**VPC: Virtual Private Cloud**

**Public cloud – Isolation**

**IP Addresses**

In order to send and direct data across a network, computers need to be able to identify destinations and origins. This identification is an IP—Internet Protocol—address. An IP address is just a set of four numbers between 1 and 254, separated by dots. An example of an IP address is 173.194.43.7.

**DHCP—Dynamic Host Configuration Protocol**

It assigns IP addresses to client devices, such as desktop computers, laptops, and phones, when they are plugged into Ethernet or connect to Wireless networks.

**Ethernet**

A type of networking protocol—it defines the types of cables and connections that are used to wire computers, switches, and routers together. Most often Ethernet cabling is Category 5 or 6, made up of twisted pair wiring similar to phone cables.

**Hub**

A network device that repeats the traffic it receives to all connected devices.

**Switch**

A network device that sends traffic it receives to a specific connected device, such as a single desktop computer or laptop.

**Router**

A network device that can bridge between different networks, determine what traffic can pass between them, and perform other functions on a network, such as assigning IP addresses.

**Firewall**

A function typically performed by routers, this filters traffic between networks and can protect them from interference or attacks.

IPV4:

The maximum value of a 32-bit number is 232, or 4,294,967,296. So the maximum number of IPv4 addresses, which is called its [address space](https://www.computerhope.com/jargon/a/address-space.htm), is about **4.3 billion**. In the 1980s, this was sufficient to address every networked device, but scientists knew that this space would quickly become exhausted. Technologies such as [NAT](https://www.computerhope.com/jargon/n/nat.htm) have delayed the problem by allowing many devices to use a single IP address, but a larger address space is needed to serve the modern Internet.

A major advantage of IPv6 is that it uses 128 bits of data to store an address, permitting 2128 unique addresses, or 340,282,366,920,938,463,463,374,607,431,768,211,456. The size of IPv6's address space — 340 duodecillion — is much, much larger than IPv4.

**IP address classes**

With an IPv4 IP address, there are five classes of available IP ranges: Class A, Class B, Class C, Class D and Class E, while only A, B, and C are commonly used. Each class allows for a range of valid IP addresses, shown in the following table.

|  |  |  |
| --- | --- | --- |
| **Class** | **Address range** | **Supports** |
| **Class A** | 1.0.0.1 to 126.255.255.254 | Supports 16 million hosts on each of 127 networks. |
| **Class B** | 128.1.0.1 to 191.255.255.254 | Supports 65,000 hosts on each of 16,000 networks. |
| **Class C** | 192.0.1.1 to 223.255.254.254 | Supports 254 hosts on each of 2 million networks. |
| **Class D** | 224.0.0.0 to 239.255.255.255 | Reserved for [multicast](https://www.computerhope.com/jargon/m/multicast.htm) groups. |
| **Class E** | 240.0.0.0 to 254.255.255.254 | Reserved for future use, or research and development purposes. |

Ranges 127.x.x.x are reserved for the [loopback or localhost](https://www.computerhope.com/jargon/l/locahost.htm), for example, **127.0.0.1** is the loopback address. Range **255.255.255.255** [broadcasts](https://www.computerhope.com/jargon/b/broadcas.htm) to all hosts on the local network.

### Reserved Private Ranges

There are also some portions of the IPv4 space that are reserved for specific uses.

One of the most useful reserved ranges is the loopback range specified by addresses from 127.0.0.0 to 127.255.255.255. This range is used by each host to test networking to itself. Typically, this is expressed by the first address in this range: 127.0.0.1.

Each of the normal classes also have a range within them that is used to designate private network addresses. For instance, for class A addresses, the addresses from 10.0.0.0 to 10.255.255.255 are reserved for private network assignment. For class B, this range is 172.16.0.0 to 172.31.255.255. For class C, the range of 192.168.0.0 to 192.168.255.255 is reserved for private usage.

Any computer that is not hooked up to the internet directly (any computer that goes through a router or other NAT system) can use these addresses at will.

CIDR Notation

A system called **Classless Inter-Domain Routing**, or CIDR, was developed as an alternative to traditional subnetting. The idea is that you can add a specification in the IP address itself as to the number of significant bits that make up the routing or networking portion.

For example, we could express the idea that the IP address 192.168.0.15 is associated with the netmask 255.255.255.0 by using the CIDR notation of 192.168.0.15/24. This means that the first 24 bits of the IP address given are considered significant for the network routing.

This allows us some interesting possibilities. We can use these to reference "supernets". In this case, we mean a more inclusive address range that is not possible with a traditional subnet mask. For instance, in a class C network, like above, we could not combine the addresses from the networks 192.168.0.0 and 192.168.1.0 because the netmask for class C addresses is 255.255.255.0.

However, using CIDR notation, we can combine these blocks by referencing this chunk as 192.168.0.0/23. This specifies that there are 23 bits used for the network portion that we are referring to.

So the first network (192.168.0.0) could be represented like this in binary:

1100 0000 - 1010 1000 - 0000 0000 - 0000 0000

While the second network (192.168.1.0) would be like this:

1100 0000 - 1010 1000 - 0000 0001 - 0000 0000

The CIDR address we specified indicates that the first 23 bits are used for the network block we are referencing. This is equivalent to a netmask of 255.255.254.0, or:

1111 1111 - 1111 1111 - 1111 1110 - 0000 0000

As you can see, with this block the 24th bit can be either 0 or 1 and it will still match, because the network block only cares about the first 23 digits.

Basically, CIDR allows us more control over addressing continuous blocks of IP addresses. This is much more useful than the subnetting we talked about originally.

**MAC Address (Media Access Control address):**  
Also known as physical address, is the unique identifier of each host and is associated with the NIC (Network Interface Card).

**DNS Server:**  
DNS stands for **Domain Name system**.  
DNS is basically a server which translate web addresses or URL (ex: www.google.com) into their corresponding IP addresses. We don’t have to remember all the IP addresses of each and every website.  
The command ‘**nslookup**’ gives you the IP address of the domain you are looking for. This also provides the information of our DNS Server.

**ARP:**  
ARP stands for **Address Resolution Protocol**.  
It is used to convert the IP address to its corresponding Physical Address(i.e.MAC Address).  
ARP is used by the Data Link Layer to identify the MAC address of the Receiver’s machine.  
**RARP:**  
RARP stands for **Reverse Address Resolution Protocol**.  
As the name suggest, it provides the IP address of the a device given physical address as input. But RARP has become obsolete since the time DHCP has come into picture.

**Understanding CIDR for AWS VPC**

**CIDR (Classless Inter-Domain Routing)** is a method used to allocate IP addresses and routing in networks. When you create a **VPC (Virtual Private Cloud)** in AWS, you need to specify an IP address range using CIDR notation. CIDR is essential for determining how many IP addresses are available for your network and its subdivisions (subnets).

**Key Concepts**

* **CIDR Notation**: It specifies an IP range. For example, 10.0.0.0/16 defines an IP block.
  + 10.0.0.0 is the **network address**.
  + /16 indicates the **subnet mask**, i.e., the number of bits that are fixed for the network. The remaining bits are available for host addresses.

**CIDR Example: 10.0.0.0/16**

* **/16** means 16 bits are reserved for the network portion of the address, leaving 16 bits for hosts (because IPv4 addresses are 32 bits long).
* A /16 CIDR block allows for **65536 IP addresses** (2^16 = 65536).

**Step-by-Step Guide to Calculating CIDR for AWS VPC**

**1. Decide the IP Range for Your VPC**

AWS allows you to create VPCs with the following private IP ranges, defined by **RFC 1918**:

* 10.0.0.0/8
* 172.16.0.0/12
* 192.168.0.0/16

You typically choose one of these ranges, depending on your need.

**Example**: Let’s choose 10.0.0.0/16 for the VPC.

**2. Calculate the Number of IPs You Need**

* **/8** → 16,777,216 IP addresses.
* **/16** → 65,536 IP addresses.
* **/24** → 256 IP addresses.

For a small VPC, /16 (65,536 IP addresses) is often a good starting point, giving flexibility for creating subnets.

**3. Calculate Subnets from VPC CIDR**

Subnets divide your VPC into smaller networks. Subnet CIDR notation should be smaller (i.e., /24, /20, /28, etc.) than the VPC CIDR.

* **Example**: Suppose you want each subnet to have 256 IP addresses.
  + /24 allows **256 IP addresses** (2^(32-24) = 2^8 = 256).
  + To create subnets from a /16 VPC, you can split it into smaller ranges like 10.0.1.0/24, 10.0.2.0/24, etc.

**4. AWS Reserved IPs**

AWS reserves **5 IP addresses** in each subnet:

* **Network Address** (first IP, e.g., 10.0.1.0).
* **Broadcast Address** (last IP, e.g., 10.0.1.255).
* Three additional IPs for AWS networking purposes.

So, for a /24 subnet, you have **251 usable IP addresses**.

**5. Choosing CIDR Block for the VPC**

To calculate the right size for your VPC and subnets:

* Determine how many subnets you need.
* Ensure that the CIDR block for the VPC can comfortably accommodate those subnets.

**Example Calculation:**

Let’s say you want:

* A VPC with IP range: 10.0.0.0/16.
* Two subnets, each with /24 CIDR, so 256 IP addresses per subnet.

**Step-by-Step for VPC and Subnet Calculation:**

1. **VPC CIDR**: 10.0.0.0/16
   * Total IPs: 2^(32 - 16) = 65,536 IPs.
2. **Subnet 1 CIDR**: 10.0.1.0/24
   * Total IPs: 2^(32 - 24) = 256 IPs, 251 usable IPs (5 reserved by AWS).
3. **Subnet 2 CIDR**: 10.0.2.0/24
   * Same calculation, 256 IPs, 251 usable.
4. You can create more subnets within this VPC by assigning new /24 blocks, like 10.0.3.0/24, 10.0.4.0/24, etc.

**CIDR Size Reference Chart:**

| **CIDR Range** | **Number of IPs** | **Usable IPs in AWS (after reserving 5)** |
| --- | --- | --- |
| /32 | 1 | 0 (single IP, not usable for subnets) |
| /30 | 4 | 1 |
| /29 | 8 | 3 |
| /28 | 16 | 11 |
| /27 | 32 | 27 |
| /26 | 64 | 59 |
| /25 | 128 | 123 |
| /24 | 256 | 251 |
| /23 | 512 | 507 |
| /22 | 1024 | 1019 |
| /16 | 65536 | 65531 |

**CIDR to IPs Formula:**

For any CIDR notation /N:

* Total IPs = 2^(32 - N)
* Usable IPs in AWS = Total IPs - 5

**Summary:**

1. **Choose a private IP range** (e.g., 10.0.0.0/16).
2. **Divide the VPC into subnets** with smaller CIDR blocks.
3. **Consider AWS reserved IPs** when calculating usable IP addresses in each subnet.
4. **Adjust subnet sizes based on requirements** (number of hosts per subnet, availability zones, etc.).

By understanding CIDR and how it is applied in AWS VPCs, you can design a scalable and efficient network architecture for your cloud infrastructure.

AWS VPC:

### VPCs and Subnets

A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, into your VPC. You can specify an IP address range for the VPC, add subnets, associate security groups, and configure route tables.

A subnet is a range of IP addresses in your VPC. You can launch AWS resources into a specified subnet. Use a public subnet for resources that must be connected to the internet, and a private subnet for resources that won't be connected to the internet.

To protect the AWS resources in each subnet, you can use multiple layers of security, including security groups and network access control lists (

NACL)

When you create a VPC, you must specify a range of IPv4 addresses for the VPC in the form of a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. This is the primary CIDR block for your VPC.

The following diagram shows a new VPC with an IPv4 CIDR block, and the main route table.


          VPC with the main route table
        

A VPC spans all the Availability Zones in the region. After creating a VPC, you can add one or more subnets in each Availability Zone. When you create a subnet, you specify the CIDR block for the subnet, which is a subset of the VPC CIDR block. Each subnet must reside entirely within one Availability Zone and cannot span zones. Availability Zones are distinct locations that are engineered to be isolated from failures in other Availability Zones. By launching instances in separate Availability Zones, you can protect your applications from the failure of a single location. We assign a unique ID to each subnet.

You can also optionally assign an IPv6 CIDR block to your VPC, and assign IPv6 CIDR blocks to your subnets.

# Internet Gateways

An internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet. It therefore imposes no availability risks or bandwidth constraints on your network traffic.

An internet gateway serves two purposes: to provide a target in your VPC route tables for internet-routable traffic, and to perform network address translation (NAT) for instances that have been assigned public IPv4 addresses.

An internet gateway supports IPv4 and IPv6 traffic.

## Enabling Internet Access

To enable access to or from the internet for instances in a VPC subnet, you must do the following:

* Attach an internet gateway to your VPC.
* Ensure that your subnet's route table points to the internet gateway.
* Ensure that instances in your subnet have a globally unique IP address (public IPv4 address, Elastic IP address, or IPv6 address).
* Ensure that your network access control and security group rules allow the relevant traffic to flow to and from your instance.

To use an internet gateway, your subnet's route table must contain a route that directs internet-bound traffic to the internet gateway. You can scope the route to all destinations not explicitly known to the route table (0.0.0.0/0 for IPv4 or ::/0 for IPv6), or you can scope the route to a narrower range of IP addresses; for example, the public IPv4 addresses of your company’s public endpoints outside of AWS, or the Elastic IP addresses of other Amazon EC2 instances outside your VPC. If your subnet is associated with a route table that has a route to an internet gateway, it's known as a public subnet.

To enable communication over the internet for IPv4, your instance must have a public IPv4 address or an Elastic IP address that's associated with a private IPv4 address on your instance. Your instance is only aware of the private (internal) IP address space defined within the VPC and subnet. The internet gateway logically provides the one-to-one NAT on behalf of your instance, so that when traffic leaves your VPC subnet and goes to the internet, the reply address field is set to the public IPv4 address or Elastic IP address of your instance, and not its private IP address. Conversely, traffic that's destined for the public IPv4 address or Elastic IP address of your instance has its destination address translated into the instance's private IPv4 address before the traffic is delivered to the VPC.

To enable communication over the internet for IPv6, your VPC and subnet must have an associated IPv6 CIDR block, and your instance must be assigned an IPv6 address from the range of the subnet. IPv6 addresses are globally unique, and therefore public by default.

In the following diagram, Subnet 1 in the VPC is associated with a custom route table that points all internet-bound IPv4 traffic to an internet gateway. The instance has an Elastic IP address, which enables communication with the internet.


                Using an internet gateway
            

## VPC and Subnets

|  |  |  |
| --- | --- | --- |
| **Resource** | **Default limit** | **Comments** |
| VPCs per Region | 5 | The limit for internet gateways per Region is directly correlated to this one. Increasing this limit increases the limit on internet gateways per Region by the same amount. |
| Subnets per VPC | 200 | - |
| IPv4 CIDR blocks per VPC | 5 | This limit is made up of your primary CIDR block plus 4 secondary CIDR blocks. |
| IPv6 CIDR blocks per VPC | 1 | This limit cannot be increased. |
| Network interfaces per Region | 350 | This limit is the greater of either the default limit (350) or your On-Demand Instance limit multiplied by 5. The default limit for On-Demand Instances is 20. If your On-Demand Instance limit is below 70, the default limit of 350 applies. To increase this limit, submit a request or increase your On-Demand Instance limit. |

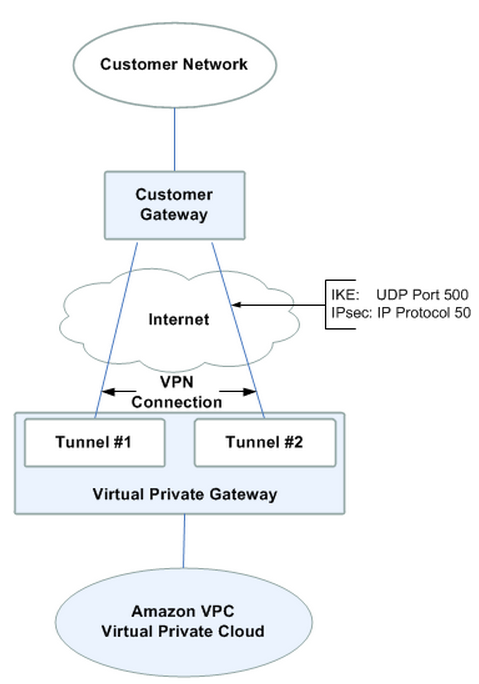
## Security Groups

|  |  |  |
| --- | --- | --- |
| **Resource** | **Default limit** | **Comments** |
| VPC security groups per Region | 2500 | The maximum is 10000. If you have more than 5000 security groups in a Region, we recommend that you paginate calls to describe your security groups for better performance. |
| Inbound or outbound rules per security group | 60 | You can have 60 inbound and 60 outbound rules per security group (making a total of 120 rules). This limit is enforced separately for IPv4 rules and IPv6 rules; for example, a security group can have 60 inbound rules for IPv4 traffic and 60 inbound rules for IPv6 traffic. A rule that references a security group or preflix list ID counts as one rule for IPv4 and one rule for IPv6.  A limit change applies to both inbound and outbound rules. This limit multiplied by the limit for security groups per network interface cannot exceed 300. For example, if you increase this limit to 100, we decrease the limit for your number of security groups per network interface to 3. |
| Security groups per network interface | 5 | The maximum is 16. The limit for security groups per network interface multiplied by the limit for rules per security group cannot exceed 300. For example, if you increase this limit to 10, we decrease the limit for your number of rules per security group to 30. |

Virtual private gateways

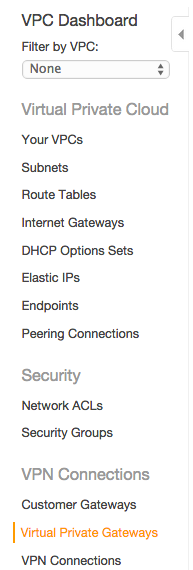
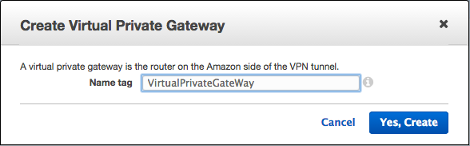
We want to use an optional Amazon VPC VPN connection that links our network to Amazon VPC virtual private cloud (VPC). A **customer gateway** is the anchor on the outer side of that connection. It can be a physical or software appliance. The anchor on the AWS side of the VPN connection is called a **virtual private gateway**.

To setup **VPN**, we need to have **Customer Gateway** which requires **Virtual Private Gateway**since as shown in the following diagram, the customer gateway, the VPN connection goes to the virtual private gateway, and the VPC.



Picture source: [Your Customer Gateway](http://docs.aws.amazon.com/AmazonVPC/latest/NetworkAdminGuide/Introduction.html)

Navigate to the VPC-2 dashboard in Region-2. Click on **Virtual Private Gateways** and then **Create Virtual Private Gateway**.

Select the virtual private gateway that we've just created and select **Attach to VPC-2**.