subnet Mask, Hosts & Network Addresses,
VPC CIDR & Subnetting

Every computer or device on the internet/network has two types of addresses associated with it:
Physical IP- MAC Address- Physical address (can't be changed)
Logical IP- Internet Address (IP- Internet Protocol) (Two versions: IPv4 & IPv6)

Before we start out journey with IPv4 and subnetting let's make sure that you understand binary numbers and how to covert decimal numbers into binary, cause computers only understand binaries.
Like all the decimal numbers are represented using 10 digits ranging from 0 – 9, binary numbers are represented using two digits only- 0 & 1. We can represent any number in binary with the combination of 0's and 1's.

Decimal to Binary Conversion Trick

Decimal Number	128	64	32	16	8	4	2	1	Binary number
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	10
10	0	0	0	0	1	0	1	0	1010
23	0	0	0	1	0	1	1	1	10111
47	0	0	1	0	1	1	1	1	101111
128	1	0	0	0	0	0	0	0	10000000
255	1	1	1	1	1	1	1	1	11111111

What is IP address & its versions?

IP address or Internet Protocol address is a **unique address (numerical label) assigned to each device** connected over a local network or over an internet.
Two main functions: Identifying the host **or network interface** and providing the **location addressing** for routing packets across the network.

Concept:
Host- *In a network, a host simply refers to any device that is connected to a computer network and has its own unique IP address.*
Eg*: computer, laptop, server, printer, or any other device capable of connecting and communicating within a network.*

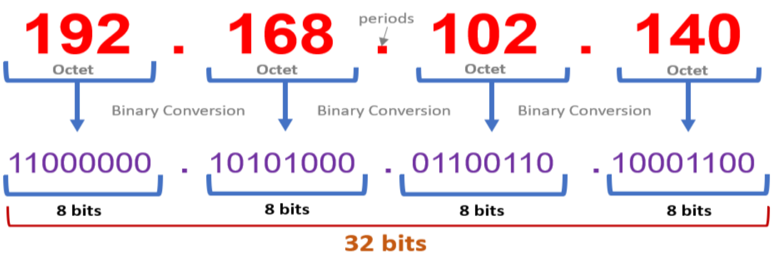
There are two versions of IP addresses in use today:

IPv4 (Internet Protocol version 4) – 32 bits
IPv6 (Internet Protocol version 6) – 128 bits

Let's understand the IPv4 format:

IPv4 addresses are 32-bit numbers expressed in a format like "**192.168.0.1**", consisting of **four sets of numbers** separated by periods (**·**) each octet ranging from **0 to 255**.

IPv4 Address Format



CLASSES OF IP

Now that you have understood what an IPv4 address is, and how IPv4 is formed. The next important thing to understand is IP classes. IPv4 addresses are divided into 5 classes to provide a structure for addressing and network allocation. Each address class defines a different number of bits for its **network address** and **host address**.

There are five classes: **A, B, C, D and E**. Each class has a specific range of IP. Primarily, class A, B, and C important & are used by most devices on the Internet. **Class D and class E are for special uses,** we are going to deal with **Class A, Class B & Class C**.

Class A:	1 – 126	**Range:** 1.0.0.0 and 126.255.255.255	127.0.0.0- Loopback IP
Class B:	128 – 191	**Range:** 128.0.0.0 and 191.255.255.255	
Class C:	192 – 223	**Range:** 192.0.0.0 and 223.255.255.255	
Class D:	224 – 239	**Range:** 224.0.0.0 to 239.255.255.255	
Class E:	240 – 255	**Range:** 240.0.0.0 to 255.255.255.255.	

127.0.0.0 – 127.255.255.255 Loop-back address *(For network testing and diagnosis purpose)* Also falls under class A

You can identify the class of IP by its first octet.
For example: **192**.168.0.0 is of class C IP

Why are this Classes so important ?

Well, depending on the IP address class, IP address is divided into two part the first part/portion of the IP is always designated to the network called as **network address** and the second part/portion of the IP is always designated to the hosts called as **host address.**

The host address, on the other hand, represents the unique device within a network. The host address is determined by the remaining bits after the network portion.

The network address is the part of an IP address that identifies the network to which a device belongs. It identifies the network and helps in routing packets across the internet. We can determine the network address by its class.

Let's understand this by an example,

Example 1) For a **class A** IP it only uses **8 bits of the IP address for the network**, leaving **24 for the host**. So, using the example 126.27.61.137, the network IP address would be 126.0.0.0 and the host address would be 0.27.61.137.

$$\underbrace{126}_{\text{Represents Network Address /Network prefix}}.\underbrace{27.61.137}_{\text{Represents Host Address}}$$

Example 2) For a **class B** IP address, **16 bits are used for the network** and **16 bits for the host**. Using 129.167.232.190 as an example, we can say that 129.167.0.0 for the network addressing and 0.0.232.190 for the host

Example 3) For a **class C** address, 24 bits are used for the network, and eight remain for the host. Using 200.23.65.10 as an example, this would result in 200.23.65.0 for the network and 0.0.0.10 for the host.

For networks such as a TCP/IP network to function, the routers passing information throughout the network don't need to know the exact host address. They only need to know the network part of the IP address; then, once the packet is delivered to the network, it can get to the right host.

Questions: Find out the class of the following IP, also find out network address and host address of the following IP address.

1) 126.27.61.137
2) 200.23.65.0
3) 150.43.52.20
4) 5.230.200.1
5) 255.257.10.0
6) 192.168.2.259

Answers:

1) The given IP address is 126.27.61.137

To determine the class, we can examine the value of the first octet:

Class A addresses have a range of 1.0.0.0 to 127.255.255.255.

As the first octet of 126.27.61.137 falls within this range, **it belongs to Class A.**

Next,

let's identify the network address and host address:

As we seen before, In Class A addresses, the first octet represents the network portion, while the remaining three octets represent the host portion.

Therefore, the network address for 126.27.61.137 is 126.0.0.0, and the host address is 0.27.61.137.

To summarize:

Class: Class A

Network address: 126.0.0.0

Host address: 0.27.61.137

2) The given IP address is 200.23.65.50

To determine the class, we can look at the value of the first octet:

Class C addresses have a range of 192.0.0.0 to 223.255.255.255

Since the first octet of 200.23.65.50 falls within this range, it belongs to Class C.

Now let's determine the network address and host address:

In Class C addresses, the first three octets represent the network portion, while the last octet represents the host portion.

Therefore, the network address for 200.23.65.50 is 200.23.65.0, and the host address is 0.0.0.50

To summarize:

Class: Class C

Network address: 200.23.65.0

Host address: 0.0.0.50

3) The given IP address is 150.43.52.20

To determine the class, we can examine the value of the first octet:

- Class B addresses have a range of 128.0.0.0 to 191.255.255.255
- Since the first octet of 150.43.52.20 falls within this range, it belongs to Class B.

Now let's determine the network address and host address:

- In Class B addresses, the first two octets represent the network portion, while the remaining two octets represent the host portion.
- Therefore, the network address for 150.43.52.20 is 150.43.0.0, and the host address is 0.0.52.20.

To summarize:

Class: Class B

Network address: 150.43.0.0

Host address: 0.0.52.20

4) Given IP is 5.230.200.1

Class: Class A

Network address: 5.0.0.0

Host address: 0.230.200.1

5) The given IP is 255.257.10.0

If you see the IP address given is not valid IPv4 address the 2nd octet is invalid as it is not within the range of 0 -255. Till now we have seen that in a valid IPv4 address, each octet should range between 0 to 255. However, "257" is greater than the maximum value of 255. Hence, IP 255.257.10.0 is **not a valid IP** address.

6) The given IP is 192.168.2.259

Invalid IP- 192.168.2.259 as last octet is not within the range of 0 – 255

Subnet Mask:

Now that you have understand the IPv4 & its classes, Network address and host address. let's see the next important term Subnet Maks.

Each IP address has a matching Subnet Mask which is easiest way to identify which part of IP address associated to network and which part is associated to host.
We can also say that a subnet mask is used to divide an IP address into two parts, the network part, and the host part. Helps us to identify which part of an IP address is the network address and which part of the IP is for Host address. Subnet mask is of 32 bits, in binary subnet mask is all about setting network bits to '1' and host bits to '0'.

For example, Subnet mask for IP: 192.168.1.100 (Class C)

Let's convert it into binary first:

IP address:	11000000.10101000.00000001.01100100	(192.168.1.100)
Network address:	11000000.10101000.00000001.00000000	(192.168.1.0)

Now we know that subnet mask is **setting network bits to '1' & host bits to '0'**

Subnet mask: 11111111.11111111.11111111.00000000 **(255.255.255.0)**

The default subnet masks for each class are as follows:

- **Class A: 255.0.0.0**
- **Class B: 255.255.0.0**
- **Class C: 255.255.255.0**

Subnetting:

Subnetting is the process of dividing a single network into multiple smaller sub-networks (subnets). It involves partitioning an IP network into multiple subnetworks by borrowing bits from the host portion of the IP address.

The primary purpose of subnetting is to improve network efficiency, security, and scalability. By creating smaller subnets, it allows for better utilization of IP addresses, reduces network congestion, and improves overall performance.

So basically, subnet mask can tell us a lot about a network, such as the following:

- The network and host portion of an IP address
- The number of hosts within a network

Benefits of the Subnetting:

- **Efficient IP address utilization:** By dividing a network into subnets, it helps to reduce IP address wastage and allows for efficient allocation of addresses.
- **Enhanced network security:** Subnets can be used to isolate different segments of a network, improving security by controlling access and limiting the impact of network breaches.
- **Improved network performance:** Subnets can help in reducing network congestion by limiting the scope of broadcast traffic and providing better control over network traffic flow.
- **Scalability:** Subnetting enables easier network expansion and the ability to accommodate a growing number of devices.

Subnetting requires careful planning and consideration of network requirements, such as the **number of subnets needed, the number of hosts per subnet,** and **the network topology**. Properly designed subnetting can greatly enhance the functionality and efficiency of an IP network.

CIDR- Classless or Classless Inter-Domain Routing:

The class A, Class B and Class C are the classful whereas **Classless or Classless Inter-Domain Routing (CIDR)** addresses use variable length subnet masking (VLSM) to alter the ratio between the network and host address bits in an IP address. A subnet mask is a set of identifiers that returns the network address's value from the IP address by turning the host address into "0".

A VLSM sequence allows network administrators to break down an IP address space into subnets of various sizes. Each subnet can have a flexible host count and a limited number of IP addresses. A CIDR IP address appends a suffix value stating the number of network address prefix bits to a normal IP address.

For example, 192.0.2.0/24 is an IPv4 CIDR address where the first 24 bits, or 192.0.2 is the network address.

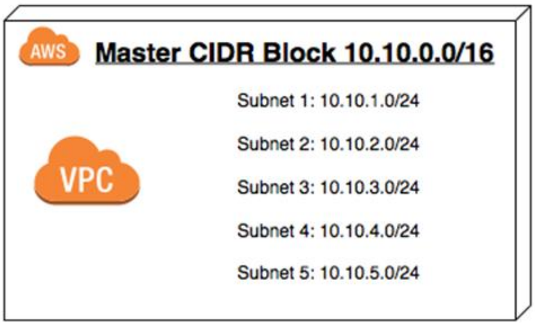
How does CIDR work?

Classless Inter-Domain Routing (CIDR) allows network routers to route data packets to the respective device based on the indicated subnet. Instead of classifying the IP address based on classes, routers retrieve the network and host address as specified by the CIDR suffix.

CIDR blocks

A CIDR block is a collection of IP addresses that share the same network prefix and number of bits. A large block consists of more IP addresses and a small suffix.

The Internet Assigned Numbers Authority (IANA) assigns large CIDR blocks to regional internet registries (RIR). Then, the RIR assigns smaller blocks to local internet registries (LIR), which then assign them to organizations.



CIDR notation

CIDR notation represents an IP address and a **suffix that indicates network identifier bits** in a specified format. For example, we can express 192.168.1.0 with a 22-bit network identifier as-

192.168.1.0/22

CIDR Notation *(Network identifier /netmask)*

We can get host bits by subtracting from 32 bits.

32 – 22 = 10 host bits

We know that subnet mask is setting host bits to '0' so let's set last 10 bits to '0' and rest to '1'

11111111.11111111.11111000.00000000

↓

Subnet Maks for given CIDR:

255.255.248.0 (Class B- 255.255.0.0 subnet mask)

AWS:

When we create a VPC, we must specify an IPv4 CIDR block for the VPC. The allowed block size is between a /16 netmask (65,536 IP addresses) and /28 netmask (16 IP addresses). After we've created your VPC, we can associate additional IPv4 CIDR blocks with the VPC.

Default CIDRs

Class A-	Network Prefix- /8	Host Identifier- 32-8 = 24	**Eg:** 44.0.0.1
Class B-	Network Prefix- /16	Host Identifier- 32-16 = 16	**Eg:** 128.32.0.1
Class C-	Network Prefix- /24	Host Identifier- 32-24 = 8	**Eg:** 192.12.33.3

Concept: Usable IP and Resaved IP

For any network or subnetwork, there are few IP's which are reserved for networks use and which we cannot use for the hosts assignments.

Generally, in normal network,

1st IP - Network IP

Last IP - Broadcast IP

(So we can get usable IPs by subtracting 2 from the total available IPs in network/subnets)

Eg: Suppose if you have IP 10.0.0.0/24

10.0.0.0/24 - Network IP

:

:

:

10.0.0.255/24 - Broadcast IP

Usable IP's: Total IP - 2 = 256 – 2 = 254

For AWS network,

The first four IP addresses and the last IP address in each subnet CIDR block are not available for your use, and they cannot be assigned to a resource, such as an EC2 instance. For example, in a subnet with CIDR block 10.0.0.0/24, the following five IP addresses are reserved:

10.0.0.0: Network IP

10.0.0.1: Router IP reserved in VPC

10.0.0.2: DNS IP

10.0.0.3: Future Use

:

:

10.0.0.255: Broadcast IP

Hence for AWS we can get total number of usable IPs by subtracting 5 from the total host.

Examples

Que 1: 192.168.10.0/28 For the given CIDR find,

- **Subnet mask**
- **Total number of subnets**
- **Total number of hosts per subnet**
- **Usable IP's**

Solution:

Given IP is- 192.168.10.0/28

Step 1: Lets calculate first thing first, that is subnet mask.

We know that subnet mask is dividing IP into network address and host address can be done by setting network bits to 1 and host bits to 0, we have network bit given as -28 lets find out host bits

total bits – network bits = 32 – 28 = 4 host bits

so let's set last 4 bits to 0 and rest all to 1

$$\underbrace{\underbrace{11111111.11111111.11111111.1111}_{\text{network portion}}\underbrace{0000}_{\text{host portion}}}_{\text{255.255.255.240/28}}$$

(class C- **/24** subnet mask- 255.255.255.0)

Decimal conversion

255.255.255.240/28 ---- Subnet Maks

Step 2: Let's find out total number of possible subnets now,

Short Trick: $2^{(\text{given bits – default subnet mask's class})} = 2^{(28-24)} = 2^4 = 16$ subnets

Step 3: Let's find out number of hosts per subnet:

Trick: $2^{(\text{host bits})} = 2^4 = 16$ hosts/subnet

We have 16 subnets and 16 host per subnet so let's write it down:

Subnet	**IP Range**			**Total Host**	**Usable IP** (Total IP -2)	**Usable IP (AWS)** (Total IP -5)
1	192.168.10.0/28	-	192.168.10.15/28	16	14	11
2	192.168.10.16/28	-	192.168.10.31/28	16	14	11
3	192.168.10.32/28	-	192.168.10.47/28	16	14	11
4	192.168.10.48/28	-	192.168.10.63/28	16	14	11
5	192.168.10.64/28	-	192.168.10.79/28	16	14	11
6	192.168.10.80/28	-	192.168.10.95/28	16	14	11
7	192.168.10.96/28	-	192.168.10.111/28	16	14	11
8	192.168.10.112/28	-	192.168.10.127/28	16	14	11
9	192.168.10.128/28	-	192.168.10.143/28	16	14	11
10	192.168.10.144/28	-	192.168.10.159/28	16	14	11
11	192.168.10.160/28	-	192.168.10.175/28	16	14	11
12	192.168.10.176/28	-	192.168.10.191/28	16	14	11
13	192.168.10.192/28	-	192.168.10.207/28	16	14	11
14	192.168.10.208/28	-	192.168.10.223/28	16	14	11
15	192.168.10.224/28	-	192.168.10.239/28	16	14	11
16	192.168.10.240/28	-	192.168.10.255/28	16	14	11
			Total	**256**	**224**	**176**

AWS Usable IP: For every subnet first four & last IP are reserved so we cannot use it,
For example, for subnet 1,

1st IP-	192.168.10.0/28	-	Network IP
2nd IP-	192.168.10.1/28	-	VPC Router IP
3rd IP-	192.168.10.2/28	-	DNS IP
4th IP-	192.168.10.3/28	-	Future use
	:		
	:		
	:		
	:		
Last IP-	192.168.10.15/15	-	Broadcast IP

For normal network only 1st and last IP is reserved.

Hence for each subnet we get only 11 IPs for use.

Que 2: 172.16.0.0/23 For the given CIDR find,
- **Subnet mask**
- **Total number of subnets**
- **Total number of hosts per subnet**
- **Usable IP's**

Solution:

Given CIDR is- 172.16.0.0/23

Step 1: Lets first find out subnet mask,

32 - 23 = 9 Host bits

Let's set last 9 bits to 0 and rest all to 1

11111111.111111111.11111110.00000000 (class B- **/16** subnet mask- 255.255.0.0)

255.255.254.0 **------- Subnet Mask**

Step 2: Total number of possible subnets,

Short Trick: $2^{(\text{given bits – default subnet mask's class})} = 2^{(23-16)} = 2^{7} = 128$ Subnets

Step 3: lets calculate number of hosts per subnet,

Trick: $2^{(\text{host bits})} = 2^{9} = 512$ Hosts per subnet

So we have 128 subnets with 512 hosts per subnet:

Subnet 1:	172.16.0.0/23 - 172.16.1.255/23	→	512 Hosts (Usable- 510, AWS- 507)
Subnet 2:	172.16.2.0/23 - 172.16.3.255/23	→	512 Hosts (Usable- 510, AWS- 507)
Subnet 3:	172.16.4.0/23 - 172.16.5.155/23	→	512 Hosts (Usable- 510, AWS- 507)
	:		
	:		
	:		
	:		
Subnet 128:	172.16.254.0/23 - 172.16.255.255/23	→	512 Hosts (Usable- 510, AWS- 507)

Total IP: **65,536**

Usable IP (normal network): **65,280**

Usable IP (AWS network): **64,768**

Que 3: 154.168.10.0/28 For the given CIDR find,

- **Subnet mask**
- **Total number of subnets**
- **Total number of hosts per subnet**
- **Usable IP's**

Solution:

Step 1: Subnet Mask,

32 – 28 = 4 Host bits

11111111.11111111.11111111.11110000 (class C subnet mask= 255.255.255.0)

255.255.255.240 --- Subnet Mask

Step 2: Number of possible subnets for given CIDR:

Short Trick: $2^{(\text{given bits – default subnet mask's class})} = 2^{(28 - 24)} = 2^4 = 16$ Subnets

Step 3: Number of hosts per subnet:

Trick: $2^{(\text{host bits})} = 2^4 = 16$ host/subnet

Subnet	**IP Range**			**Total Host**	**Usable IP** (Total IP -2)	**Usable IP (AWS)** (Total IP -5)
1	154.168.10.0/28	-	154.168.10.15/28	16	14	11
2	154.168.10.16/28	-	154.168.10.31/28	16	14	11
3	154.168.10.32/28	-	154.168.10.47/28	16	14	11
4	154.168.10.48/28	-	154.168.10.63/28	16	14	11
5	154.168.10.64/28	-	154.168.10.79/28	16	14	11
6	154.168.10.80/28	-	154.168.10.95/28	16	14	11
7	154.168.10.96/28	-	154.168.10.111/28	16	14	11
8	154.168.10.112/28	-	154168.10.127/28	16	14	11
9	154.168.10.128/28	-	154.168.10.143/28	16	14	11
10	154.168.10.144/28	-	154.168.10.159/28	16	14	11
11	154.168.10.160/28	-	154.168.10.175/28	16	14	11
12	154.168.10.176/28	-	154.168.10.191/28	16	14	11
13	154.168.10.192/28	-	154.168.10.207/28	16	14	11
14	154.168.10.208/28	-	154.168.10.223/28	16	14	11
15	154.168.10.224/28	-	154.168.10.239/28	16	14	11
16	154.168.10.240/28	-	154.168.10.255/28	16	14	11
			Total	**256**	**224**	**176**