

INSY 5345 & INSY 4307

Dr. Santoso Budiman

# Cloud Computing Theory and Practice

The background of the slide features a stylized, low-poly illustration of various shapes in shades of blue, white, and light grey. These shapes are arranged to look like large, fluffy clouds in a clear blue sky. Some shapes overlap, creating a sense of depth and texture.

# Networking

---

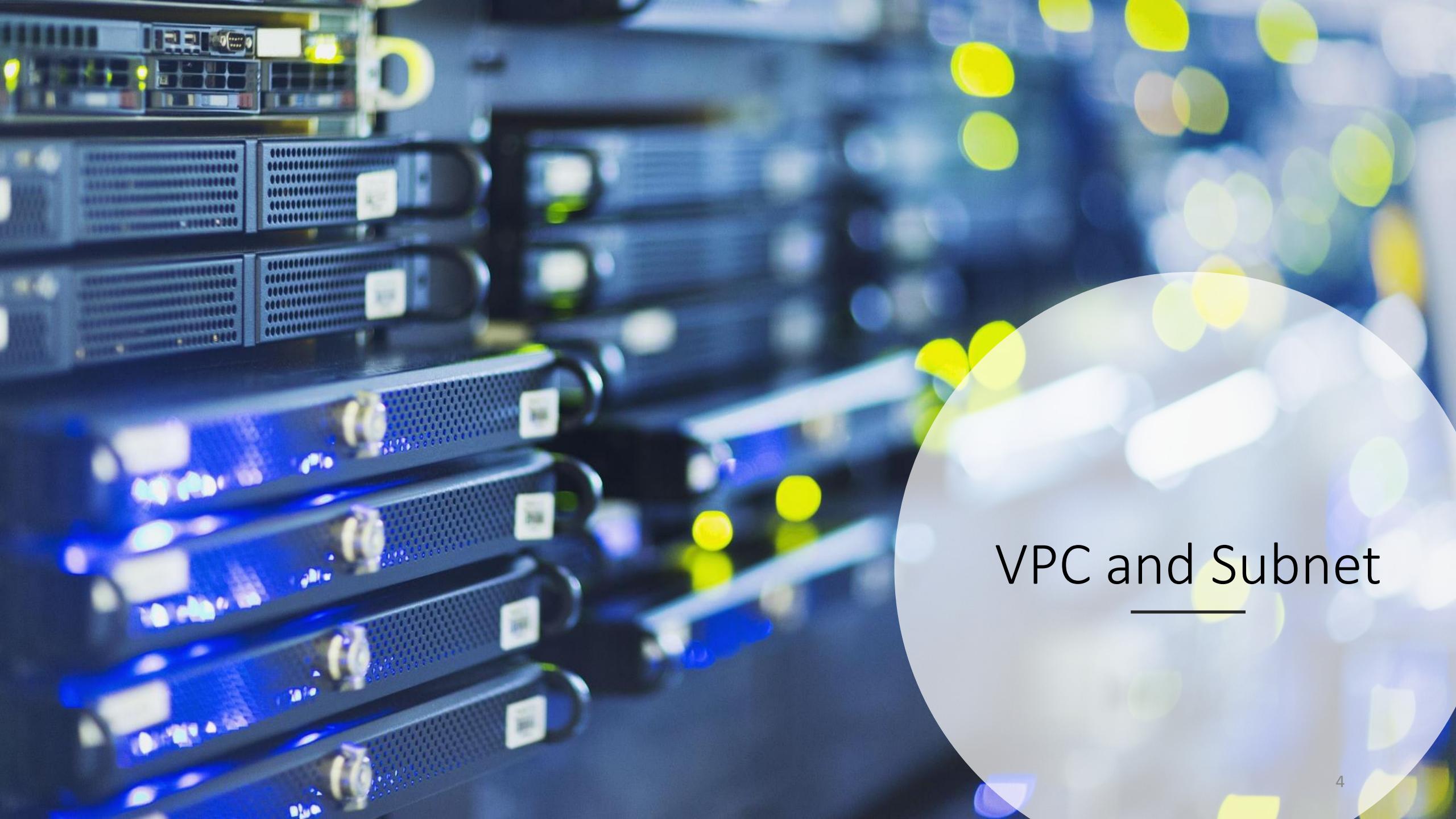
Section 4



# Topics

---

- VPC and Subnet
- Connecting AWS networking environment to the internet
  - Gateways
    - Internet Gateway
    - NAT Gateway (Outbound Internet Traffic only for IPv4)
  - Routers and Routing Tables
  - Elastic IP Address – Elastic Network Interface
- VPC End Points
- Network Security
- Homework



# VPC and Subnet

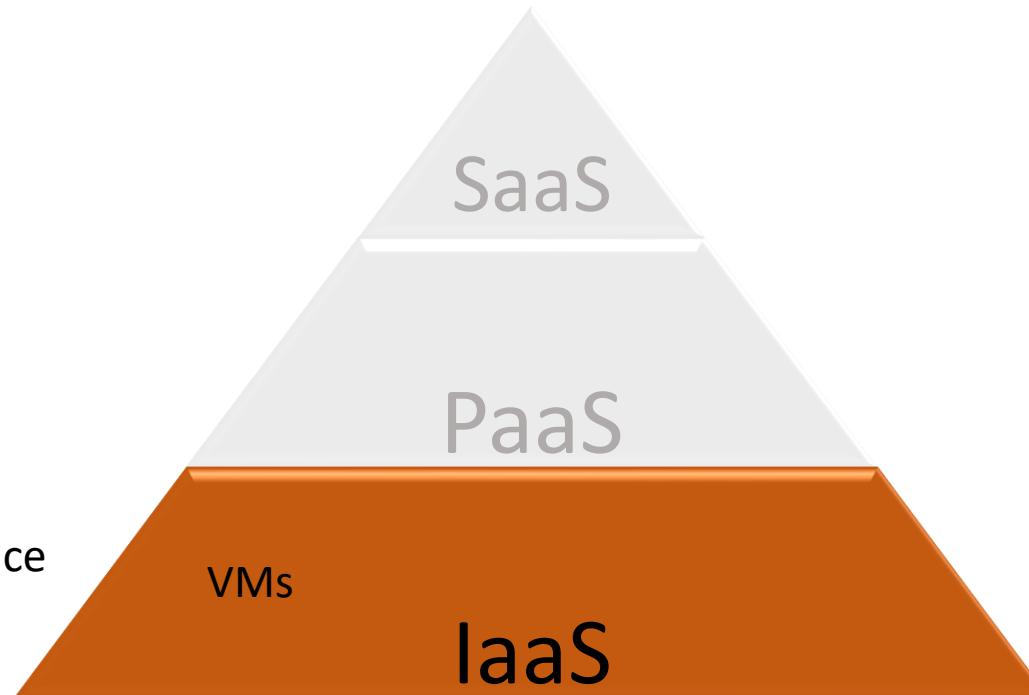
---

# Cloud Computing – Service Model

SaaS = Software as a Service

PaaS = Platform as a Service

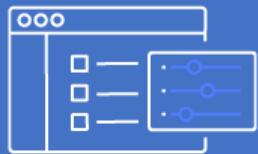
IaaS = Infrastructure as a Service



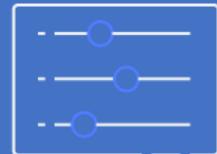
# Amazon VPC

Provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define.

Bring your own network



IP Addresses



Subnets



Routing rules



Network  
configuration



Security rules

# AWS Virtual Private Cloud

## Default (**DON'T DELETE** default VPC and its SUBNETS)

- 1 VPC per AWS Region
- CIDR - /16
- A subnet per AZ (/20)
- An internet Gateway is attached

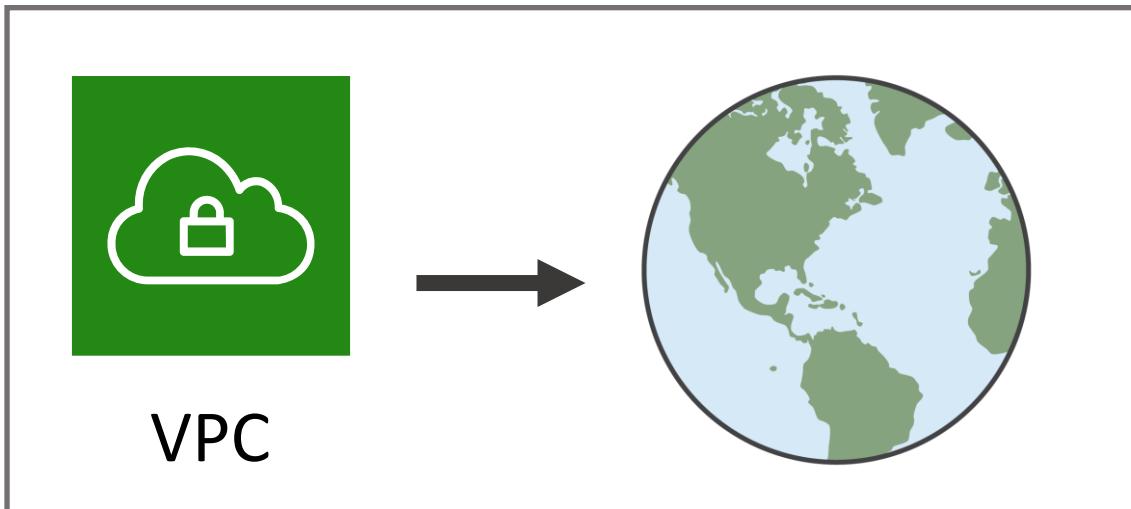
## Customized

- VPC must be in a region
- Define CIDR - /16 - /28
- Can create more than 1 VPC in a region
- Internet Gateway is optional

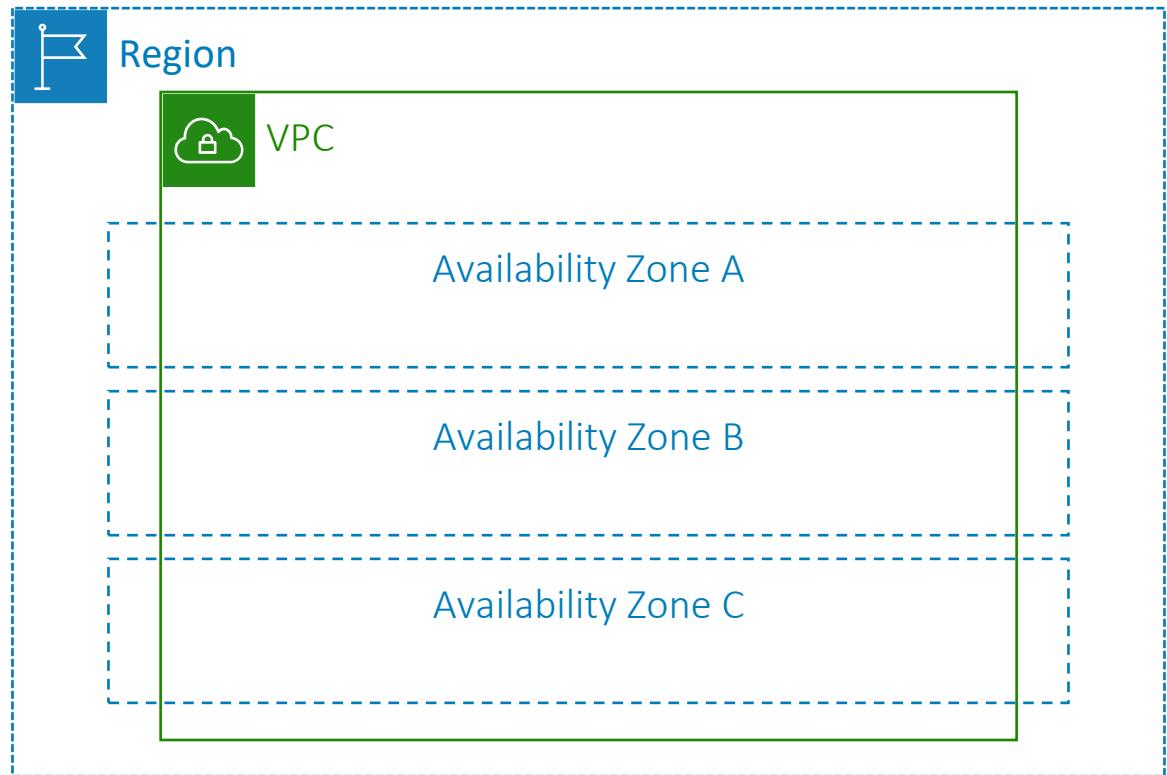
Note, in default VPC, just like in customized VPC, you can:

- Add additional nondefault subnets.
- Modify the main route table.
- Add additional route tables.

# VPC deployment



You can deploy a VPC in any AWS Region.  
Can create multiple VPCs and multiple subnets in a Region (there are quotas).  
A VPC is limited to one region (can not span to other regions)



A VPC can host supported resources from any Availability Zone within its Region.

# Classless Inter-Domain Routing (CIDR)

0.0.0.0/0	= All IP addresses
10.22.33.44/32	= 10.22.33. <b>44</b>
10.22.33.0/24	= 10.22.33.*
10.22.0.0/16	= 10.22.*.*

CIDR	Total IP addresses
/28	16
...	...
/20	4,096
/19	8,192
/18	16,384
/17	32,768
/16	65,536

# CIDR IPv4 Block and Available Private IP addresses

CIDR Block	Number of IP Addresses
/28	$2^{(32-28)}=16$
/27	$2^{(32-27)}=32$
/26	$2^{(32-26)}=64$
:	
:	
/16	$2^{(32-16)}=65536$

Note: some can not be used (reserved).

- IPv4 - 32 bits (4 octets)
- The Internet Assigned Numbers Authority (IANA) reserves three blocks of the IP address space for private internets (RFC 1918 private IP addresses)
  - 10.0.0.0 - 10.255.255.255 (10/8 prefix)
  - 172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
  - 192.168.0.0 - 192.168.255.255 (192.168/16 prefix)
- <http://www.faqs.org/rfcs/rfc1918.html>
- AWS allows block sizes between /16 and /28 (**More restrictive than IANA**)
  - [https://docs.aws.amazon.com/vpc/latest/userguide/VPC\\_Subnets.html](https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Subnets.html)
- AWS - 5 IP addresses in each subnet (CIDR block) are reserved (**the first 4 IP addresses and the last**). For example, 10.0.0.0/24:
  - 10.0.0.0 – Network address
  - 10.0.0.1 – Reserved for VPC router
  - 10.0.0.2 – Reserved for DNS server (for VPC network range), reserved by AWS otherwise
  - 10.0.0.3- Reserved by AWS for future use
  - 10.0.0.255 – Network broadcast address
- **Can not alter CIDR size after creation**

# AWS is more restrictive than IANA

RFC 1918 range	Example CIDR block
10.0.0.0 - 10.255.255.255 (10/8 prefix)	Your VPC must be /16 or smaller, for example, 10.0.0.0/16.
172.16.0.0 - 172.31.255.255 (172.16/12 prefix)	Your VPC must be /16 or smaller, for example, 172.31.0.0/16.
192.168.0.0 - 192.168.255.255 (192.168/16 prefix)	Your VPC can be smaller, for example 192.168.0.0/20.

[https://docs.aws.amazon.com/vpc/latest/userguide/VPC\\_Subnets.html](https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Subnets.html)

Ex: a VPC with CIDR block 10.0.0.0/24 supports 256 IP addresses. Can break this CIDR block into two subnets (each 128 IP addresses (including reserved)).

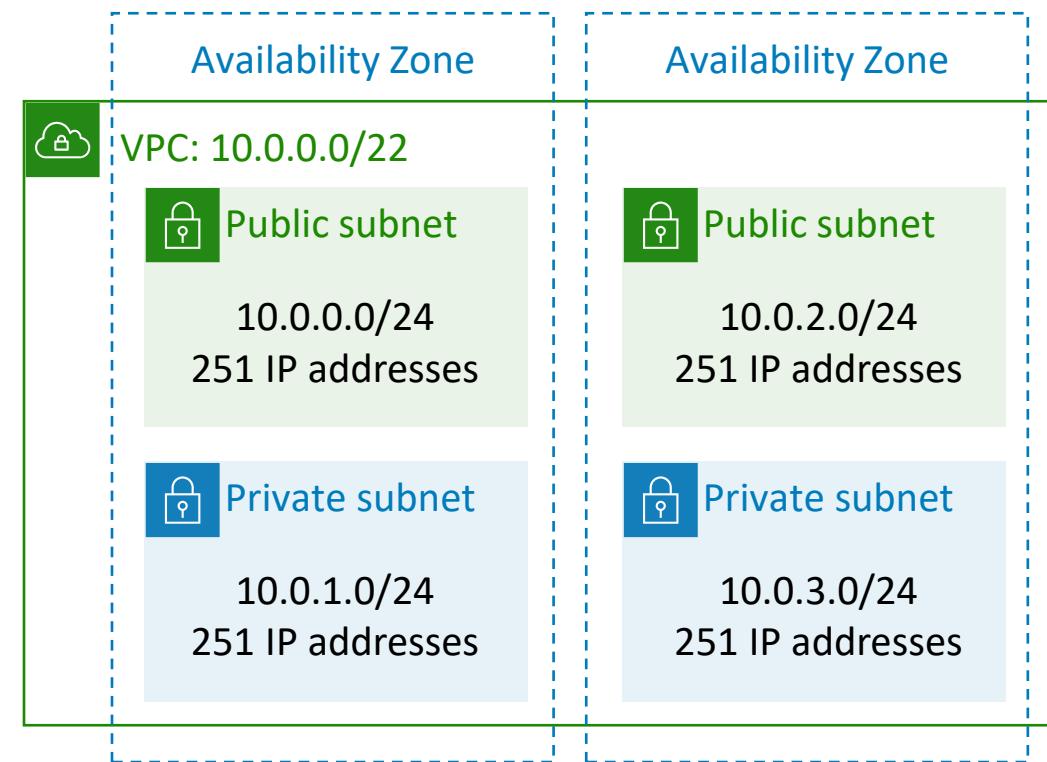
- One subnet uses CIDR block 10.0.0.0/25 (for addresses 10.0.0.0 - 10.0.0.127)
- the other uses CIDR block 10.0.0.128/25 (for addresses 10.0.0.128 - 10.0.0.255)

The CIDR block of a subnet can be:

- the same as the CIDR block for the VPC (for a single subnet in the VPC), or
  - a subset of the CIDR block for the VPC (for multiple subnets).
- The allowed block size is between a /28 netmask and /16 netmask.
  - The CIDR blocks of the subnets of a VPC cannot overlap.

# Subnets: Dividing your VPC

- A **subnet** is a segment or partition of a VPC's IP address range where you can allocate a group of resources
- Subnets are **not isolation boundaries**
- Subnets are a **subset** of the VPC CIDR block
- Subnet CIDR blocks **cannot overlap**
- Each subnet resides entirely within one Availability Zone
- You can add one or more subnets in each Availability Zone or in a Local Zone
- AWS **reserves five IP addresses** in each subnet



Example: A VPC with **CIDR /22** includes 1,024 total IP addresses.

# Default VPC – Region N. Virginia

vpc-472bcd3a - 172.31.0.0/16

172.31.48.0/20

172.31.80.0/20

172.31.32.0/20

172.31.16.0/20

172.31.0.0/20

172.31.64.0/20

		Octet 4										Octet 3										Octet 2										Octet 1							
	Bit value	128	64	32	16	8	4	2	1		128	64	32	16	8	4	2	1		128	64	32	16	8	4	2	1		128	64	32	16	8	4	2	1			
VPC	172.31.0.0/16	1	0	1	0	1	1	0	0		0	0	0	1	1	1	1	1		0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0			
Subnet	172.31.48.0/20	1	0	1	0	1	1	0	0		0	0	0	1	1	1	1	1		0	0	1	1	0	0	0	0		0	0	0	0	0	0	0	0			
Subnet	172.31.80.0/20	1	0	1	0	1	1	1	0		0	0	0	1	1	1	1	1		0	1	0	1	0	0	0	0		0	0	0	0	0	0	0	0			
Subnet	172.31.32.0/20	1	0	1	0	1	1	1	0		0	0	0	1	1	1	1	1		0	0	1	0	0	0	0	0		0	0	0	0	0	0	0	0			
Subnet	172.31.16.0/20	1	0	1	0	1	1	1	0		0	0	0	1	1	1	1	1		0	0	0	1	0	0	0	0		0	0	0	0	0	0	0	0			
Subnet	172.31.0.0/20	1	0	1	0	1	1	1	0		0	0	0	1	1	1	1	1		0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0			
Subnet	172.31.64.0/20	1	0	1	0	1	1	1	0		0	0	0	1	1	1	1	1		0	1	0	0	0	0	0	0		0	0	0	0	0	0	0	0			

- Defined by AWS (172.31.0.0/16)
- Comply with IANA - **172.16. 0.0 - 172.31. 255.255 (172.16/12 prefix)**
- 16 bits from the left to identify the network (/16)
- Bit 17-Bit20 from the left to identify the subnet within that network (between /16 of VPC and /20 of subnets)
  - You could have  $2^4 = 16$  subnets
- The rest 12 bits (32-20) to identify the hosts in each subnet
  - Total host per subnet =  $2^{12} - 5 = 4091$  (AWS reserves 5)

1 bit	0			$2^1$	2
	1				
2 bits	0	0		$2^2$	4
	0	1			
	1	0			
	1	1			
3 bits	0	0	0	$2^3$	8
	0	0	1		
	0	1	0		
	0	1	1		
	1	0	0		
	1	0	1		
	1	1	0		
	1	1	1		

**IPv4 - 32 bits (4 octets)**

AWS is more restrictive than IANA

AWS allows block sizes between /16 and /28

[https://docs.aws.amazon.com/vpc/latest/userguide/VPC\\_Subnets.html](https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Subnets.html)

The Internet Assigned Numbers Authority (IANA) reserves three blocks of the IP address space for private internets (RFC 1918 private IP addresses)

**10.0.0.0 - 10.255.255.255 (10/8 prefix)**

**172.16. 0.0 - 172.31. 255.255 (172.16/12 prefix)**

**192.168. 0.0 - 192.168. 255.255 (192.168/16 prefix)**

# VPC design best practices

- Create one subnet per available Availability Zone for each group of hosts that have unique routing requirements.
- Divide your VPC network range evenly across all available Availability Zones in a Region.
- Do not allocate all network addresses at once. Instead, ensure that you reserve some address space for future use.
- Size your VPC CIDR and subnets to support significant growth for the expected workloads.
- Ensure that your VPC network range (CIDR block) does not overlap with your organization's other private network ranges.

# Single VPC deployment

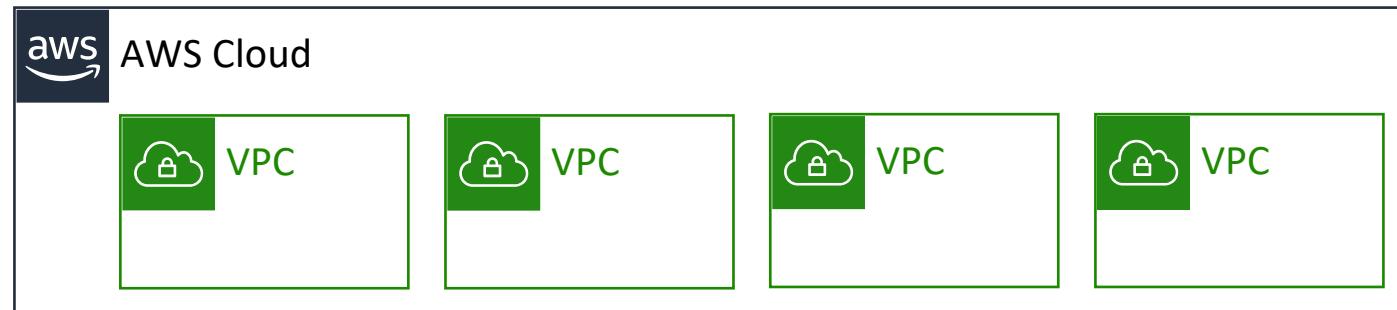
There are limited use cases where deploying [one VPC](#) might be appropriate:

- Small, single applications managed by a small team
- High performance computing (HPC)
- Identity management

For [most](#) use cases, there are two primary patterns for organizing your infrastructure: multi-VPC and multi-account.

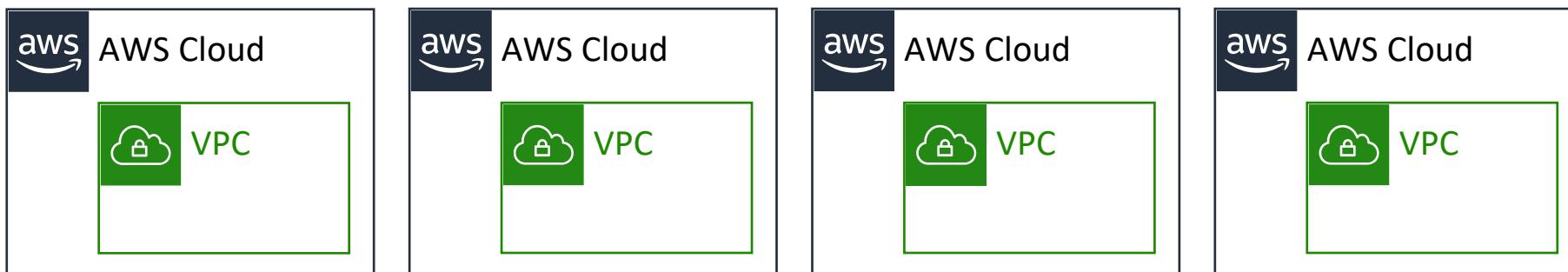
# Multiple VPCs

- Best suited for –
  - Single team or single organizations, such as managed service providers
  - Limited teams, which makes it easier to maintain standards and manage access
- Exception –
  - Governance and compliance standards might require greater workload isolation regardless of organizational complexity



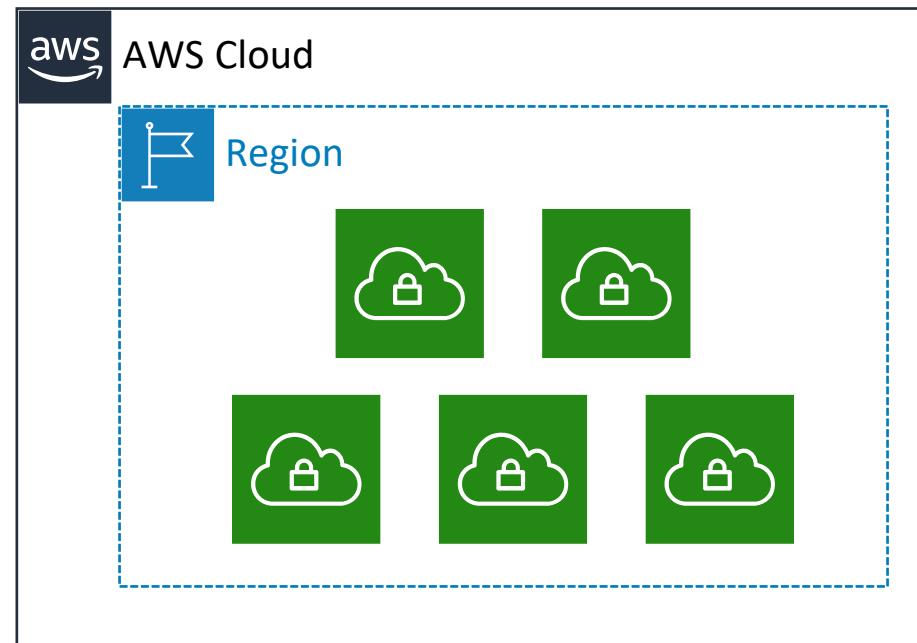
# Multiple accounts

- Best suited for –
  - Large organizations and organizations with multiple IT teams
  - Medium-sized organizations that anticipate rapid growth
- Why?
  - It can be more challenging to manage access and standards in more complex organizations



# Amazon VPC quotas

Default quota: 5 VPCs per Region per account \*



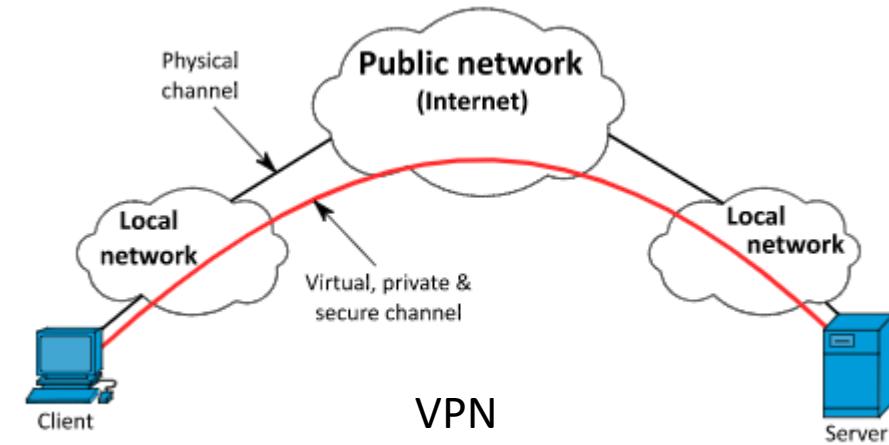
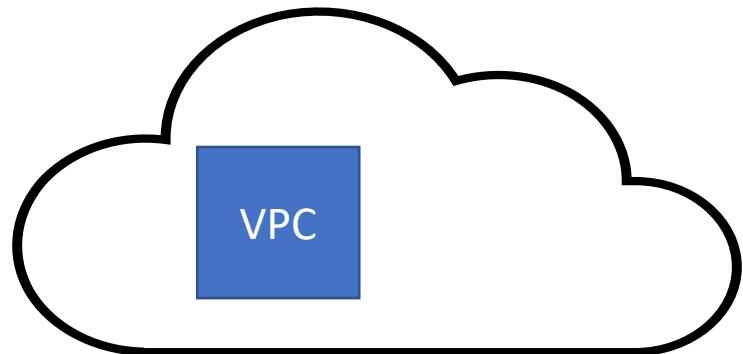
\* The default quota is 5 VPCs per Region, but you can request a quota increase.

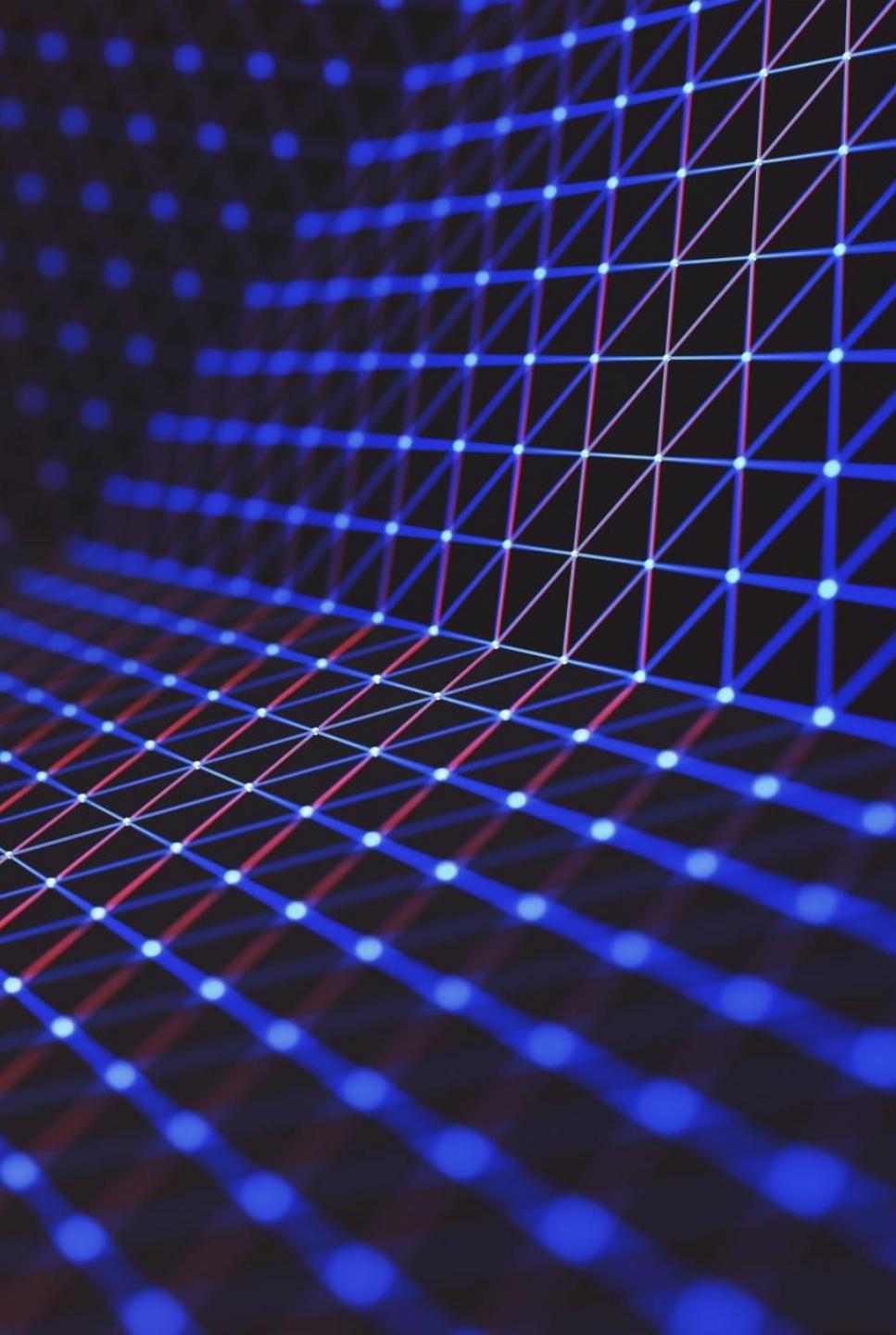
# AWS Domain Name System

- An instance launched into a **default VPC** has
  - a public DNS hostname (corresponds to the public IPv4 address)
  - a private DNS hostname (corresponds to the private IPv4 address)
- An instance launched into a **custom VPC**:
  - a private DNS hostname
  - may have a public DNS hostname depending on the DNS attributes specified in VPC (both must be true):
    - EnableDnsHostnames – Indicates whether the instances launched in VPC get public DNS hostnames. If this attribute is true, instances in VPC get public DNS hostnames, but only if the enableDnsSupport attribute is also set to true.
    - enableDnsSupport – Indicates whether DNS resolution is supported for the VPC.

# Virtual Private Cloud (VPC) and Virtual Private Network (VPN)

- VPC is not VPN
- VPC is a **virtual stand-alone network** – logically isolated network in the cloud
- Virtual Private Network (**VPN**) - a **private network across the internet**, it enables users to exchange data across shared or public networks.
  - Multiple networks (including VPCs) can be connected to form a VPN





# VPC key takeaways

---

Amazon VPC enables you to provision VPCs, which are logically isolated sections of the AWS Cloud where you can launch your AWS resources.

---

A VPC belongs to only one Region and is divided into subnets.

---

A subnet belongs to one Availability Zone or Local Zone. It is a subset of the VPC CIDR block.

---

You can create multiple VPCs in the same Region or in different Regions, and in the same account or different accounts.

---

Follow best practices when you design your VPC.

A close-up photograph of a network switch or hub with multiple ports. Many white Ethernet cables are plugged in, and their blue and yellow RJ-45 connectors are visible. Each port has a small green LED indicator light above it, which is illuminated for connected ports. The ports are labeled with numbers such as 1, 5, 11, 15, 17, 18, 20, 22, 24, 26, 28, 30, 31, and 32. Some ports have small labels like "Link" and "Mode".

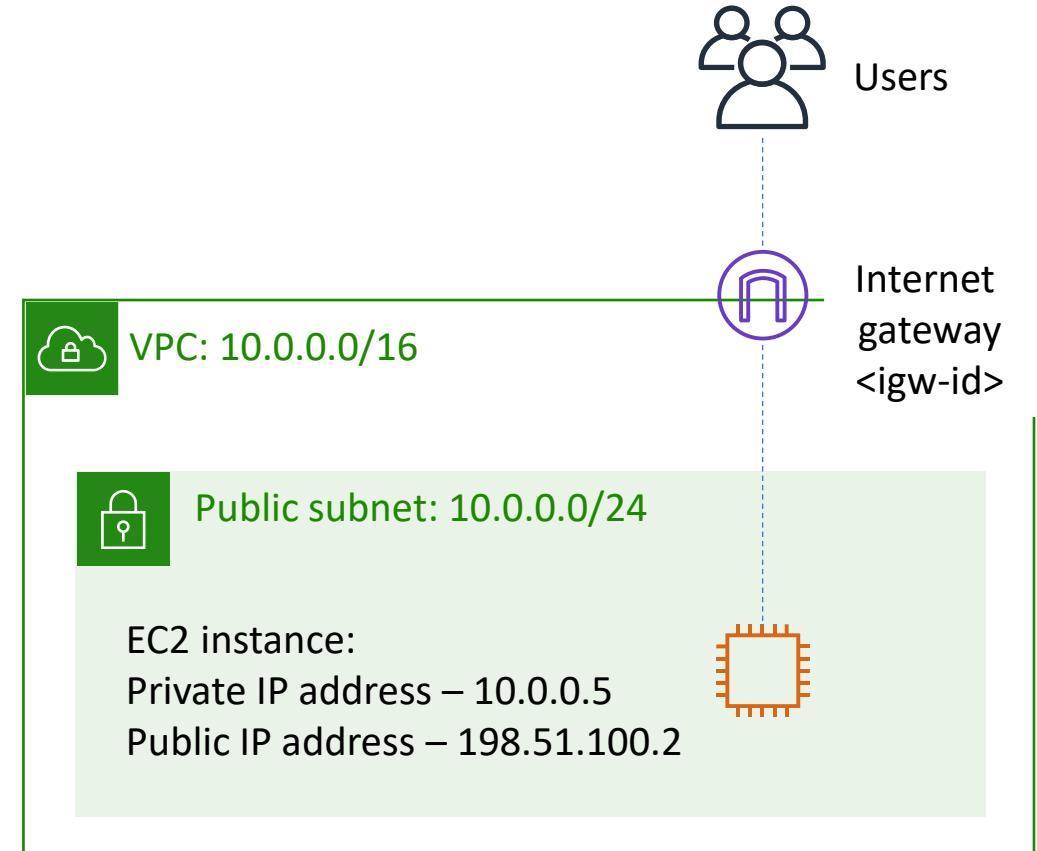
Connecting AWS  
networking  
environment to the  
internet

# Creating a public subnet



## Internet gateways

- Allow communication between resources in your VPC and the internet
- Are horizontally scaled, redundant, and highly available by default
- Provide a target in your subnet route tables for internet-routable traffic

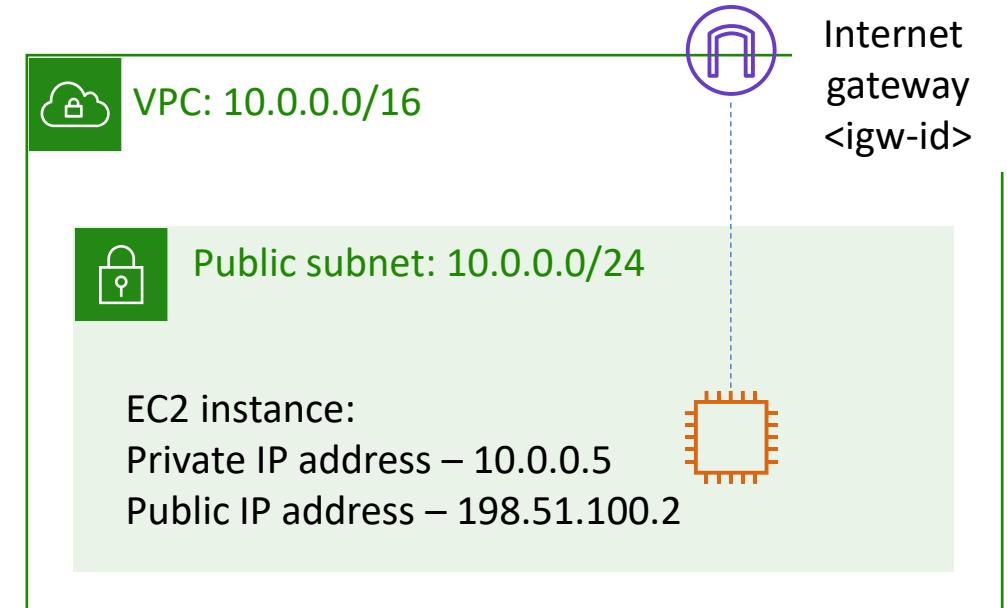


# Directing traffic between VPC resources

- **Route tables** are required to direct traffic between VPC resources
- Each VPC has a **main (default)** route table
- All subnets **must** be associated with a route table
- You can create **custom** route tables
- **Public subnet** – if has route to internet gateway

**Best practice:** Use custom route tables for each subnet.

Default and custom route tables have **immutable local route to the VPC**



Public route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

# Elastic IP Address & Elastic Network Interface

# Elastic IP Address (EIP)

- An **Elastic IP (EIP)** address is a **reserved static IPv4** address (AWS currently does not support IPv6)
- An Elastic IP address is **associated with a user's AWS account**.
- With an **Elastic IP address**, user can mask the failure of an instance or software by **remapping** the address to another instance.
- An Elastic IP address is a public IPv4 address, which is reachable from the internet.
  - **Public IP** addresses are dynamic - i.e., if user stop/start an instance a new **public IP is reassigned**.
  - **Elastic IPs** get allocated to a user-account, and it will **remain the same**.
- 3 steps to use an EIP:
  - Allocate one EIP to a user's account from the console
  - Associate the EIP either with a user's instance or with a network interface
  - Start using it
- <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html>

# Compare the Public IP addresses before and after the “stop” - Before

This screenshot shows the AWS CloudWatch Metrics console. At the top, there's a search bar with 'Budiman1' and a 'Launch Instance' button. Below the search bar is a table header with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IPv4 Public IP. A single row is selected, showing 'Budiman1' with an Instance ID of 'i-08379a11707e95bfb', an 't2.micro' Instance Type, 'us-east-1b' Availability Zone, 'running' Instance State, 2/2 status checks, 'None' alarm status, a Public DNS name of 'ec2-18-206-125-134.co...', and a Public IP of '18.206.125.134'. The table has a dark header and light gray rows.

This screenshot shows the AWS CloudWatch Metrics console. It's identical to the first one, but a context menu is open over the 'Actions' button. The menu options are: Connect, Get Windows Password, Create Template From Instance, Launch More Like This, Instance State, Instance Settings, Image, Networking, and CloudWatch Monitoring. The 'Instance State' option is expanded, showing 'Start', 'Stop' (which is highlighted in orange), 'Stop - Hibernate', 'Reboot', and 'Terminate'. The main table below is identical to the first one, showing the instance is still running.

This screenshot shows the AWS CloudWatch Metrics console. It's identical to the first one, but a context menu is open over the 'Actions' button. The menu options are: Connect, Get Windows Password, Create Template From Instance, Launch More Like This, Instance State, Instance Settings, Image, Networking, and CloudWatch Monitoring. The 'Instance State' option is expanded, showing 'Start', 'Stop' (which is highlighted in orange), 'Stop - Hibernate', 'Reboot', and 'Terminate'. The main table below is identical to the first one, showing the instance is still running.

# Compare the Public IP addresses before and after the “stop” - After

The screenshot shows the AWS EC2 console interface. In the top navigation bar, the 'Actions' button is highlighted. A dropdown menu is open over the instance 'Budiman1'. The menu items include: Connect, Get Windows Password, Create Template From Instance, Launch More Like This, Instance State (with 'Start' selected), Instance Settings, Image, Networking, and CloudWatch Monitoring. Below the menu, the main EC2 dashboard displays the instance details: Name: Budiman1, Instance ID: i-08379a11707e95bfb, Instance Type: t2.micro, Availability Zone: us-east-1b, Status: running, Status Checks: Initializing, Alarm Status: None, Public DNS (IPv4): ec2-3-89-8-22.compute..., and Public IP: 3.89.8.22.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
Budiman1	i-08379a11707e95bfb	t2.micro	us-east-1b	running	Initializing	None	ec2-3-89-8-22.compute...	3.89.8.22

# Elastic Network Interface

- A network interface is **the point of interconnection** between a **computer** and a **private or public network**.
- **All EC2 has primary Network Interface (eth0).**
  - **Can not change the default network interface** of any instance (primary network interface).
  - But can create additional network interface.
- An **elastic network interface (ENI)** is a **logical networking component** in a VPC that represents a virtual network card.
  - ENI gives users the ability to create one or more network interfaces and attach them to instances.
  - Users can detach ENI from an instance and attach to another (Elastic), redirecting the network traffic and all attributes of the ENI
  - The number of ENIs that can be attached to a particular instance varies depending on the instance type.
  - ENI doesn't impact the network bandwidth to the instance. For example, adding an ENI does not increase or double network bandwidth.

# Elastic Network Interface Attributes

- An elastic network interface is a logical networking component in a VPC that represents a **virtual network card (ENI)**.
- An ENI can have the following attributes:
  - A **MAC addresses**
  - One public IPv4 address
  - One or more IPv6 addresses
  - A primary private IPv4 address
  - One or more secondary private IPv4 addresses
  - One elastic IP address (IPv4) per private IPv4 address
  - One or more security groups
  - A source/destination check flag and description
- <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html>

# EC2-Network Interface

Screenshot of the AWS EC2 Instance Details page for instance i-05c5c9a4dbfa40f21 (budiman1).

Instance: i-05c5c9a4dbfa40f21 (budiman1) Public DNS: ec2-54-152-3-31.compute-1.amazonaws.com

Description tab selected.

Attribute	Value
Instance ID	i-05c5c9a4dbfa40f21
Instance state	running
Instance type	t2.micro
Finding	You may not have permission to access AWS Compute Optimizer.
Private DNS	ip-172-31-60-225.ec2.internal
Private IPs	172.31.60.225
Secondary private IPs	
VPC ID	vpc-04230f7e
Subnet ID	subnet-c39abef0
Network interfaces	eth0
IAM role	-
Key pair name	EC2testkeypair1
Owner	608910113900
Launch time	June 14, 2020 at 5:27:15 PM UTC-5
Termination protection	False
Lifecycle	normal
Monitoring	basic
Alarm status	None
Kernel ID	-
RAM disk ID	-
Placement group	-
Partition number	-

**Network Interface eth0** details:

Attribute	Value
Interface ID	eni-00c6d79de26b4e189
VPC ID	vpc-04230f7e
Attachment Owner	608910113900
Attachment Status	attached
Attachment Time	Sun Jun 14 17:27:15 GMT-500 2020
Delete on Terminate	true
Private IP Address	172.31.60.225
Private DNS Name	ip-172-31-60-225.ec2.internal
Public IP Address	54.152.3.31
Source/Dest. Check	true
Description	-
Security Groups	launch-wizard-4
Elastic Fabric Adapter	Disabled

IPv4 Public IP: 54.152.3.31

IPv6 IPs: -

Elastic IPs: -

Availability zone: us-east-1e

Security groups: launch-wizard-4, view inbound rules, view outbound rules

Scheduled events: No scheduled events

AMI ID: ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20200408 (ami-085925f297f89fce1)

Platform details: Linux/UNIX

Usage operation: RunInstances

Source/dest. check: True

T2/T3 Unlimited: Disabled

EBS-optimized: False

Root device type: ebs

Root device: /dev/sda1

Block devices: /dev/sda1

Elastic Graphics ID: -

Elastic Inference accelerator ID: -

Capacity Reservation: -

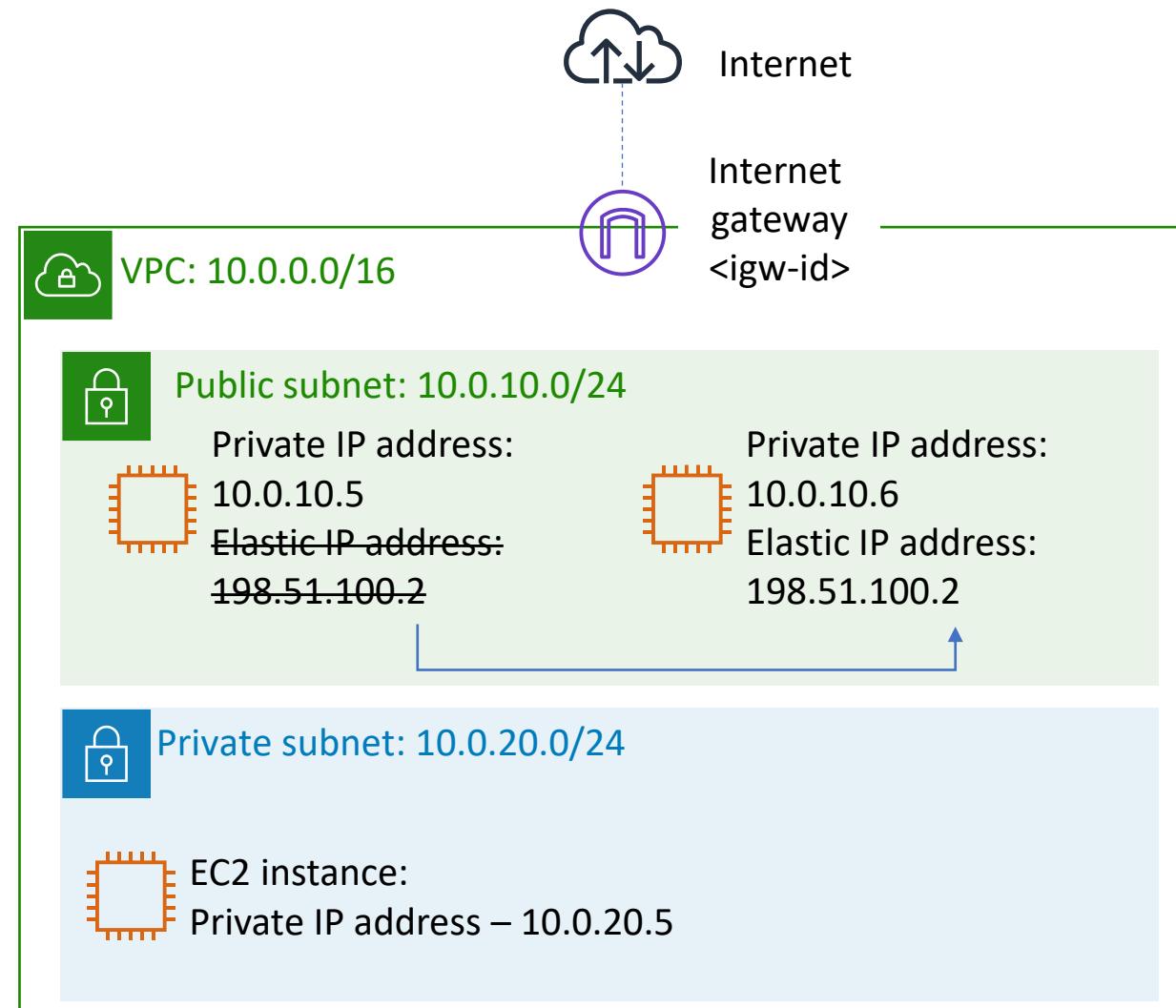
Capacity Reservation Settings: Open

Outpost Arn: -

# Remapping an IP address from one instance to another

## → Elastic IP addresses

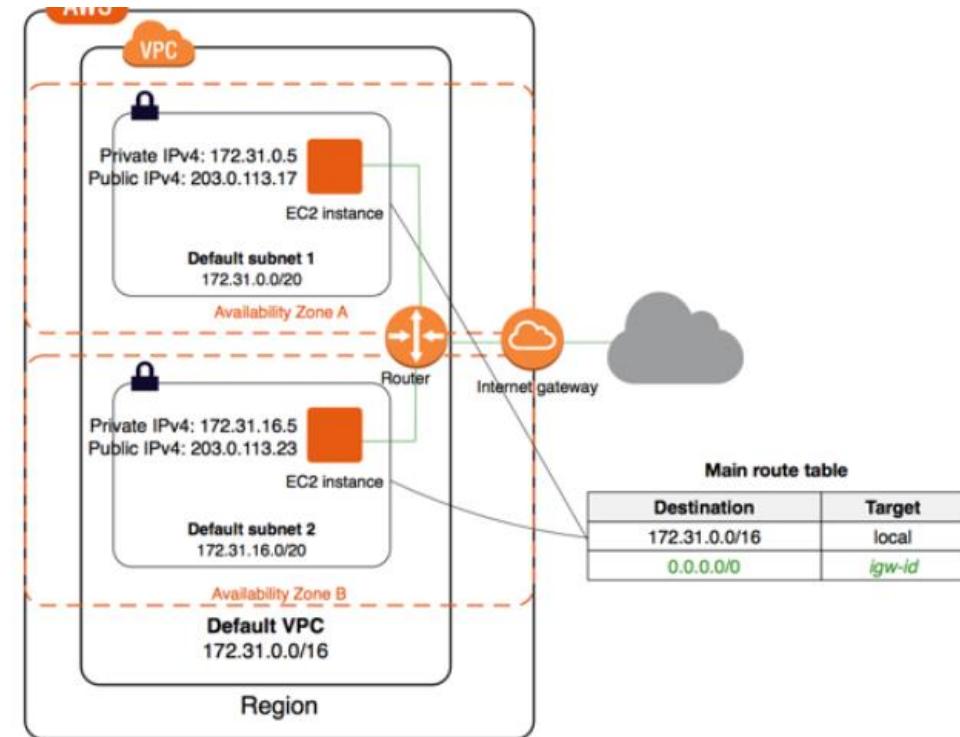
- Are static, public IPv4 addresses associated with your AWS account
- Can be associated with an instance or elastic network interface
- Can be remapped to another instance in your account
- Are useful for redundancy when load balancers are not an option



# Internet Gateway

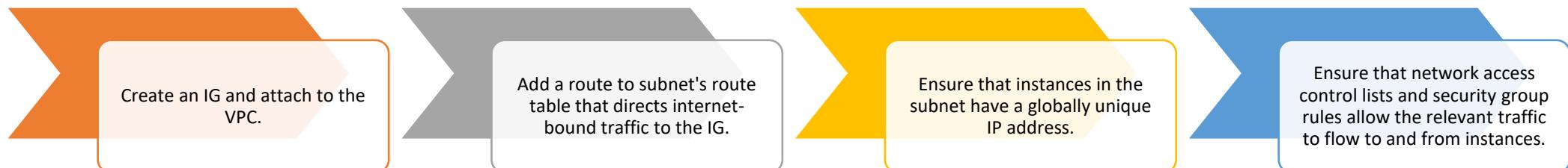
# Internet Gateway

- Internet Gateway (IG) – a component of VPC that **allows** a VPC to **communicate** with the **internet**.
  - **IG is set as a target in a subnet route table**
- IG is horizontally scaled, redundant, and highly available component in VPC
- **IG two purposes (inbound and outbound):**
  - **a target in the VPC route tables for internet-routable traffic (outbound)**
  - to perform network address translation (NAT) for instances that have been assigned public IPv4 addresses (**inbound**).
- **Must be defined for customized VPCs** (not automatically created as for default VPC).
- No charge.



# To enable internet access

To enable access to or from the internet for instances in a subnet in a VPC:



# NAT Devices

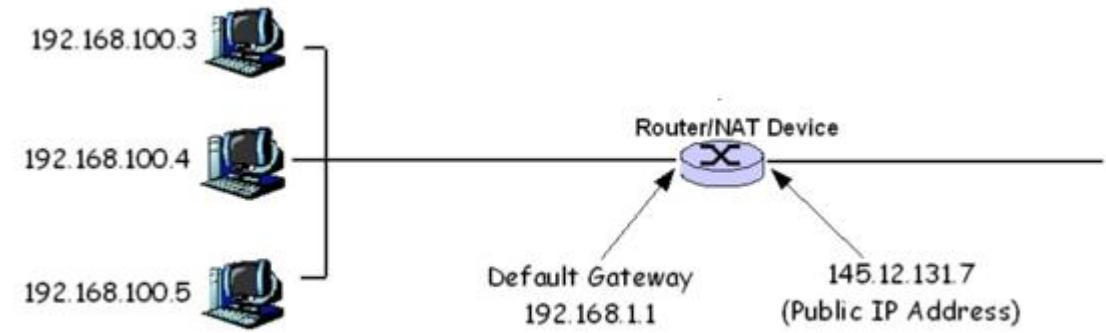
# NAT (Network Address Translation)

NAT translates external IP address to internal IP address and vice versa

NAT TYPES	DESCRIPTION	FOR	
Static NAT (SNAT)	<b>1 external</b> public address is mapped to <b>1 internal</b> public address. The external Ip address is static.	Incoming	Can serve both intranet and internet. Waste ext. IP address
Dynamic NAT (DNAT)	Mapping of a private IP address to a public IP address from a group of public IP addresses called as <b>NAT pool</b> .	Outgoing	Difficult external attack, keep changing public IP address.
Port Address Translation (PAT)	Use combination of ip address and non-reserved TCP/IP ports. <b>One public IP address – many internal hosts.</b>	Incoming and Outgoing	The main difference between DNAT and PAT, PAT use one public IP address, DNAT can use one or more.

# NAT (Network Address Translation)

- PAT is typically built-in to a Router
  - Benefits:
    - Improve Security – hide devices in the network from the outside world
    - Conserve external IP addresses.
- A Router has an external IP address.
- When a user calls bestbuy.com and bestbuy.com server sends information back to the router, the NAT translates the IP address to the internal/private IP address.
- In the subnet, each host has an internal IP address.
- With NAT not every host must have a globally unique IP address (reserving Public IP address).



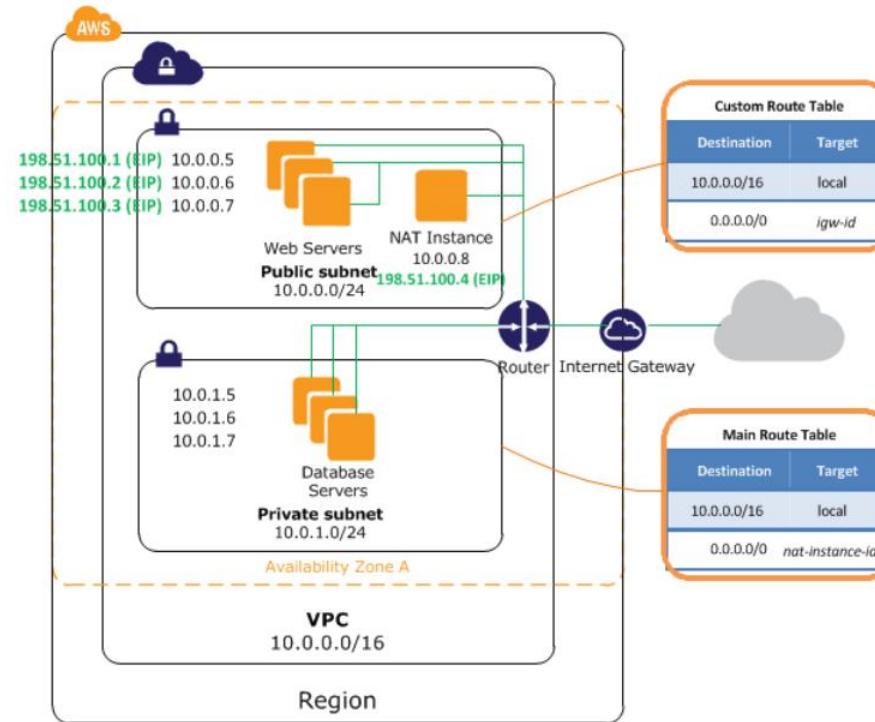
# AWS Network Address Translation

AWS NAT devices:

- A NAT device forwards traffic from instances in the private subnet to the Internet (**Outbound**) and then sends response to the instances – (similar to router)
  - No inbound
  - Purpose: to protect instances in Private subnet from the internet while allowing the instances to access it.
- When traffic goes to the internet, the source IPv4 address is replaced with the NAT device's address
- In the response traffic, the NAT device translates the address back to private IPv4 address
- Only for IPv4 traffic
- 2 types of NAT devices
  - NAT instances
  - NAT gateways
- NAT Devices (instance or gateway) still need **Internet Gateway** to connect to the internet.

# NAT Instances

- The main route table is associated with the private subnet and sends traffic from the instances in the private subnet to the NAT instance in the public subnet.
- The NAT instance sends the traffic to the **Internet gateway** for the VPC.
- The traffic is attributed to the Elastic IP address of the NAT instance.
- The NAT instance specifies a high port number for the response (**PAT**); if a response comes back, the NAT instance sends it to an instance in the private subnet based on the port number for the response.
- Involve more settings than NAT gateway.
- Use NAT AMI to create the NAT instance.
- Can use public or elastic IP address.

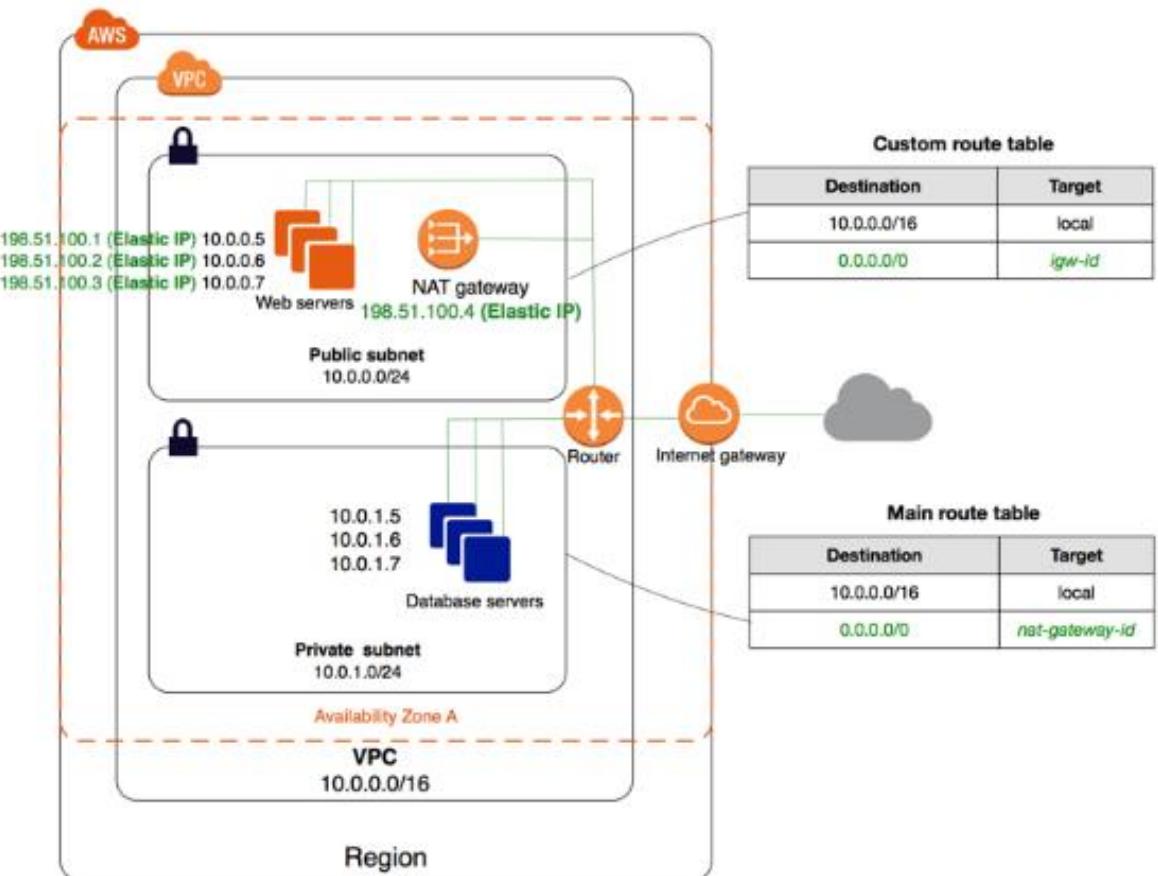


# NAT Gateway

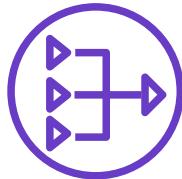
- A NAT gateway performs similarly to NAT instance
  - Managed service
  - NAT gateway is created in a specific AZ in a redundant fashion.
  - NAT gateway provides better availability and bandwidth.
  - EC2s in private subnets located in any AZ within the VPC can use it.
  - For HA purposes, you can create NAT Gateway in multiple Azs.
  - Use Elastic IP address.
  - More expensive than NAT instance but better features and performance.

<https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway.html#nat-gateway-basics>

<https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-comparison.html>



# Connecting private subnets to the internet



## NAT gateways

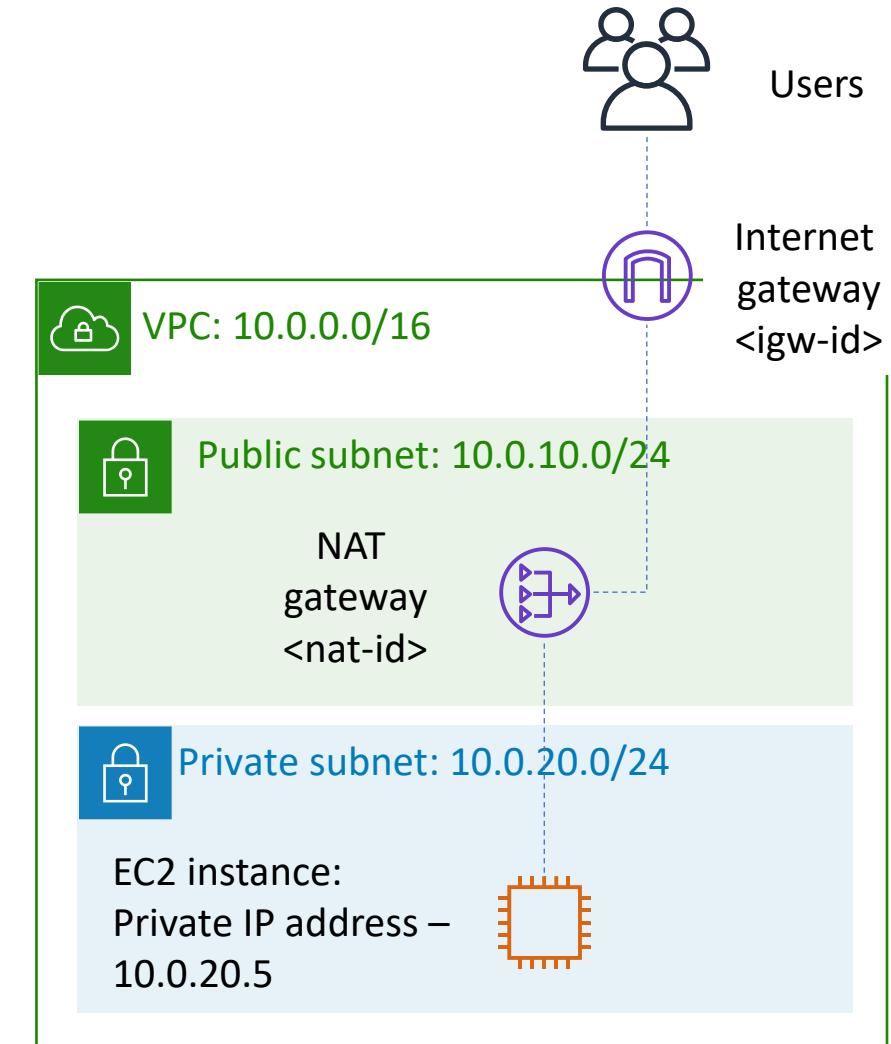
- Enable instances in a private subnet to initiate outbound traffic to the internet or other AWS services
- **Prevent** private instances from receiving **inbound** connection requests from the internet

Public route table

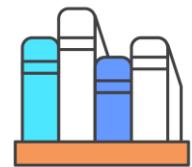
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Private route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<nat-id>



# Subnet use case examples



Data store instances



Private subnet



Batch-processing instances



Private subnet



Backend instances



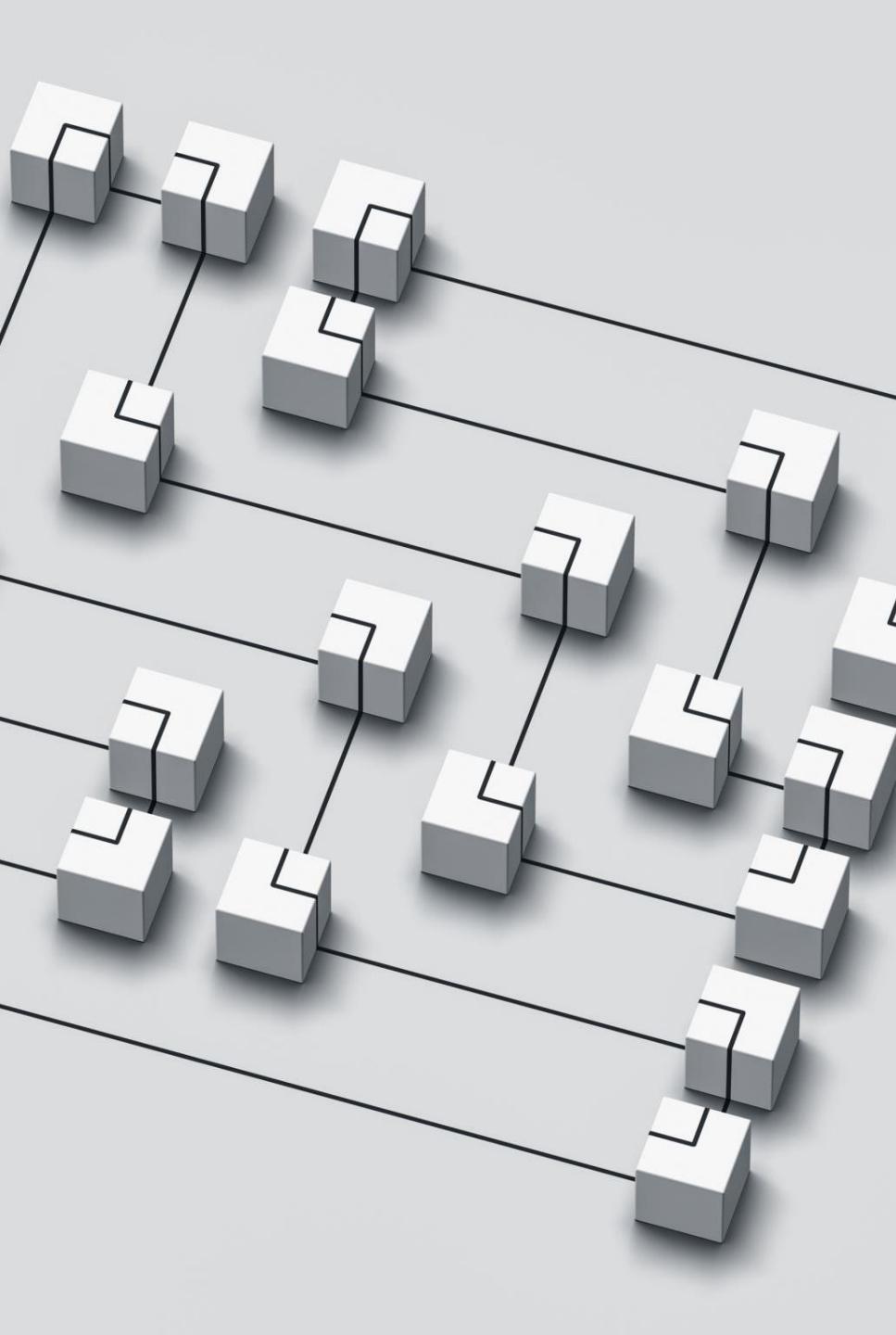
Private subnet



Web application instances



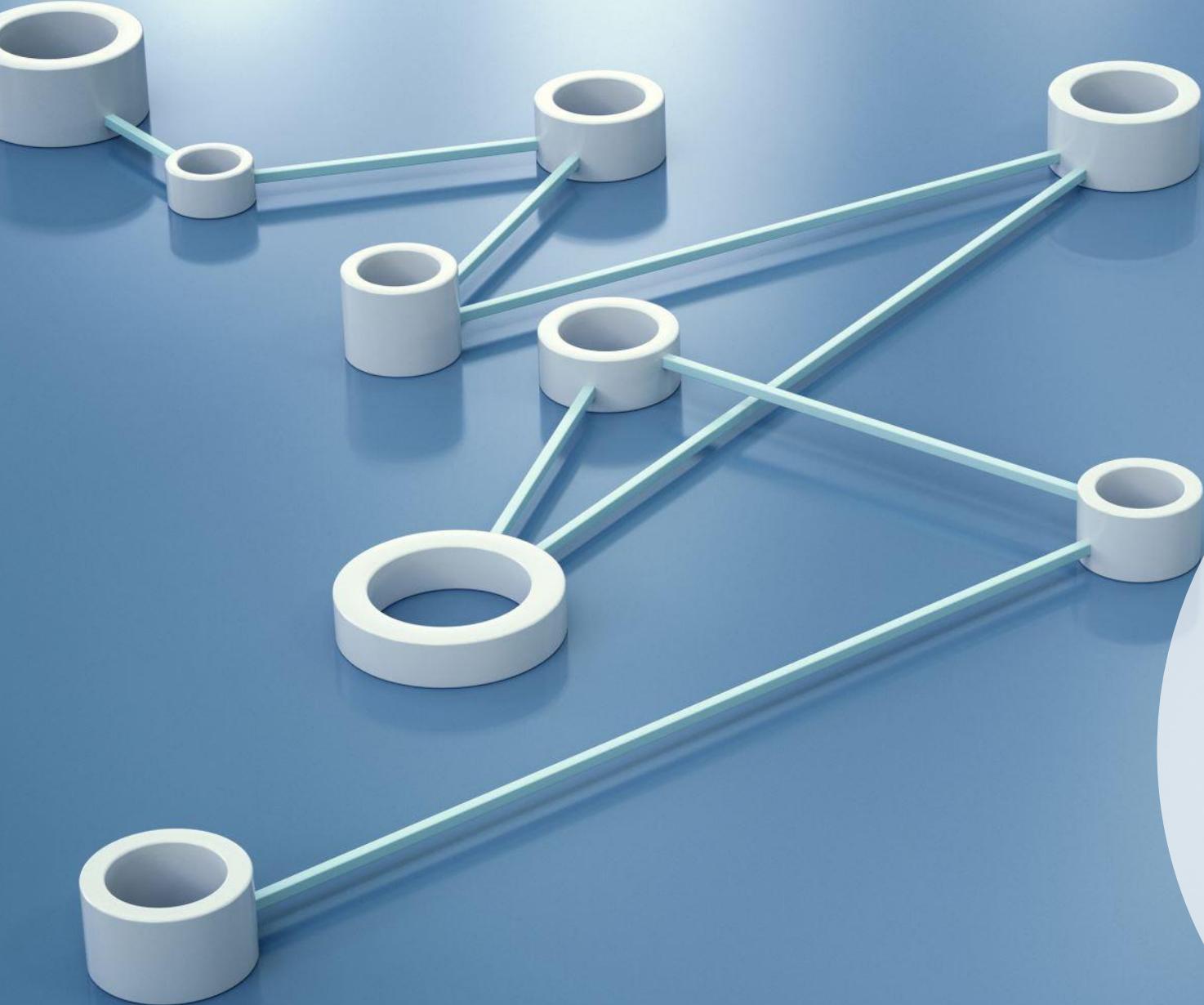
Public or private subnet



# key takeaways

---

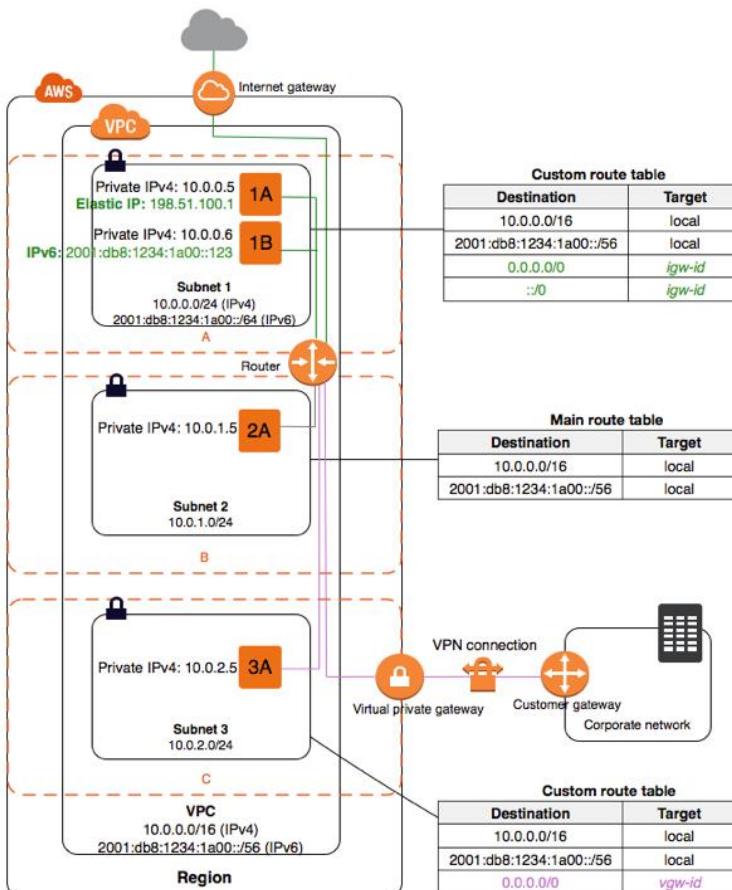
- An internet gateway allows communication between instances in your VPC and the internet.
- Route tables control traffic from your subnet or gateway.
- Elastic IP addresses are static, public IPv4 addresses that can be associated with an instance or elastic network interface. They can be remapped to another instance in your account.
- NAT devices (instance/gateway) enable instances in the private subnet to initiate **outbound traffic** to the internet or other AWS services.



## Routers and Routing Tables

---

# VPC Routers & Route Tables



A VPC comes with an implicit router that is not visible.

**Route table** is a table in a **router**

- A route table contains all the information necessary to forward a packet along the best path toward its destination (rules that determine where traffic is directed).
- **VPC route table (main route table)** is created automatically.

When a subnet is created, it is automatically associated to the VPC route table (main route table).

Can create a custom route table and associate it to subnets.

**Each subnet must be associated with a route table**, which controls the routing for the subnet (subnet route table).

- User can explicitly associate a subnet with a particular custom route table.
- Otherwise, the subnet is implicitly associated with the main route table.
- A subnet can **only be associated with one route table at a time**.
- A **subnet route table can be associated with multiple subnets**.
- If a subnet route table contains an **Internet Gateway**, the subnet has internet access.

Routes for IPv4 and IPv6 are treated separately

# Route Table Concepts

The following are the key concepts for route tables.

- **Main route table**—The route table that automatically comes with your VPC. It controls the routing for all subnets that are not explicitly associated with any other route table.
- **Custom route table**—A route table that you create for your VPC.
- **Edge association** - A route table that you use to route inbound VPC traffic to an appliance. You associate a route table with the internet gateway or virtual private gateway, and specify the network interface of your appliance as the target for VPC traffic.
- **Route table association**—The association between a route table and a subnet, internet gateway, or virtual private gateway.
- **Subnet route table**—A route table that's associated with a subnet.
- **Gateway route table**—A route table that's associated with an internet gateway or virtual private gateway.
- **Local gateway route table**—A route table that's associated with an Outposts local gateway. For information about local gateways, see [Local Gateways](#) in the [AWS Outposts User Guide](#).
- **Destination**—The destination CIDR where you want traffic to go. For example, an external corporate network with a `172.16.0.0/12` CIDR.
- **Target**—The target through which to send the destination traffic; for example, an internet gateway.
- **Local route**—A default route for communication within the VPC.

[https://docs.aws.amazon.com/vpc/latest/userguide/VPC\\_Route\\_Tables.html](https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Route_Tables.html)

# Routing Table

Destination	Target
10.0.0.0/16	Local
2001:db8:1234:1a00::/56	Local
172.31.0.0/16	pcx-11223344556677889
0.0.0.0/0	igw-12345678901234567
::/0	eigw-aabbccdde1122334

Route table have 2 columns: **Destination** and **Target**

- Destination – specifies IP range that can be directed to the target
- Target – where traffic is directed to
  - Local – only local traffic can flow
- AWS uses **the most specific route** that matches either IPv4 traffic or IPv6 traffic to determine how to route the traffic.
- Ex:
  - **The targets of the first two entries are local**
  - The third entry is the routing option for VPC peering (to another VPC with CIDR block 172.31.0.0/16)
  - **The fourth goes to internet gateway** (allowing all traffic to the internet)
  - The fifth is for IPv6 egress-only to the internet gateway



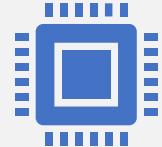
# Network Security

# Network Security



Security Groups

Firewall  
**Per Instance**

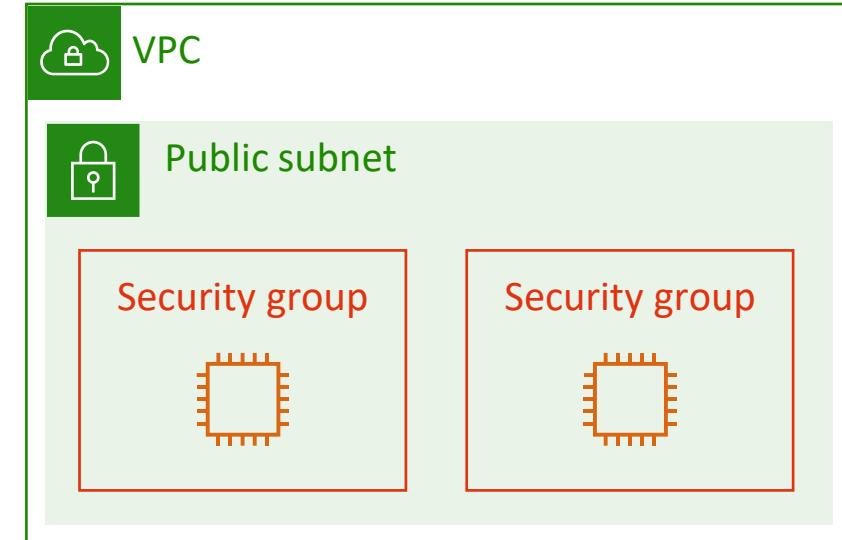


NACL

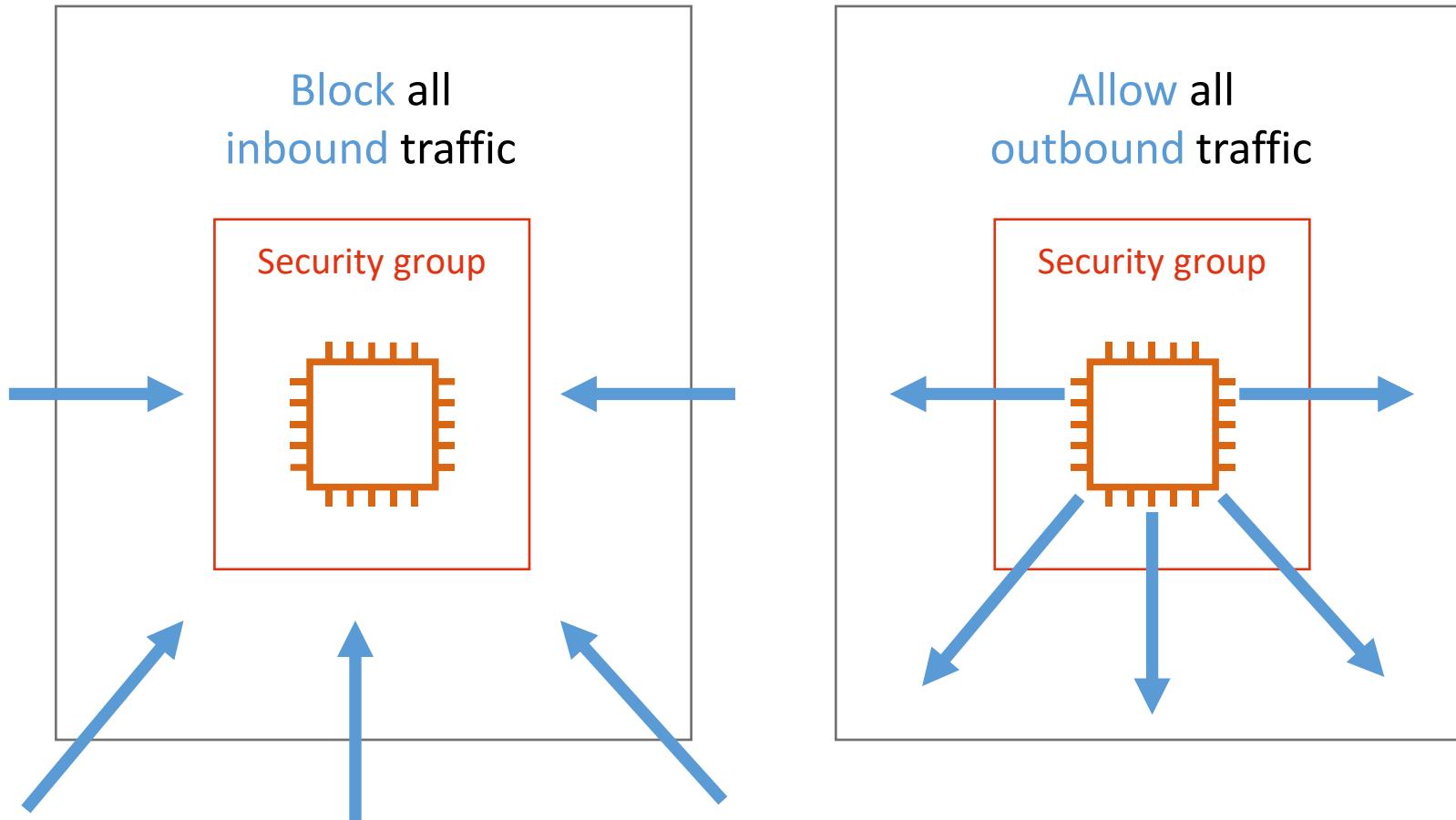
Network  
Access Control  
List  
**Subnet level**

# Security groups

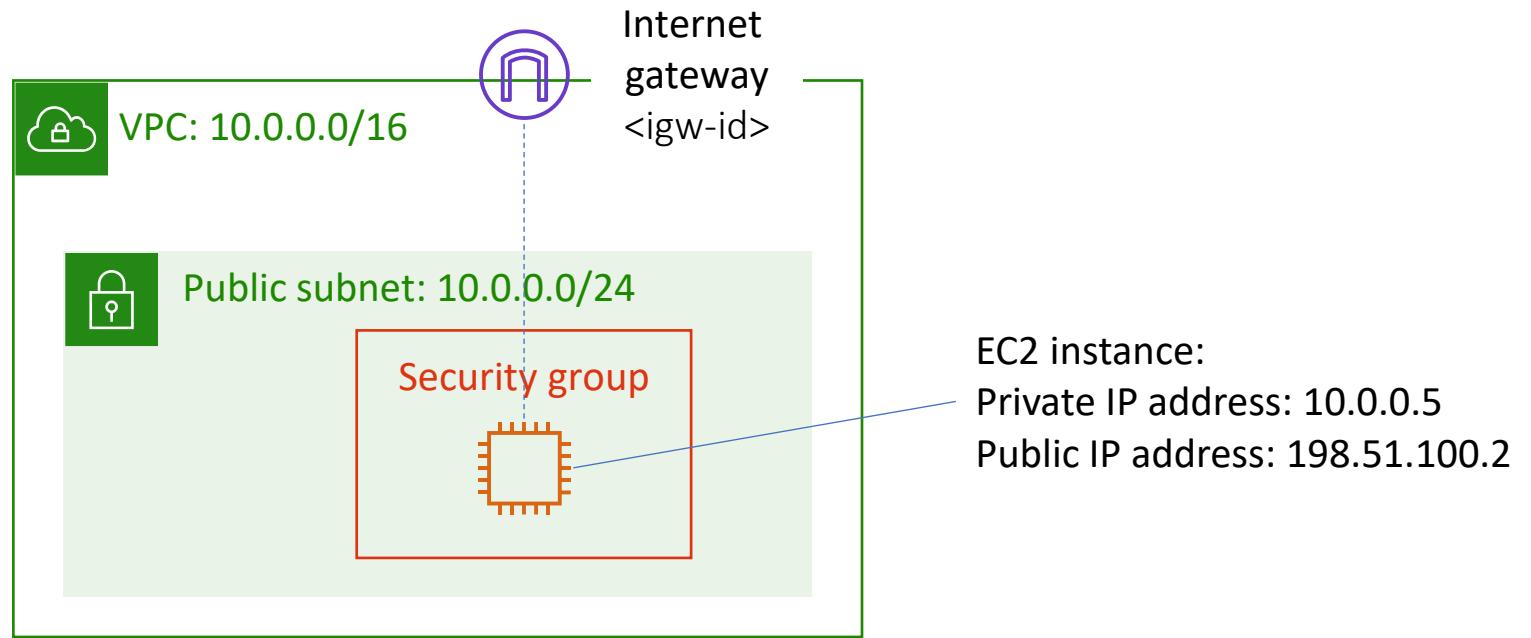
- Are **stateful firewalls** that control inbound and outbound traffic to AWS resources
- Act at the **level of the instance or network interface**



# Default for new custom security groups



# Custom security groups



Inbound				
Type	Protocol	Port Range	Source	Destination
HTTP	TCP	80	Anywhere	Allow web access

# Chaining security groups

Inbound rules

Allow: HTTP (port 80) or HTTPS (port 443)

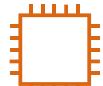
Source: 0.0.0.0/0 (any)

Allow: SSH (port 22) to Web tier

Source: Corporate IP range



Web tier security group



Inbound rule  
Allow: port 8000  
(application specific)  
Source: Web tier

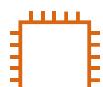
Inbound rule

Allow: SSH (port 22) to Application tier

Source: Corporate IP range



Application tier security group



Inbound rule  
Allow: TCP port 3306  
Source: Application tier

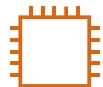
Inbound rule

Allow: SSH (port 22) to Database tier

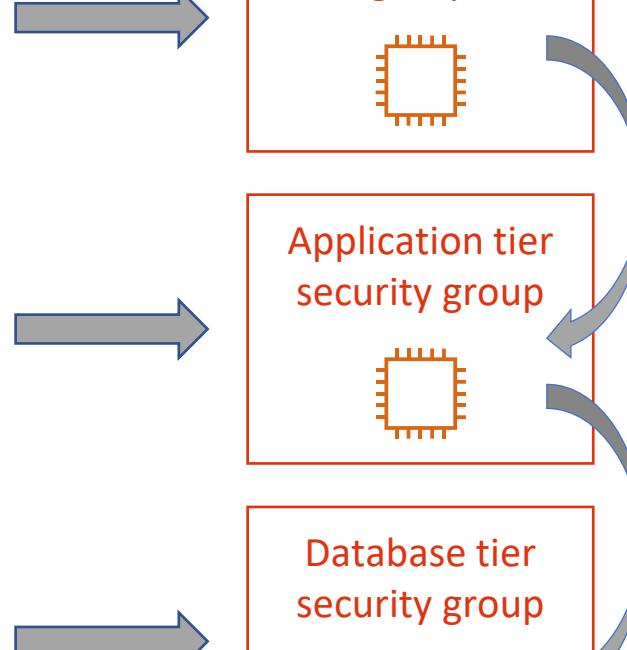
Source: Corporate IP range



Database tier security group



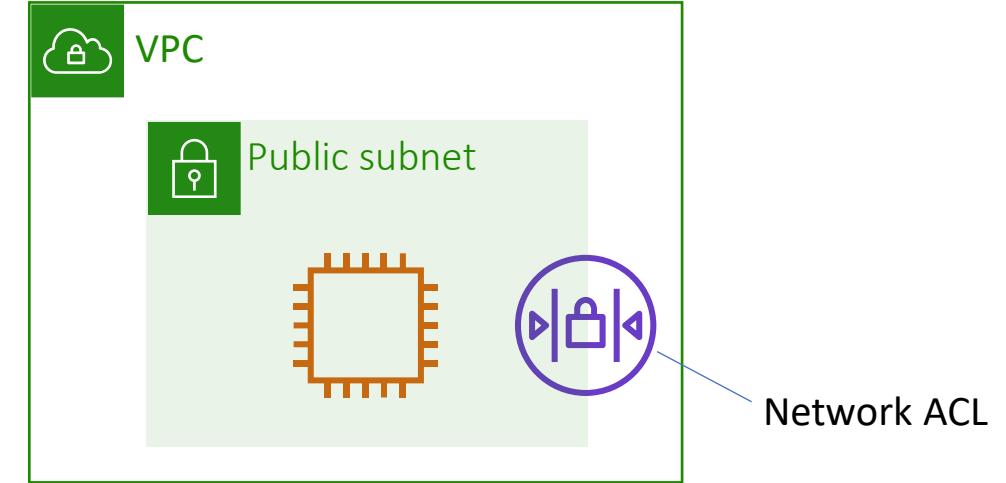
All other ports  
blocked by default



# Network access control lists (network ACLs)



- Act at the subnet level
- Allow all inbound and outbound traffic by default
- Are stateless firewalls that require explicit rules for both inbound and outbound traffic

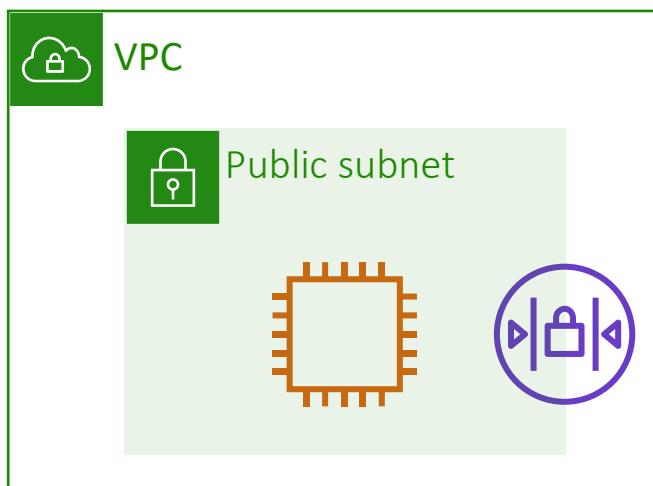


# Network Access Control List

- Network Access Control List (**NACL**) is an **optional** layer of security, a firewall at the **subnet level**.
- Amazon VPC comes with a default NACL that can be modified.
  - Allows all inbound and outbound IPv4 and IPv6 traffic (if applicable).
- Custom NACL can be created and associated with a subnet.
- can associate a network ACL with multiple subnets. However, a subnet can be associated with **only one** network ACL at a time
- **A subnet can only be associated with one NACL at a time.**
  - When an NACL is associated with a subnet, the previous association is removed.
- NACLs contain a numbered list of rules that are evaluated in order to decide whether the traffic is allowed to a particular subnet associated with the NACL.
  - Rules are evaluated **starting with the lowest numbered rule**.
  - The highest number that can be used for a rule is 32766
  - It is recommended that rules are created with numbers that are multiples of 100.
- NACLs are stateless
- NACLS has inbound and outbound rules
- NACL also supports protocols (can **allow or deny** traffic)
  - Remember, Security Groups can only “allow”

# Custom network ACLs

Recommended for  
specific network security requirements only



Nacl-11223344

Inbound:

Rules # 100: SSH 172.31.1.2/32 ALLOW  
Rules # \*: ALL traffic 0.0.0.0/0 DENY

Outbound:

Rules # 100: Custom TCP 172.31.1.2/31 ALLOW  
Rules # \*: All traffic 0.0.0.0/0 DENY

# Default Network ACL

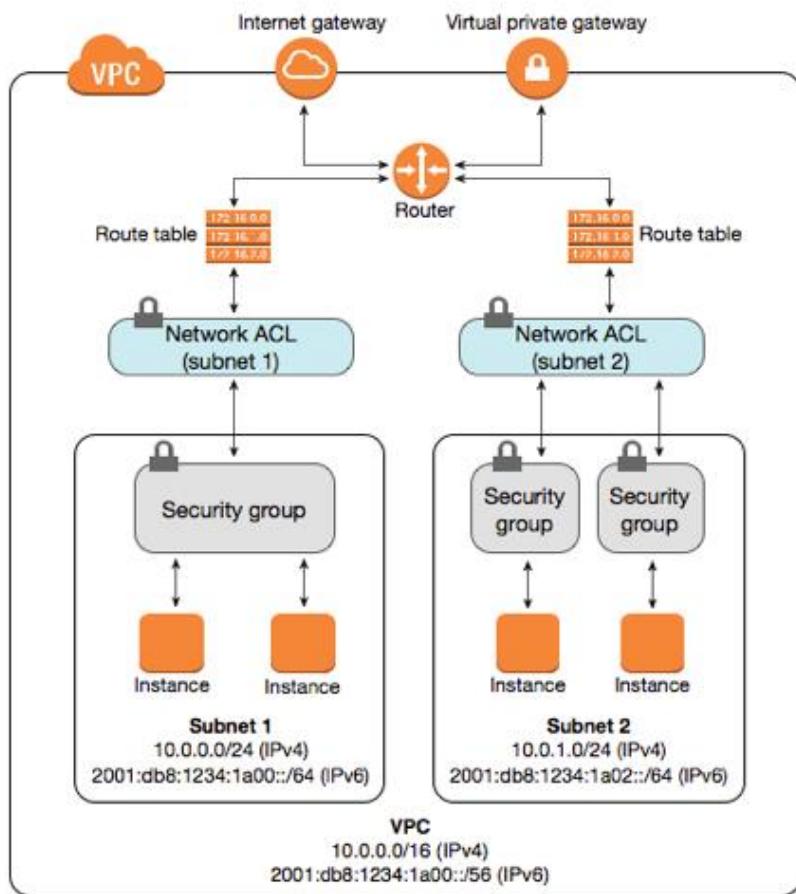
The default network ACL is configured to allow all traffic to flow in and out of the subnets with which it is associated.

Each network ACL also includes a rule which rule number is an asterisk. This rule ensures that if a packet doesn't match any of the other numbered rules, it's denied. You can't modify or remove this rule.

<https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html>

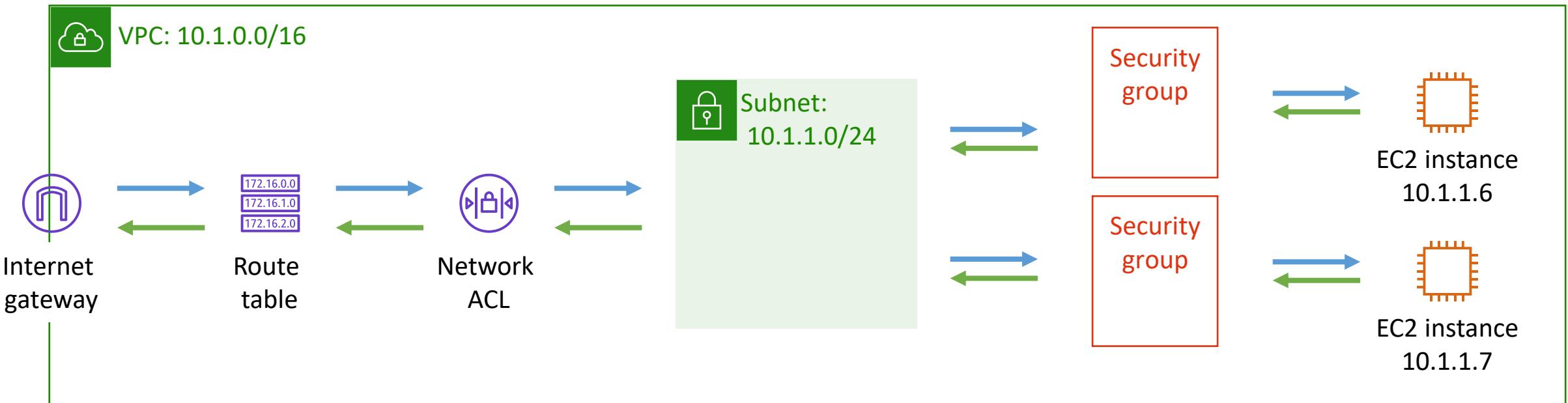
Inbound					
Rule #	Type	Protocol	Port Range	Source	Allow/Deny
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0	DENY
Outbound					
Rule #	Type	Protocol	Port Range	Destination	Allow/Deny
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0	DENY

# Security Group vs. NACL



NACL		
Security Group	Security Group	Security Group
Security Group	NACL	
Applied only at <b>instance level</b>	<b>Subnet level</b>	
Stateful (return traffic is allowed by default)	Stateless (return traffic is not allowed by default)	
<b>Allow</b> rules only; can't specify a deny rule explicitly	<b>Allow and Deny</b> rules	
All rules are evaluated before allowing traffic	The rule number gets precedence (starting lowest numbered rule)	

# Structure your infrastructure with multiple layers of defense



# Network Security key takeaways

- Security groups are stateful firewalls that act at the instance level
- Network ACLs are stateless firewalls that act at the subnet level
- When you set inbound and outbound rules to allow traffic to flow from the top tier to the bottom tier of your architecture, you can chain security groups together to isolate a security breach
- You should structure your infrastructure with multiple layers of defense

# Review: How to create a public subnet

To create a [public subnet](#) to allow communication between instances in your VPC and the internet, you must:



Attach an [internet gateway](#) to your VPC.

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Point your instance subnet's [route table](#) to the internet gateway.



Make sure that your instances have [public IP](#) or [Elastic IP](#) addresses.



[Security group](#)



Make sure that your [security groups](#) and [network ACLs](#) allow relevant traffic to flow.

Connecting your VPC to  
supported AWS services  
(VPC Endpoints)



# VPC endpoints

- Enable you to privately connect your VPC to supported AWS services and to VPC endpoint services that are powered by AWS PrivateLink
- Enable traffic between your VPC and the other service without leaving the Amazon network
- Do not require an internet gateway, VPN, network address translation (NAT) devices, or firewall proxies
- Are horizontally scaled, redundant, and highly available
- Endpoints are supported **within the same Region only**

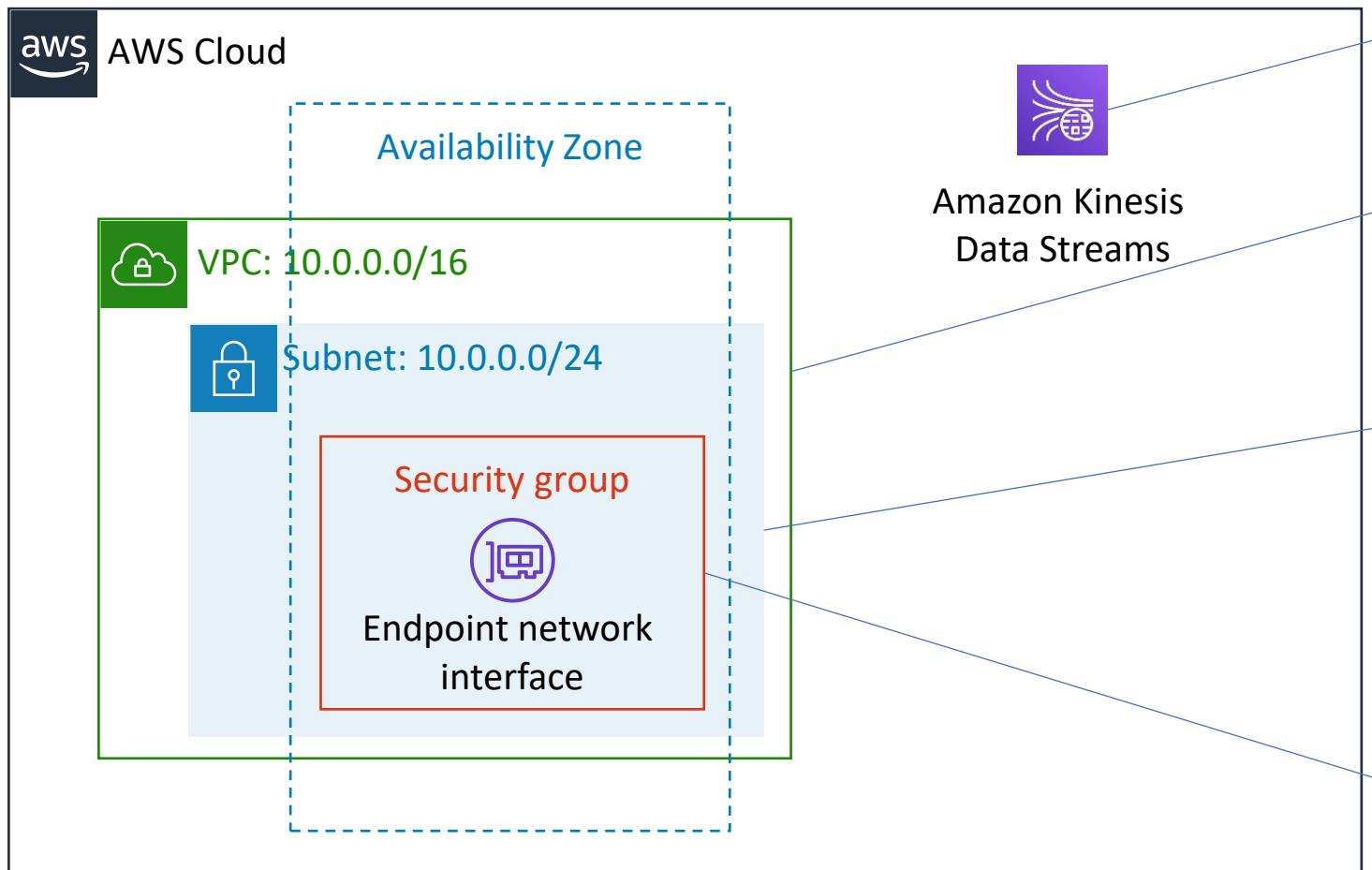


# Two types of VPC endpoints



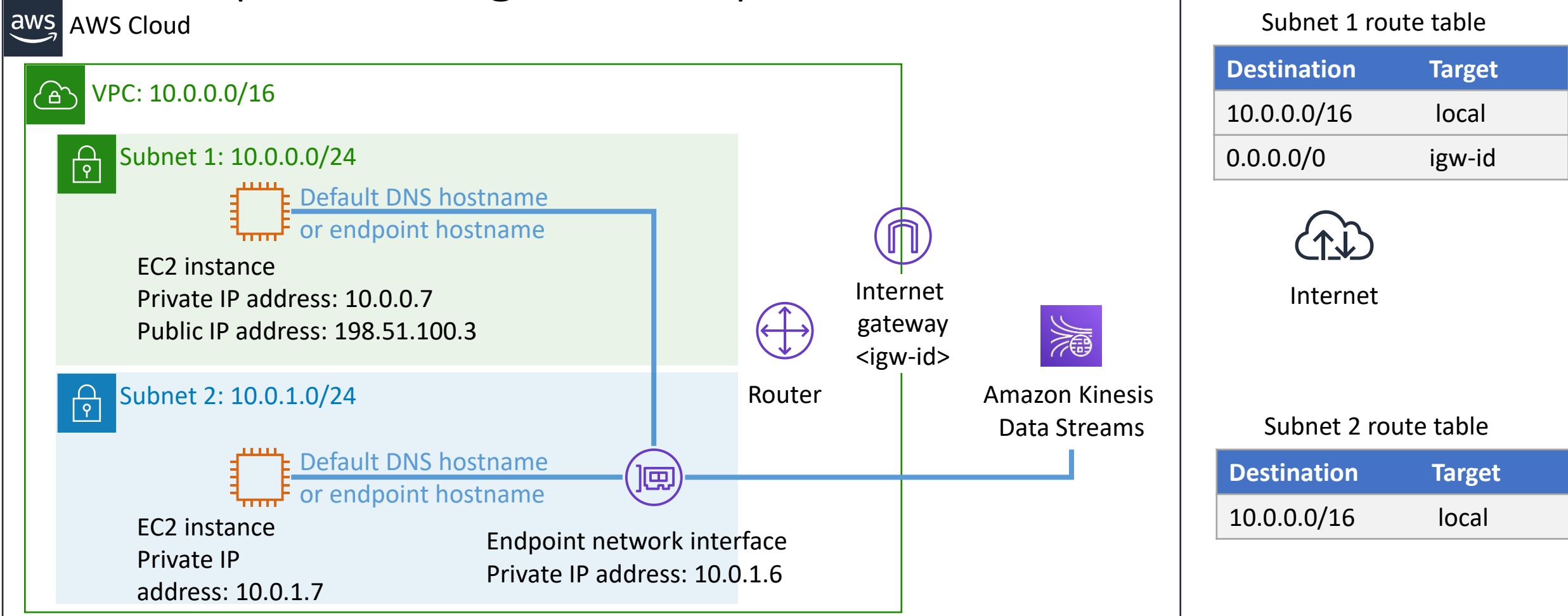
- **Interface endpoint** – An elastic network interface with a private IP address that serves as an entry point for traffic destined to a supported service
- Powered by [AWS PrivateLink](#)
- Examples –
  - Amazon CloudWatch
  - Amazon EC2 API
  - Elastic Load Balancing
- **Gateway endpoint** – A gateway that you specify as a target for a route in your route table for traffic destined to a supported AWS service
- Supported AWS services –
  - Amazon S3
  - Amazon DynamoDB

# How to set up an interface endpoint



1. Specify the AWS service, endpoint service, or AWS Marketplace service you want to connect to.
2. Choose the VPC where you want to create the interface endpoint.
3. Choose a subnet in your VPC that will use the interface endpoint. You can specify more than one subnet in different Availability Zones (as supported by the service).
4. (Optional) Enable private Domain Name System (DNS) for the endpoint.
5. Specify the security groups to associate with the network interface.

# Example of using VPC endpoints



Default DNS hostname: kinesis.us-east-1.amazonaws.com

Endpoint-specific DNS hostname: vpce-123-ab-kinesis.us-east-1.vpce.amazonaws.com

# VPC Flow Logs

- VPC Flow Logs – capture information about the IP traffic going to and from **network interfaces** in a VPC.
- Flow log data can be published to: CloudWatch Logs, S3, or Kinesis Data Firehose.
- Information can be used for:
  - **Trouble shooting** why traffic is not reaching instances
  - Can be as **security tool** to **monitor traffic** to instances
- Flow logs can be created for network interfaces, subnets, and VPCs
- No additional charges other than standard Cloud Watch charges.

# Flow log Syntax

Flow log syntax:

```
<version> <account-id> <interface-id> <srcaddr> <dstaddr> <srcport> <dstport> <protocol> <packets> <bytes> <start> <end>  
<action> <log-status>
```

Protocol = The IANA protocol number of the traffic. ICMP=1, TCP=6

Action = Accept/Reject

<https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html#flow-logs-default>

# Some Protocol Numbers

Decimal ☒	Keyword ☒	Protocol ☒	IPv6 Extension Header ☒	Reference ☒
0	HOPOPT	IPv6 Hop-by-Hop Option	Y	[RFC8200]
1	ICMP	Internet Control Message		[RFC792]
2	IGMP	Internet Group Management		[RFC1112]
3	GGP	Gateway-to-Gateway		[RFC823]
4	IPv4	IPv4 encapsulation		[RFC2003]
5	ST	Stream		[RFC1190][RFC1819]
6	TCP	Transmission Control		[RFC9293]
7	CBT	CBT		[Tony Ballardie]
8	EGP	Exterior Gateway Protocol		[RFC888][David Mills]
9	IGP	any private interior gateway (used by Cisco for their IGRP)		[Internet Assigned Numbers Authority]
10	BBN-RCC-MON	BBN RCC Monitoring		[Steve Chipman]
11	NVP-II	Network Voice Protocol		[RFC741][Steve Casner]
12	PUP	PUP		[Boggs, D., J. Shoch, E. Taft, and R. Metcalfe, "A PUP Protocol Specification", RFC 79-10, July 1979; also in IEEE Transactions on Communications, Vol. COM-28, No. 5, May 1980.]

<https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

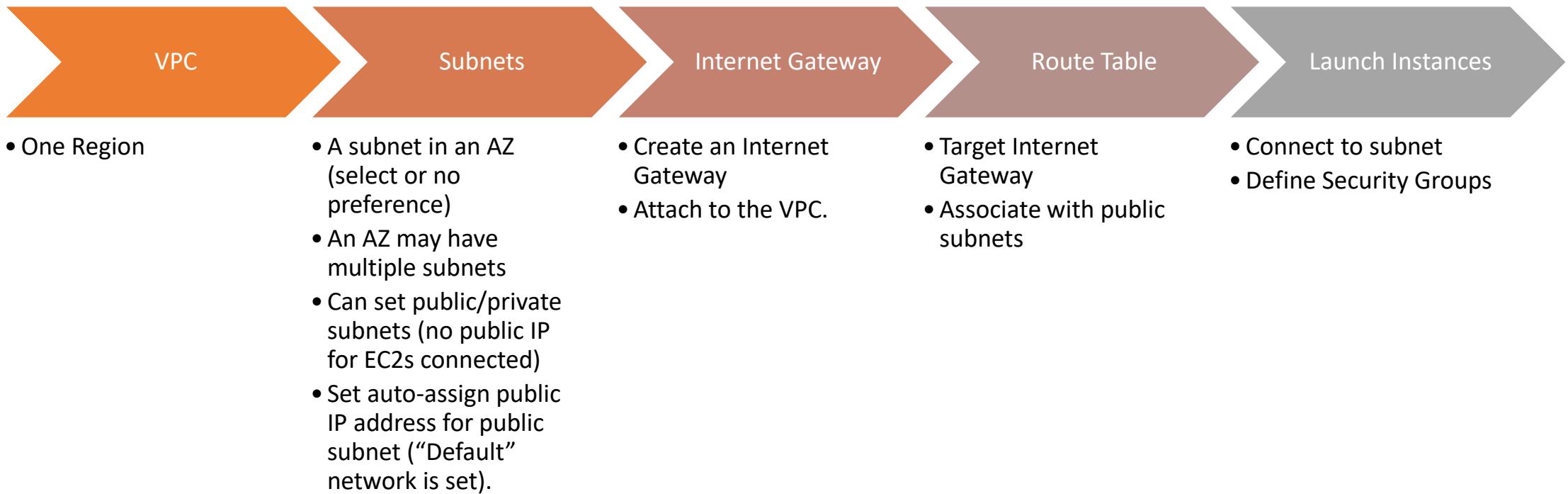
# Demos

---

- Demo 1: Create customized VPC
- Demo 2: NACL (stateless)
- Demo 3: VPC Endpoint – Gateway Endpoint – S3
- Demo 4: Public & Private Subnets + NAT Gateway
- Demo 5: VPC Flow logs

# Demo1: Create Customized VPC

# Steps to Create Customized VPC



# AWS VPC Quotas

## VPC and subnets

Resource	Default	Comments
VPCs per Region	5	<p>The quota for internet gateways per Region is directly correlated to this one. Increasing this quota increases the quota on internet gateways per Region by the same amount.</p> <p>You can have 100s of VPCs per Region for your needs even though the default quota is 5 VPCs per Region.</p>
Subnets per VPC	200	-

[https://docs.amazonaws.cn/en\\_us/vpc/latest/userguide/amazon-vpc-limits.html](https://docs.amazonaws.cn/en_us/vpc/latest/userguide/amazon-vpc-limits.html)

# VPC Demo

- VPC Name: santoso-vpc
- CIDR: 172.16.0.0/16
- Subnet 1: santoso-subnet-public
- Subnet 1 CIDR: 172.16.1.0/24

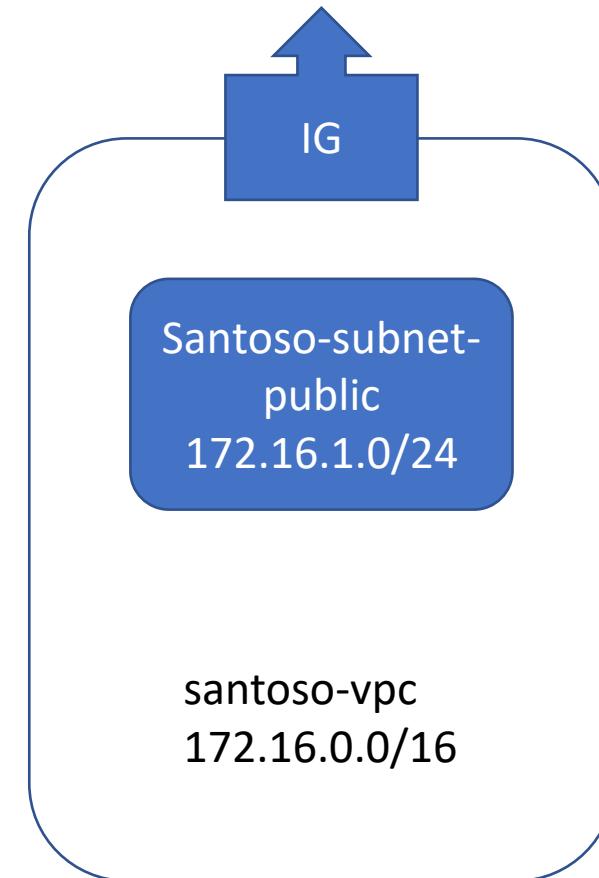
AWS allows block sizes between **/16** and **/28**

RFC 1918 private IP addresses (IANA)

10.0.0.0 - 10.255.255.255 (10/8 prefix)

**172.16.0.0 - 172.31.255.255 (172.16/12 prefix)**

192.168.0.0 - 192.168.255.255 (192.168/16 prefix)



After you are done:

1. Terminate EC2s
2. Delete VPC

# We need to create a VPC

The image consists of three vertically stacked screenshots of the AWS VPC service dashboard, illustrating the steps to create a new VPC.

**Screenshot 1: Initial Dashboard**

This screenshot shows the AWS Services navigation bar with "Services" selected. Below it, the "VPC" service is highlighted in the "Isolated Cloud Resources" section. A red arrow points from the top-left towards the "Services" button. Another red arrow points from the top-right towards the "N. Virginia" region selector.

**Screenshot 2: VPC Dashboard**

This screenshot shows the "VPC Dashboard" page. It features a "Launch VPC Wizard" button and a "Launch EC2 Instances" button. The "Resources by Region" section shows "VPCs" (4) in the N. Virginia region. A red arrow points from the bottom-left towards the "VPCs" link. Another red arrow points from the bottom-right towards the "Create VPC" button.

**Screenshot 3: Your VPCs List**

This screenshot shows the "Your VPCs" list page. It displays four existing VPCs with columns for Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR (Network Border Group), IPv6 pool, DHCP options set, Main route table, and Main network ACL. A red arrow points from the bottom-right towards the "Create VPC" button.

aws Services Resource Groups

VPC > Your VPCs > Create VPC

### Create VPC Info

A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances.

**VPC settings**

Name tag - *optional*  
Creates a tag with a key of 'Name' and a value that you specify.  
 ←

IPv4 CIDR block Info  
 ←

IPv6 CIDR block Info  
 No IPv6 CIDR block  
 Amazon-provided IPv6 CIDR block  
 IPv6 CIDR owned by me

Tenancy Info

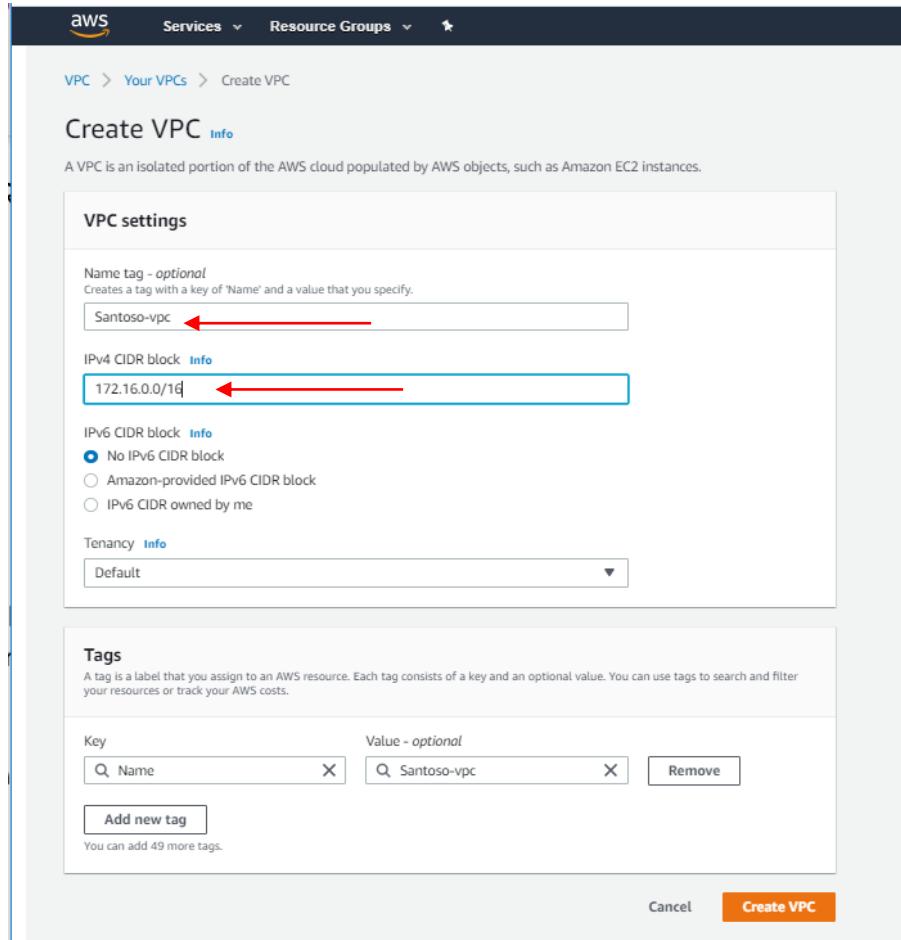
**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - <i>optional</i>
<input type="text" value="Name"/> <span style="color: red;">X</span>	<input type="text" value="Santoso-vpc"/> <span style="color: red;">X</span>

←

You can add 49 more tags.

Cancel Create VPC



ⓘ New VPC Experience  
Tell us what you think

ⓘ You successfully created vpc-02da94f22b37d9922 / Santoso-vpc

VPC Dashboard New

Filter by VPC:  
Select a VPC

VIRTUAL PRIVATE CLOUD

- Your VPCs New
- Subnets
- Route Tables
- Internet Gateways New
- Egress Only Internet Gateways New
- Carrier Gateways New
- DHCP Options Sets New
- Elastic IPs New
- Managed Prefix Lists New
- Endpoints
- Endpoint Services
- NAT Gateways New
- Peering Connections

SECURITY

- Network ACLs
- Security Groups New

VIRTUAL PRIVATE NETWORK (VPN)

VPC > Your VPCs > vpc-02da94f22b37d9922

vpc-02da94f22b37d9922 / Santoso-vpc

Actions ▾

Details		Info	
VPC ID	vpc-02da94f22b37d9922	State	Available
Tenancy	Default	DHCP options set	dopt-ebbe3d91
Default VPC	No	IPv4 CIDR	172.16.0.0/16
Owner ID	183451715204	IPv6 pool	-
DNS hostnames	Disabled	DNS resolution	Enabled
Route table	rtb-03aad3ef9b0442140	Network ACL	acl-03ed4a09379efdfb2
IPv6 CIDR (Network Border Group)	-	IPv6 CIDR (Network Border Group)	-

CIDRs | Flow logs | Tags

IPv4 CIDRs		Info	
CIDR	Status		
172.16.0.0/16	Associated		

IPv6 CIDRs			Info	
CIDR	Pool	Status		
You have no IPv6 CIDR blocks associated with your VPC.				

⌚ New VPC Experience  
Tell us what you think

VPC Dashboard **New**

Filter by VPC:  
Select a VPC

**VIRTUAL PRIVATE CLOUD**

- Your VPCs** **New**
- Subnets
- Route Tables
- Internet Gateways **New**
- Egress Only Internet Gateways **New**
- Carrier Gateways **New**
- DHCP Options Sets **New**
- Elastic IPs **New**
- Managed Prefix Lists **New**
- Endpoints
- Endpoint Services
- NAT Gateways **New**
- Peering Connections

**SECURITY**

- Network ACLs
- Security Groups **New**

**VIRTUAL PRIVATE NETWORK (VPN)**

You successfully created vpc-02da94f22b37d9922 / Santoso-vpc

Your VPCs (1/2) **Info**

Filter VPCs

**Create VPC**

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR (Network Border Group)	IPv6 pool	DHCP options set	Main route table	Main network ACL
Santoso-vpc	vpc-02da94f22b37d9922	Available	172.16.0.0/16	-	-	dopt-ebbe3d91	rtb-03aad3ef9b0442140	acl-03ed4a09379efdfb2
-	vpc-472bcd3a	Available	172.31.0.0/16	-	-	dopt-ebbe3d91	rtb-27abc659	acl-2291d85f

vpc-02da94f22b37d9922 / Santoso-vpc

**Details** | CIDRs | Flow logs | Tags

**Details**

VPC ID vpc-02da94f22b37d9922	State Available	DNS hostnames Disabled	DNS resolution Enabled
Tenancy Default	DHCP options set dopt-ebbe3d91	Route table rtb-03aad3ef9b0442140	Network ACL acl-03ed4a09379efdfb2
Default VPC No	IPv4 CIDR 172.16.0.0/16	IPv6 pool -	IPv6 CIDR (Network Border Group) -
Owner ID 183451715204			

# Create a Subnet (one at a time)

The screenshot shows the AWS VPC Dashboard and a 'Create subnet' wizard.

**VPC Dashboard:**

- Shows a table of default subnets attached to the default VPC.
- Filters: Name, Subnet ID, State, VPC, IPv4 CIDR, Available IPv4, IPv6 CIDR, Availability Zone, Availability Zone ID, Network Border Group, Route table, Network ACL, Default subnet, Auto-assign IP.
- Actions: Create subnet, Actions dropdown.

**Create subnet Wizard:**

- Step 1: Create subnet**
  - Name tag: santoso-subnet-public
  - Give a name
  - VPC\*: vpc-02da94f22b37d9922
  - You should see the VPC you created in the drop-down menu
  - Availability Zone: us-east-1a
  - You can select an AZ within the region or AWS can select for you
  - VPC CIDRs: CIDR 172.16.0.0/16 Status: associated
  - IPv4 CIDR block\*: 172.16.1.0/24 Define the CIDR block for this subnet
- Step 2: Confirmation**
  - The following Subnet was created:  
Subnet ID: subnet-030cde6afaf7990a

# Check if subnet is created properly

The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with various VPC-related options like Subnets, Route Tables, Internet Gateways, and Security Groups. The main area has a table of subnets and a detailed view of a selected subnet.

**Subnet Table Headers:**

- Name
- Subnet ID
- State
- VPC
- IPv4 CIDR
- Available IPv4
- IPv6 CIDR
- Availability Zone
- Availability Zone ID
- Network Border Group
- Route table
- Network ACL
- Default subnet
- Auto-assign public IPv4 address

**Subnet Table Data:**

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Network Border Group	Route table	Network ACL	Default subnet	Auto-assign public IPv4 address
santoso-subnet-public	subnet-03fc6efa66af7f90a	available	vpc-02da94f22b37d9922   Santoso-vpc	172.16.1.0/24	251	-	us-east-1a	use1-az1	us-east-1	rtb-03aad3ef9b0442140	acl-03ed4a09379efdfb2	No	No
	subnet-460e1778	available	vpc-472bcd3a	172.31.48.0/20	4091	-	us-east-1e	use1-az3	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-82d101a3	available	vpc-472bcd3a	172.31.80.0/20	4091	-	us-east-1b	use1-az2	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-8c6cbb3d	available	vpc-472bcd3a	172.31.32.0/20	4091	-	us-east-1d	use1-az6	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-8f2a4fc2	available	vpc-472bcd3a	172.31.16.0/20	4091	-	us-east-1c	use1-az4	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-b50cc6d3	available	vpc-472bcd3a	172.31.0.0/20	4091	-	us-east-1a	use1-az1	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-ff2ca0f1	available	vpc-472bcd3a	172.31.64.0/20	4091	-	us-east-1f	use1-az5	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes

**Selected Subnet Details:**

Subnet: subnet-03fc6efa66af7f90a

Description: (selected)

Flow Logs: Route Table: Network ACL: Tags: Sharing:

Subnet ID: subnet-03fc6efa66af7f90a	State: available
VPC: vpc-02da94f22b37d9922   Santoso-vpc	IPv4 CIDR: 172.16.1.0/24
Available IPv4 Addresses: 251	IPv6 CIDR: -
Availability Zone: us-east-1a (use1-az1)	Network Border Group: us-east-1
Route Table: rtb-03aad3ef9b0442140	Network ACL: acl-03ed4a09379efdfb2
Default subnet: No	Auto-assign public IPv4 address: No
Auto-assign customer-owned IPv4 address: No	Customer-owned IPv4 pool: -
Auto-assign IPv6 address: No	Outpost ID: -
Owner: 183451715204	

# Subnet -Enable auto-assign IP settings (Assign a Public IP for all EC2s spun up)

The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with 'New VPC Experience' and 'VPC Dashboard New'. Below that, it says 'Filter by VPC' and 'Select a VPC'. Under 'VIRTUAL PRIVATE CLOUD', it lists 'Your VPCs New'. The main area has a table with columns: Name, State, VPC, IPv4 CIDR, Available IPv4, IPv6 CIDR, Availability Zone, Availability Zone ID, Network Border Group, Route table, Network ACL, Default subnet, and Auto-assign. A context menu is open over the first row ('santoso-subnet'), showing options: Delete subnet, Create flow log, Modify auto-assign IP settings (which is highlighted in orange), Edit IPv6 CIDRs, Edit network ACL association, Edit route table association, Share subnet, and Add/Edit Tags.

Name	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Network Border Group	Route table	Network ACL	Default subnet	Auto-assign
santoso-subnet	available	vpc-02da94f22b37d9922	172.16.1.0/24	251	-	us-east-1a	use1-az1	us-east-1	rtb-03aad3ef9b0442140	acl-03ed4a09379efdb2	No	No
	available	vpc-472bcd3a	172.31.48.0/20	4091	-	us-east-1e	use1-az3	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	available	vpc-472bcd3a	172.31.80.0/20	4091	-	us-east-1b	use1-az2	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	available	vpc-472bcd3a	172.31.32.0/20	4091	-	us-east-1d	use1-az6	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes

**Subnets > Modify auto-assign IP settings**

## Modify auto-assign IP settings

Enable the auto-assign IP address setting to automatically request a public IPv4 or IPv6 address for an instance launched in this subnet. You can override the auto-assign IP settings for an instance at launch time.

Subnet ID: subnet-03fc6efa66af7f90a

Auto-assign IPv4  Enable auto-assign public IPv4 address [?](#)

Auto-assign Co-IP  Enable auto-assign customer-owned IPv4 address [?](#)

\* Required

[Cancel](#) [Save](#)

# Check

Screenshot of the AWS VPC Dashboard showing the subnet configuration for the 'santoso-subnet-public' VPC.

**Create subnet** Actions ▾

Filter by tags and attributes or search by keyword

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Network Border Group	Route table	Network ACL	Default subnet	Auto-assign public IPv4 address
santoso-subnet-public	subnet-03fc6efa66af7f90a	available	vpc-02da94f22b37d9922   Santoso-vpc	172.16.1.0/24	251	-	us-east-1a	use1-az1	us-east-1	rtb-03aad3ef9b0442140	acl-03ed4a09379efdfb2	No	Yes
	subnet-460e1778	available	vpc-472bcd3a	172.31.48.0/20	4091	-	us-east-1e	use1-az3	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-82d101a3	available	vpc-472bcd3a	172.31.80.0/20	4091	-	us-east-1b	use1-az2	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-8c6cbbd3	available	vpc-472bcd3a	172.31.32.0/20	4091	-	us-east-1d	use1-az6	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-8f2a4fc2	available	vpc-472bcd3a	172.31.16.0/20	4091	-	us-east-1c	use1-az4	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-b50cc6d3	available	vpc-472bcd3a	172.31.0.0/20	4091	-	us-east-1a	use1-az1	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes
	subnet-f12ca0f1	available	vpc-472bcd3a	172.31.64.0/20	4091	-	us-east-1f	use1-az5	us-east-1	rtb-27abc659	acl-2291d85f	Yes	Yes

Subnet: subnet-03fc6efa66af7f90a

Description Flow Logs Route Table Network ACL Tags Sharing

Subnet ID: subnet-03fc6efa66af7f90a	State: available
VPC: vpc-02da94f22b37d9922   Santoso-vpc	IPv4 CIDR: 172.16.1.0/24
Available IPv4 Addresses: 251	IPv6 CIDR: -
Availability Zone: us-east-1a (use1-az1)	Network Border Group: us-east-1
Route Table: rtb-03aad3ef9b0442140	Network ACL: acl-03ed4a09379efdfb2
Default subnet: No	Auto-assign public IPv4 address: Yes
Auto-assign customer-owned IPv4 address: No	Customer-owned IPv4 pool: -
Auto-assign IPv6 address: No	Outpost ID: -
Owner: 183451715204	

# Create Internet Gateway

The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with 'New VPC Experience' and 'VPC Dashboard New'. Below that is a 'Filter by VPC:' dropdown and a 'Select a VPC' button. Under 'VIRTUAL PRIVATE CLOUD', it lists 'Your VPCs New', 'Subnets', 'Route Tables', and 'Internet Gateways New'. The main area is titled 'Internet gateways (1/1)'. It has a search bar 'Filter internet gateways'. A table header includes columns for 'Name', 'Internet gateway ID', 'State', 'VPC ID', and 'Owner'. One row is shown: 'igw-bd7a2fc6' (Attached), 'vpc-472bcd3a', and '183451715204'. To the right of the table is a 'Create internet gateway' button. A red arrow points from the text 'We will create a new IG for our customized VPC' down to the 'Create internet gateway' button.

You will see an existing internet gateway. AWS automatically creates an IG for the default VPC. Don't remove this. We will create a new IG for our customized VPC

The screenshot shows the 'Create internet gateway' wizard. The title is 'Create internet gateway'. It says: 'An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.' Below this is a 'Name tag' field containing 'santoso-IG'. A blue arrow points from the text 'Name it' to this field. The next section is 'Tags - optional', which contains a table with a single row: 'Key: Name' and 'Value - optional: santoso-IG'. There are 'Add new tag' and 'Remove' buttons. At the bottom are 'Cancel' and 'Create internet gateway' buttons.

Name it

# Attach IG to VPC

The following internet gateway was created: igw-0d1deacd5cbf898b7. You can now attach to a VPC to enable the VPC to communicate with the internet.

Internet gateways (1/2) [Info](#)

Name	Internet gateway ID	State	VPC ID	Owner
santoso-IG	igw-0d1deacd5cbf898b7	Detached	-	183451715204
	igw-bd7a2fc6	Attached	vpc-472bcd3a	183451715204

Actions ▾ [Create internet gateway](#)

View details

Attach to VPC

Detach from VPC

Manage tags

Delete internet gateway

VPC > Internet gateways > Attach to VPC (igw-0d1deacd5cbf898b7)

## Attach to VPC (igw-0d1deacd5cbf898b7) [Info](#)

**VPC**

Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs

Attach the internet gateway to this VPC.

AWS Command Line Interface command

Cancel [Attach Internet gateway](#)

You should see the VPC in the drop-down menu

Internet gateway igw-0d1deacd5cbf898b7 successfully attached to vpc-02da94f22b37d9922

Internet gateways (2) [Info](#)

Name	Internet gateway ID	State	VPC ID	Owner
santoso-IG	igw-0d1deacd5cbf898b7	Attached	vpc-02da94f22b37d9922   Santoso-vpc	183451715204

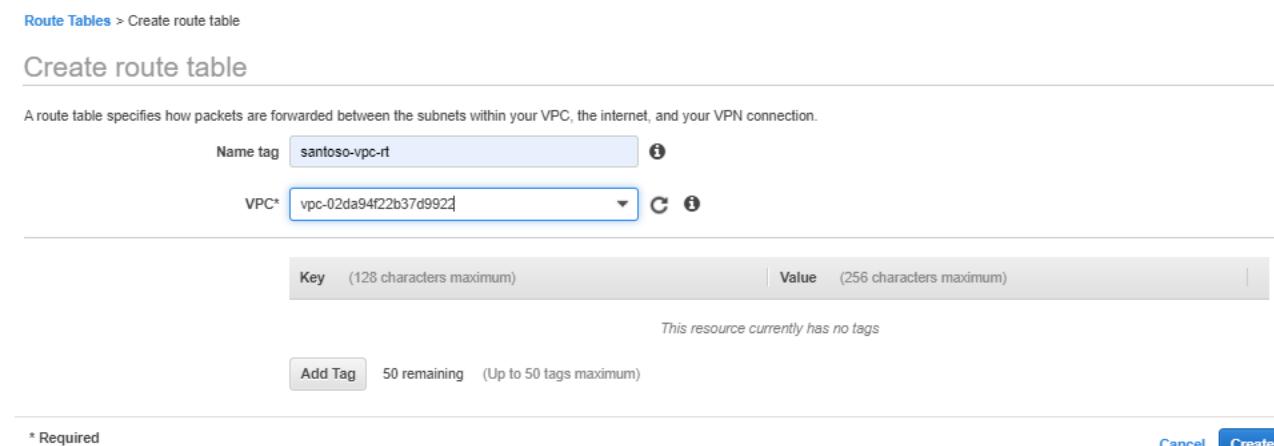
# Create a Route Table for the subnet



The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with 'Route Tables' selected. The main area displays a table of route tables. A red arrow points to the first row, which represents the 'Main Route Table'. The table columns include Name, Route Table ID, Explicit subnet association, Edge associations, Main, VPC ID, and Owner. The first row has a 'Main' value of 'Yes'.

Name	Route Table ID	Explicit subnet association	Edge associations	Main	VPC ID	Owner
	rtb-03aad3ef9b0442140	-	-	Yes	vpc-02da94f22b37d9922   Santoso-vpc	183451715204
	rtb-27abc659	-	-	Yes	vpc-472bcd3a	183451715204

Main Route Table, automatically created for the VPC, we are going to create a subnet route table to direct internet traffic to the IG



Route Tables > Create route table

## Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Name tag: santoso-vpc-rt

VPC\*: vpc-02da94f22b37d9922

Key (128 characters maximum) | Value (256 characters maximum)

This resource currently has no tags

Add Tag 50 remaining (Up to 50 tags maximum)

\* Required

Cancel Create



Route Tables > Create route table

## Create route table

The following Route Table was created:

Route Table ID: rtb-06c3963ca7efec2aa

Close

# Associate to Subnet

The screenshot shows the AWS VPC Route Tables interface. At the top, there's a navigation bar with 'Create route table' and 'Actions' dropdown. The 'Actions' dropdown is open, showing options: 'Set Main Route Table', 'Delete Route Table', and 'Edit subnet associations' (which is highlighted in orange). Below this is a table with columns: Route Table ID, Explicit subnet association, Edge associations, Main, VPC ID, and Owner. There are three rows in the table:

Route Table ID	Explicit subnet association	Edge associations	Main	VPC ID	Owner
rtb-03aad3ef9b0442140	-	-	Yes	vpc-02da94f22b37d9922   Santoso-vpc	183451715204
rtb-06c3963ca7efec2aa	-	-	No	vpc-02da94f22b37d9922   Santoso-vpc	183451715204
rtb-27abc659	-	-	Yes	vpc-472bcd3a	183451715204

On the left sidebar, under 'VIRTUAL PRIVATE CLOUD', it says 'Select a VPC' and shows 'santoso-vpc-rt' selected. Below the table, the URL is 'Route Tables > Edit subnet associations' and the page title is 'Edit subnet associations'. It shows a list of associated subnets: 'subnet-03fc6efa66af7f90a'. A modal window is open, showing a table with columns: Subnet ID, IPv4 CIDR, IPv6 CIDR, and Current Route Table. It contains one row: 'subnet-03fc6efa66af7f90a | santoso-sub...' with '172.16.1.0/24' and 'Main'.

# Edit Route to include IG

The screenshot shows the AWS VPC Dashboard with the 'Route Tables' section selected. The main table lists three route tables:

Name	Route Table ID	Explicit subnet association	Edge associations	Main	VPC ID	Owner
santoso-vpc-rt	rtb-06c3963ca7efec2aa	subnet-03fc6efa66af7f90a	-	No	vpc-02da94f22b37d9922   Santoso-vpc	183451715204
	rtb-27abc659	-	-	Yes	vpc-472bcd3a	183451715204

A red arrow points from the 'Edit routes' button in the 'Routes' tab of the detailed view for route table 'rtb-06c3963ca7efec2aa' to the 'Destination' column in the 'Routes' table, highlighting the need to add a route to the Internet Gateway.

**Route Table: rtb-06c3963ca7efec2aa**

Summary    Routes    Subnet Associations    Edge Associations    Route Propagation    Tags

Edit routes

The subnet route can direct local traffic, but not yet internet so we need to add a route to IG

View All routes

Destination	Target	Status	Propagated
172.16.0.0/16	local	active	No

Route Tables > Edit routes

## Edit routes

Destination	Target	Status	Propagated
172.16.0.0/16	local	active	No
0.0.0.0	igw-		No

**Add route**

igw-0d1deacd5cbf898b7 santoso-IG

\* Required

Cancel Save routes

Route Tables > Edit routes

## Edit routes

✓ Routes successfully edited

Close

# Check Subnet route table

New VPC Experience  
Tell us what you think

VPC Dashboard [New](#)

Filter by VPC: [Select a VPC](#)

**VIRTUAL PRIVATE CLOUD**

- Your VPCs [New](#)
- Subnets
- Route Tables**
  - Internet Gateways [New](#)
  - Egress Only Internet Gateways [New](#)
  - Carrier Gateways [New](#)
  - DHCP Options Sets [New](#)
  - Elastic IPs [New](#)
  - Managed Prefix Lists [New](#)
- Endpoints
- Endpoint Services
- NAT Gateways [New](#)
- Peering Connections

**SECURITY**

- Network ACLs
- Security Groups [New](#)

**VIRTUAL PRIVATE NETWORK (VPN)**

- Customer Gateways

Create route table Actions ▾

Filter by tags and attributes or search by keyword

Name	Route Table ID	Explicit subnet association	Edge associations	Main	VPC ID	Owner
	rtb-03aad3ef9b0442140	-	-	Yes	vpc-02da94f22b37d9922   Santoso-vpc	183451715204
<a href="#">santoso-vpc-rt</a>	rtb-06c3963ca7efec2aa	<a href="#">subnet-03fc6efa66af7f90a</a>	-	No	vpc-02da94f22b37d9922   Santoso-vpc	183451715204
	rtb-27abc659	-	-	Yes	vpc-472bod3a	183451715204

Route Table: rtb-06c3963ca7efec2aa

Summary [Routes](#) [Subnet Associations](#) [Edge Associations](#) [Route Propagation](#) [Tags](#)

Edit routes

View All routes

Destination	Target	Status	Propagated
172.16.0.0/16	local	active	No
0.0.0.0/0	<a href="#">igw-0d1deacd5cbf898b7</a>	active	No

Remember, AWS uses **the most specific route**

# Spin up EC2

Screenshot of the AWS EC2 console showing the instance creation process.

**Step 1: Choose an Amazon Machine Image (AMI)**

The user has selected the SUSE Linux Enterprise Server 15 SP2 (HVM) AMI. The "Select" button is highlighted in blue.

**Step 2: Choose an Instance Type**

The user has selected the t2.micro instance type. The "Select" button is highlighted in blue.

**Instance Types Table:**

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro	1	1	EBS only	-	Low to Moderate	Yes

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	<input type="text" value="1"/>	Launch into Auto Scaling Group												
Purchasing option	<input type="checkbox"/> Request Spot instances													
Network	vpc-02da94f22b37d9922   Santoso-vpc <input type="button" value="Create new VPC"/>													
Subnet	subnet-03fc6fa66af7f90a   santoso-subnet-public <input type="button" value="Create new subnet"/>	251 IP Addresses available												
Auto-assign Public IP	<input type="checkbox"/> Use subnet setting (Enable)													
Placement group	<input type="checkbox"/> Add instance to placement group													
Capacity Reservation	Open <input checked="" type="checkbox"/>													
Domain join directory	No directory <input type="button" value="Create new directory"/>													
IAM role	None <input type="button" value="Create new IAM role"/>													
Shutdown behavior	Stop <input checked="" type="checkbox"/>													
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior													
Enable termination protection	<input type="checkbox"/> Protect against accidental termination													
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring <small>Additional charges apply.</small>													
Tenancy	Shared - Run a shared hardware instance <input checked="" type="checkbox"/> <small>Additional charges will apply for dedicated tenancy.</small>													
Elastic Inference	<input type="checkbox"/> Add an Elastic Inference accelerator <small>Additional charges apply.</small>													
Credit specification	<input type="checkbox"/> Unlimited <small>Additional charges may apply</small>													
File systems	<input type="button" value="Add file system"/> <input type="button" value="Create new file system"/>													
<b>Network interfaces</b> <input type="button" value="Add Device"/> <table border="1"> <thead> <tr> <th>Device</th> <th>Network Interface</th> <th>Subnet</th> <th>Primary IP</th> <th>Secondary IP addresses</th> <th>IPv6 IPs</th> </tr> </thead> <tbody> <tr> <td>eth0</td> <td>New network interface <input type="button" value="▼"/></td> <td>subnet-03fc6e <input type="button" value="▼"/></td> <td>Auto-assign <input type="button" value="Add IP"/></td> <td></td> <td>Add IP</td> </tr> </tbody> </table>			Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IPv6 IPs	eth0	New network interface <input type="button" value="▼"/>	subnet-03fc6e <input type="button" value="▼"/>	Auto-assign <input type="button" value="Add IP"/>		Add IP
Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IPv6 IPs									
eth0	New network interface <input type="button" value="▼"/>	subnet-03fc6e <input type="button" value="▼"/>	Auto-assign <input type="button" value="Add IP"/>		Add IP									

Cancel Previous Review and Launch Next: Add Storage

#### Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-04a9880eff53afd3e	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

#### Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(128 characters maximum)	Value	(256 characters maximum)	Instances	Volumes
Name		Santoso-VPC-EC2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more about Amazon EC2 security groups.](#)

Assign a security group:  Create a new security group  Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
All ICMP - I	ICMP	0 - 65535	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

[Add Rule](#)

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click Launch to assign a key pair to your instance and complete the launch process.

**⚠ Improve your instances' security: Your security group, Santoso-VPC-SG, is open to the world.**

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.

You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

**AMI Details** [Edit AMI](#)

**Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-06b263d6ceff0b3dd**

**Free tier eligible** Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

**Instance Type** [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized
t2.micro	Variable	1	1	EBS only	-

**Security Groups** [Edit security groups](#)

Security group name: Santoso-VPC-SG  
Description: launch-wizard-2 created 2020-09-13T13:20:51.824-05:00

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0
All ICMP - IPv4	All	N/A	0.0.0.0/0

**Select an existing key pair or create a new key pair**

A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair:   
AWSSantoso

I acknowledge that I have access to the selected private key file (AWSSantoso.pem), and that without this file, I won't be able to log into my instance.

[Cancel](#) [Launch Instances](#)

**Instance Details** [Edit instance details](#)

**Storage** [Edit storage](#)

[New EC2 Experience](#) Tell us what you think

[EC2 Dashboard](#) [Events](#) [Tags](#) [Limits](#)

[Instances](#)

- [Instances](#)
- [Instance Types](#)
- [Launch Templates](#)
- [Spot Requests](#)
- [Savings Plans](#)
- [Reserved Instances](#)
- [Dedicated Hosts](#) [New](#)
- [Scheduled Instances](#)
- [Capacity Reservations](#)

[Images](#)

- [AMIs](#)

[Elastic Block Store](#)

- [Volumes](#)
- [Snapshots](#)
- [Lifecycle Manager](#)

[Network & Security](#)

- [Security Groups](#) [New](#)
- [Elastic IPs](#) [New](#)
- [Placement Groups](#) [New](#)
- [Key Pairs](#) [New](#)
- [Network Interfaces](#)

[Load Balancing](#)

- [Load Balancers](#)
- [Target Groups](#) [New](#)

[Auto Scaling](#)

- [Launch Configurations](#)
- [Auto Scaling Groups](#)

[Launch Instance](#) [Connect](#) [Actions](#)

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring	Launch Time	Security Groups	Owner
Santoso-VPC...	i-060d7101c38e4229e	t2.micro	us-east-1a	running	2/2 checks ...	None	-	18.209.230.219	-	AWS Santoso	disabled	September 13, 2020 at 1:22:37 PM UTC-5 (less than one hour)	Santoso-VPC-SG	183451715204

Instance: i-060d7101c38e4229e (Santoso-VPC-EC2) Public IP: 18.209.230.219

Description Status Checks Monitoring Tags

Instance ID: i-060d7101c38e4229e	Public DNS (IPv4): -
Instance state: running	IPv4 Public IP: 18.209.230.219
Instance type: t2.micro	IPv6 IPs: -
Finding: Opt-in to AWS Compute Optimizer for recommendations. <a href="#">Learn more</a>	Elastic IPs: -
Private DNS: ip-172-16-1-135.ec2.internal	Availability zone: us-east-1a
Private IPs: 172.16.1.135	Security groups: Santoso-VPC-SG, view inbound rules, view outbound rules
VPC ID: vpc-02da94f22b37d9922 (Santoso-vpc)	Scheduled events: No scheduled events
Platform: Ubuntu	AMI ID: ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20200903 (ami-06b263d6cef0b3dd)
Platform details: Linux/UNIX	Subnet ID: subnet-03fe5efaf86af7f90a (santoso-subnet-public)
Usage operation: Run instances	Network interfaces: eth0
Source/dest. check: True	IAM role: -
T2/T3 Unlimited: Disabled	Key pair name: AWS Santoso
Owner: 183451715204	
Launch time: September 13, 2020 at 1:22:37 PM UTC-5 (less than one hour)	
Termination protection: False	
Lifecycle: normal	
Monitoring: basic	
Alarm status: None	
Kernel ID: -	
RAM disk ID: -	
Placement group: -	
Partition number: -	
Virtualization: hvm	
Reservation: r-09141d0a76aafc3a3	
AMI launch index: 0	
Tenancy: default	
Host ID: -	
Host resource group name: -	
Affinity: -	
State transition reason: -	
State transition reason message: -	
Stop - Hibernation behavior: Disabled	
Number of vCPUs: 1	

# Test Connection (Ping)

```
Command Prompt
Microsoft Windows [Version 10.0.17134.1425]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\budimans>ping 18.209.230.219

Pinging 18.209.230.219 with 32 bytes of data:
Reply from 18.209.230.219: bytes=32 time=41ms TTL=41
Reply from 18.209.230.219: bytes=32 time=41ms TTL=41
Reply from 18.209.230.219: bytes=32 time=42ms TTL=41
Reply from 18.209.230.219: bytes=32 time=41ms TTL=41

Ping statistics for 18.209.230.219:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 41ms, Maximum = 42ms, Average = 41ms

C:\Users\budimans>
```

# Test SSH

The screenshot shows the AWS EC2 Dashboard with the following details:

- Instance:** i-060d7101c38e4229e (Santoso-VPC-EC2) Public IP: 18.209.230.219
- Description:** Instance ID: i-060d7101c38e4229e, Instance state: running, Instance type: t2.micro, Private DNS: ip-172-16-1-135.ec2.internal, Private IPs: 172.16.1.135, Secondary private IPs: VPC ID: vpc-02da94f22b37d9922 (Santoso-vpc), Platform: Ubuntu, Platform details: Linux/UNIX, Usage operation: RunInstances, Source/dest. check: True, T2/T3 Unlimited: Disabled.
- Connect to your instance:** Connection method:  A standalone SSH client [\(i\)](#),  Session Manager [\(i\)](#),  EC2 Instance Connect (browser-based SSH connection) [\(i\)](#).  
To access your instance:
  - Open an SSH client. (find out how to [connect using PuTTY](#))
  - Locate your private key file (AWSSantoso.pem). The wizard automatically detects the key you used to launch the instance.
  - Your key must not be publicly viewable for SSH to work. Use this command if needed:  
`chmod 400 AWSSantoso.pem`
  - Connect to your instance using its Public IP:  
**18.209.230.219****Example:**  
`ssh -i "AWSSantoso.pem" ubuntu@18.209.230.219`

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

```
ubuntu@ip-172-16-1-135: ~
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 41ms, Maximum = 42ms, Average = 41ms

C:\Users\budimans>ssh -i "C:\Users\budimans\OneDrive - University of Texas at Arlington\Cloud\AWS\Keypair\AWSSantoso.pem"
" ubuntu@18.209.230.219
The authenticity of host '18.209.230.219 (18.209.230.219)' can't be established.
ECDSA key fingerprint is SHA256:5ubsM5RNXclb73aBsf60DkR1xsPVF3bqLmf4LTcW45s.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '18.209.230.219' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.3.0-1034-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

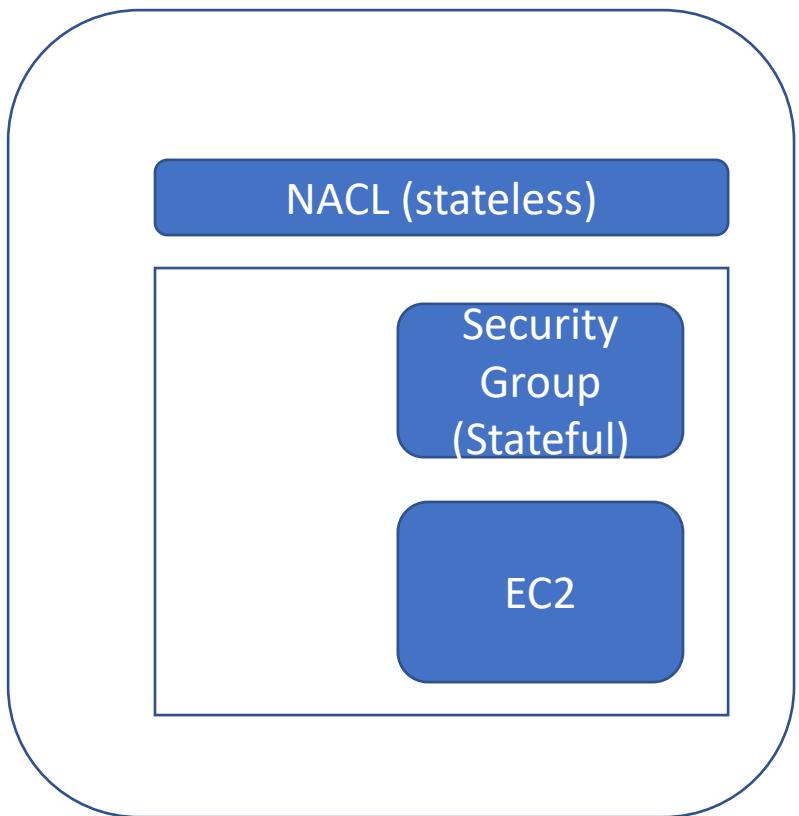
System information as of Sun Sep 13 18:28:47 UTC 2020

System load:  0.01          Processes:      91
Usage of /:   14.4% of 7.69GB  Users logged in:  0
Memory usage: 18%           IP address for eth0: 172.16.1.135
Swap usage:   0%

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
```

# Demo 2 : NACL



- Security Group inbound is set to allow ping (all icmp) from anywhere.
- The NACL default is allowing all inbound and outbound traffic.
- Add an NACL rule to deny outbound icmp.

New EC2 Experience Tell us what you think

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

Instances

- Instances New
- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances New

Placement Groups

Key Pairs

Network Interfaces

Load Balancing

- Load Balancers
- Target Groups New

Auto Scaling

Instances (1/1) Info

Search

Instance state = running X Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
EC2Endpoint	i-0a1623040a73b9b36	<span style="color: green;">Running</span> <span style="color: #ccc;">@</span> <span style="color: #ccc;">Q</span>	t2.micro	<span style="color: green;">2/2 checks passed</span>	No alarms <span style="border: 1px solid #ccc; padding: 2px;">+</span>	us-east-1a	ec2-44-203-34-173.co...	44.203.34.173	-

Instance: i-0a1623040a73b9b36 (EC2Endpoint)

Details Security Networking Storage Status checks Monitoring Tags

▼ Instance summary Info

Instance ID <span style="border: 1px solid #ccc; padding: 2px;">i-0a1623040a73b9b36 (EC2Endpoint)</span>	Public IPv4 address <span style="border: 1px solid #ccc; padding: 2px;">44.203.34.173   open address <span style="color: blue;">Q</span></span>	Private IPv4 addresses <span style="border: 1px solid #ccc; padding: 2px;">10.0.0.235</span>
---	--	---

Inbound rules (3)

Filter security group rules

<input type="checkbox"/>	Name	Security group rule... <span style="color: blue;">▼</span>	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sgr-0748cebe19b2092fa	IPv4	HTTP	TCP	80	0.0.0.0/0	-
<input type="checkbox"/>	-	sgr-07dfe31d17038aa2b	IPv4	All ICMP - IPv4	ICMP	All	0.0.0.0/0	-
<input type="checkbox"/>	-	sgr-073aa113c1e0d88c7	IPv4	SSH	TCP	22	0.0.0.0/0	-

```
C:\Windows\System32>ping 44.203.34.173

Pinging 44.203.34.173 with 32 bytes of data:
Reply from 44.203.34.173: bytes=32 time=40ms TTL=45
Reply from 44.203.34.173: bytes=32 time=40ms TTL=45
Reply from 44.203.34.173: bytes=32 time=79ms TTL=45
Reply from 44.203.34.173: bytes=32 time=40ms TTL=45

Ping statistics for 44.203.34.173:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 40ms, Maximum = 79ms, Average = 49ms

C:\Windows\System32>
```

# Original default NACL

The screenshot shows the AWS Network ACLs console interface. On the left, a navigation sidebar lists various VPC and security-related services. The main area displays the 'Network ACLs' table and two detailed views of the selected NACL.

**Network ACLs (1/3) Table:**

Name	Network ACL ID	Associated with	Default	VPC ID	Inbound rules count	Outbound rules count
acl-06856367b7a30ba...	acl-06856367b7a30ba...	subnet-02febf994fe4d5ec7 / VPCEndpoint	Yes	vpc-07a224409df181773 / VPCEndpo...	2 Inbound rules	2 Outbound rules
acl-012e3fb1b602dc7c4	acl-012e3fb1b602dc7c4	2 Subnets	Yes	vpc-09570d2f4ab052d7a / loadbalanc...	2 Inbound rules	2 Outbound rules
acl-2291d85f	acl-2291d85f	6 Subnets	Yes	vpc-472bcd3a	2 Inbound rules	2 Outbound rules

**Selected NACL (acl-06856367b7a30ba59) Details:**

**Inbound rules (2):**

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	All traffic	All	All	0.0.0.0	Allow
*	All traffic	All	All	0.0.0.0	Deny

**Outbound rules (2):**

Rule number	Type	Protocol	Port range	Destination	Allow/Deny
100	All traffic	All	All	0.0.0.0	Allow
*	All traffic	All	All	0.0.0.0	Deny

# Add inbound rule number 200 deny all ICMP

VPC > Network ACLs > acl-06856367b7a30ba59 > Edit inbound rules

## Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the VPC.

Rule number <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Allow/Deny <small>Info</small>	
100	All traffic	All	All	0.0.0/0	Allow	<button>Remove</button>
200	All ICMP - IPv4	ICMP (1)	All	0.0.0/0	Deny	<button>Remove</button>
*	All traffic	All	All	0.0.0/0	Deny	<button>Remove</button>

[Add new rule](#) [Sort by rule number](#)

```
C:\Windows\System32>ping 44.203.34.173
```

```
Pinging 44.203.34.173 with 32 bytes of data:  
Reply from 44.203.34.173: bytes=32 time=42ms TTL=45  
Reply from 44.203.34.173: bytes=32 time=42ms TTL=45  
Reply from 44.203.34.173: bytes=32 time=42ms TTL=45  
Reply from 44.203.34.173: bytes=32 time=40ms TTL=45
```

```
Ping statistics for 44.203.34.173:  
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
Minimum = 40ms, Maximum = 42ms, Average = 41ms
```

```
C:\Windows\System32>
```

[Cancel](#) [Preview changes](#) [Save changes](#)

Ping still works since the lower rule number 100 allows all traffic

# Change inbound rule number deny all ICMP to rule number 50

VPC > Network ACLs > acl-06856367b7a30ba59 > Edit inbound rules

## Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the VPC.

Rule number <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Allow/Deny <small>Info</small>	
100	All traffic	All	All	0.0.0/0	Allow	<button>Remove</button>
50	All ICMP - IPv4	ICMP (1)	All	0.0.0/0	Deny	<button>Remove</button>
*	All traffic	All	All	0.0.0/0	Deny	

Add new rule Sort by rule number

```
C:\Windows\System32>ping 44.203.34.173

Pinging 44.203.34.173 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 44.203.34.173:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Windows\System32>
```

Cancel Preview changes Save changes

Now it does not work since rule 50 takes precedence over 100.

# Change the inbound rule and remove the deny all icmp.

VPC > Network ACLs > acl-06856367b7a30ba59 > Edit inbound rules

## Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the VPC.

Rule number <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Allow/Deny <small>Info</small>
100	All traffic	All	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

[Add new rule](#) [Sort by rule number](#)

[Cancel](#) [Preview changes](#) [Save changes](#)

```
C:\Windows\System32>ping 44.203.34.173

Pinging 44.203.34.173 with 32 bytes of data:
Reply from 44.203.34.173: bytes=32 time=42ms TTL=45
Reply from 44.203.34.173: bytes=32 time=46ms TTL=45
Reply from 44.203.34.173: bytes=32 time=99ms TTL=45
Reply from 44.203.34.173: bytes=32 time=90ms TTL=45

Ping statistics for 44.203.34.173:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
        Minimum = 42ms, Maximum = 99ms, Average = 69ms

C:\Windows\System32>
```

It works again.

# Add outbound rule to deny all ICMP

## Edit outbound rules Info

Outbound rules control the outgoing traffic that's allowed to leave the VPC.

Rule number <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Destination <small>Info</small>	Allow/Deny <small>Info</small>	
100	All traffic	All	All	0.0.0.0/0	Allow	<button>Remove</button>
50	All ICMP - IPv4	ICMP (1)	All	0.0.0.0/0	Deny	<button>Remove</button>
*	All traffic	All	All	0.0.0.0/0	Deny	<button>Remove</button>
<button>Add new rule</button> <button>Sort by rule number</button>						
						<button>Cancel</button> <button>Preview changes</button> <button>Save changes</button>

```
C:\Windows\System32>ping 44.203.34.173

Pinging 44.203.34.173 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 44.203.34.173:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
                                         
C:\Windows\System32>
```

It does not work.

Remember NACL is stateless unlike SG. Even if the inbound is allowed, it will deny the outbound traffic if there is deny rule.

All of these Accept and Reject can be captured in VPC Flow logs if used.

# Restore the outbound rule

## Edit outbound rules Info

Outbound rules control the outgoing traffic that's allowed to leave the VPC.

Rule number <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Destination <small>Info</small>	Allow/Deny <small>Info</small>	
100	All traffic	All	All	0.0.0.0/0	Allow	<button>Remove</button>
*	All traffic	All	All	0.0.0.0/0	Deny	<button>▼</button>
<a href="#">Add new rule</a> <a href="#">Sort by rule number</a>						
						<a href="#">Cancel</a> <a href="#">Preview changes</a> <a href="#">Save changes</a>

```
C:\Windows\System32>ping 44.203.34.173

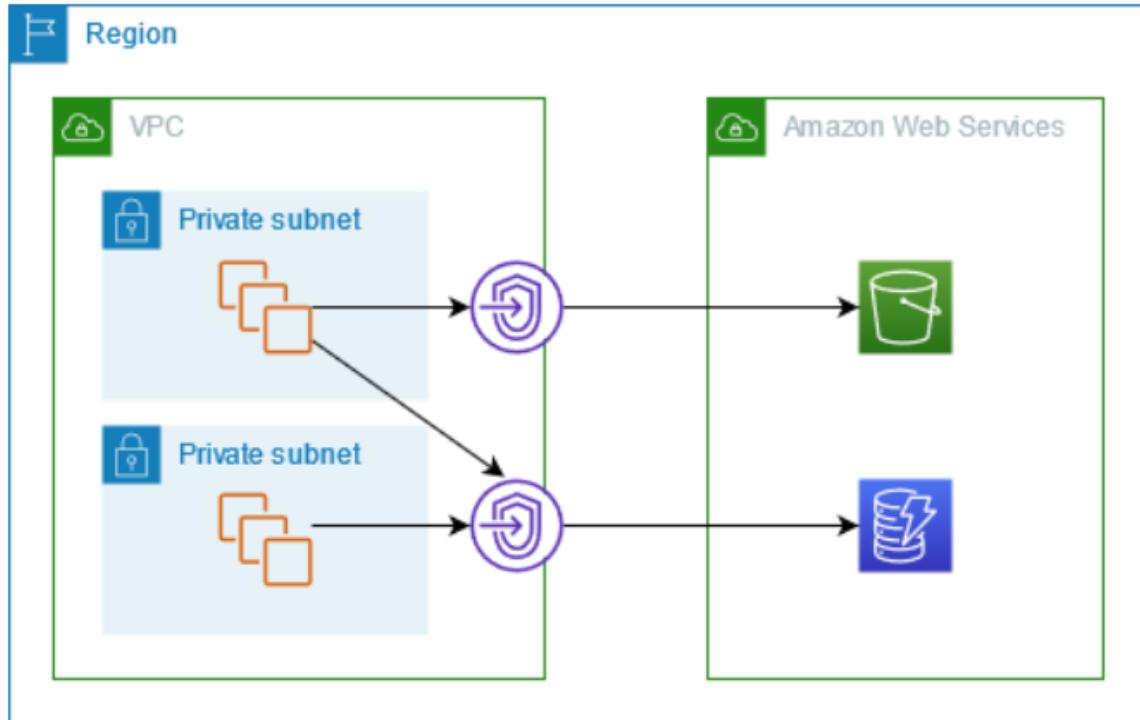
Pinging 44.203.34.173 with 32 bytes of data:
Reply from 44.203.34.173: bytes=32 time=40ms TTL=45
Reply from 44.203.34.173: bytes=32 time=45ms TTL=45
Reply from 44.203.34.173: bytes=32 time=64ms TTL=45
Reply from 44.203.34.173: bytes=32 time=44ms TTL=45

Ping statistics for 44.203.34.173:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 40ms, Maximum = 64ms, Average = 48ms

C:\Windows\System32>
```

It works again. We will continue to the next demo.

# Demo 3: VPC Endpoint – Gateway Endpoint – S3



VPC endpoint is the entry point in your VPC that enables you to connect privately to a service, does not go through the internet.  
Gateway Endpoint only for S3 and DynamoDB  
<https://docs.aws.amazon.com/vpc/latest/privatelink/gateway-endpoints.html>

- In this demo, I have created an object (a picture) in an S3 bucket.
- Using the wget command we can download file from the web. We will use this to test.
  - wget is a networking command-line tool that allow people to download files and interact with REST APIs.
  - wget package is pre-installed in most linux distros
  - Ex: wget http://.....
- Test 1: use the wget command to download the object from the S3 bucket through the internet.
- Test 2: change the Route table so that no traffic will go to the internet gateway. Test wget command and it should not work.
- Test 3: install the Gateway endpoint and update the route table. Test the wget command. If this works, the traffic goes through the private link.

# EC2 original setup and Route Table

The image displays two screenshots of the AWS Management Console. The top screenshot shows the EC2 Instances page with one instance named 'EC2Endpoint' running in the 'us-east-1a' availability zone. The bottom screenshot shows the VPC Route Tables page with one route table associated with a VPC endpoint.

**EC2 Instances (1/1) Info**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
EC2Endpoint	i-0a1623040a73b9b36	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-44-203-34-173.co...	44.203.34.173	-

**Route tables (1/2) Info**

Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Owner ID
-	rtb-0d4b95ec82f471ca3	-	-	Yes	vpc-07a224409df181773   VP...	183451715204
<b>VPCEndpoint</b>	<b>rtb-01cdcc7cc632120fa</b>	<b>subnet-02fefb994fe4d5...</b>	<b>-</b>	<b>No</b>	<b>vpc-07a224409df181773   VP...</b>	<b>183451715204</b>

# Object URL

Amazon S3 X

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose [Provide feedback](#). Provide feedback X

Amazon S3 > Buckets > santoso1demo > folder1/ > 2019\_UTA\_COB\_Status\_0748.jpg

**2019\_UTA\_COB\_Status\_0748.jpg** Info

[Copy S3 URI](#) [Download](#) [Open](#) [Object actions ▾](#)

[Properties](#) [Permissions](#) [Versions](#)

**Object overview**

Owner	S3 URI
santoso.budiman	<a href="s3://santoso1demo/folder1/2019_UTA_COB_Status_0748.jpg">s3://santoso1demo/folder1/2019_UTA_COB_Status_0748.jpg</a>
AWS Region	Amazon Resource Name (ARN)
US East (N. Virginia) us-east-1	<a href="arn:aws:s3:::santoso1demo/folder1/2019_UTA_COB_Status_0748.jpg">arn:aws:s3:::santoso1demo/folder1/2019_UTA_COB_Status_0748.jpg</a>
Last modified	Entity tag (Etag)
June 29, 2022, 18:40:34 (UTC-05:00)	<a href="#">a6e0b4b04de5a8407cfa5a757557c636</a>
Size	Object URL
15.0 MB	<a href="https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Status_0748.jpg">https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Status_0748.jpg</a>
Type	
jpg	
Key	

**Buckets**

- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

**Storage Lens**

- Dashboards
- AWS Organizations settings

**Feature spotlight** 3

AWS Marketplace for S3

# Download Object from S3

- Download is successful through the internet.

```
ubuntu@ip-10-0-0-235:~$ wget https://santosoldemo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
--2022-08-03 01:02:48--  https://santosoldemo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
Resolving santosoldemo.s3.amazonaws.com (santosoldemo.s3.amazonaws.com) ... 52.217.197.89
Connecting to santosoldemo.s3.amazonaws.com (santosoldemo.s3.amazonaws.com)|52.217.197.89|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 15707507 (15M) [image/jpeg]
Saving to: '2019_UTA_COB_Statue_0748.jpg.1'

2019_UTA_COB_Statue_0748.jpg.1          100%[=====] 14.98M  --.-KB/s   in 0.1s

2022-08-03 01:02:49 (103 MB/s) - '2019_UTA_COB_Statue_0748.jpg.1' saved [15707507/15707507]

ubuntu@ip-10-0-0-235:~$ ls
2019_UTA_COB_Statue_0748.jpg  2019_UTA_COB_Statue_0748.jpg.1
ubuntu@ip-10-0-0-235:~$
```

# Create Gateway Endpoint and update Route Table

Interface (ENI) as an entry point for traffic destined to the service. Interface endpoints are typically accessed using the public or private DNS name associated with the service, while Gateway endpoints and Gateway Load Balancer endpoints serve as a target for a route in your route table for traffic destined for the service.

### Endpoint settings

Name tag - optional  
Creates a tag with a key of 'Name' and a value that you specify.

Service category  
Select the service category

AWS services  
Services provided by Amazon

PrivateLink Ready partner services  
Services with an AWS Service Ready designation

AWS Marketplace services  
Services that you've purchased through AWS Marketplace

Other endpoint services  
Find services shared with you by service name

### Services (1/3)

Filter services

Service Name: com.amazonaws.us-east-1.s3

Service Name	Owner	Type
<input checked="" type="radio"/> com.amazonaws.us-east-1.s3	amazon	Gateway
<input type="radio"/> com.amazonaws.us-east-1.s3	amazon	Interface
<input type="radio"/> com.amazonaws.us-east-1.s3-outposts	amazon	Interface

### VPC

Select the VPC in which to create the endpoint

VPC  
The VPC in which to create your endpoint.

### Route tables (1/2)

Filter route tables

Name	Route Table ID	Main
-	rtb-0d4b95ec82f471ca3	Yes
<input checked="" type="checkbox"/> VPCEndpoint	rtb-01cdcc7cc632120fa (VPCEndpoint)	No

When you use an endpoint, the source IP addresses from your instances in your affected subnets for accessing the AWS service in the same region will be private IP addresses, not public IP addresses. Existing connections from your affected subnets to the AWS service that use public IP addresses may be dropped. Ensure that you don't have critical tasks running when you create or modify an endpoint.

rtb-01cdcc7cc632120fa

### Policy

VPC endpoint policy controls access to the service.

Full access  
Allow access by any user or service within the VPC using credentials from any Amazon Web Services accounts to any resources in this Amazon Web Services service. All policies — IAM user policies, VPC endpoint policies, and Amazon Web Services service-specific policies (e.g. Amazon S3 bucket policies, any S3 ACL policies) — must grant the necessary permissions for access to succeed.

Custom  
Use the [policy creation tool](#) to generate a policy, then paste the generated policy below.

**New VPC Experience**

**Successfully created VPC endpoint**  
vpce-06745d876f7dbc32c

**Endpoints (1/1)** [Info](#)

[Filter endpoints](#)  [Clear filters](#)

Name	VPC endpoint ID	VPC ID	Service name	Endpoint type	Status
VPCEndpointS3	vpce-06745d876f7dbc32c	vpc-07a224409df181773   VPCEndpoint	com.amazonaws.us-east-1.s3	Gateway	

**Route tables (1/2)** [Info](#)

[Filter route tables](#)  [Clear filters](#)

Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Owner ID
-	rtb-0d4b95ec82f471ca3	-	-	Yes	vpc-07a224409df181773   VP...	183451715204
<b>VPCEndpoint</b>	<b>rtb-01cdcc7cc632120fa</b>	<b>subnet-02febf994fe4d5...</b>	<b>-</b>	<b>No</b>	<b>vpc-07a224409df181773   VP...</b>	<b>183451715204</b>

**rtb-01cdcc7cc632120fa / VPCEndpoint**

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

**Routes (3)**

Destination	Target	Status	Propagated
pl-63a5400a	vpce-06745d876f7dbc32c		No
0.0.0.0/0	igw-0ff717c15305e6058		No
10.0.0.0/16	local		No

**The Route Table is updated automatically.**

```
ubuntu@ip-10-0-0-235:~$ wget https://santosoldemo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
--2022-08-03 01:20:17-- https://santosoldemo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
Resolving santosoldemo.s3.amazonaws.com (santosoldemo.s3.amazonaws.com) ... 52.217.80.108
Connecting to santosoldemo.s3.amazonaws.com (santosoldemo.s3.amazonaws.com)|52.217.80.108|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 15707507 (15M) [image/jpeg]
Saving to: '2019_UTA_COB_Statue_0748.jpg.2'

2019_UTA_COB_Statue_0748.jpg.2          100%[=====] 14.98M  78.5MB/s    in 0.2s

2022-08-03 01:20:17 (78.5 MB/s) - '2019_UTA_COB_Statue_0748.jpg.2' saved [15707507/15707507]

ubuntu@ip-10-0-0-235:~$
```

It is working without going through the internet

# Create an EC2 without Public IP in the same subnet just for testing

**Launch an instance Info**

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags Info**

Name: EC2EndpointInternalIPonly

**Application and OS Images (Amazon Machine Image) Info**

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

**Recents** **Quick Start**

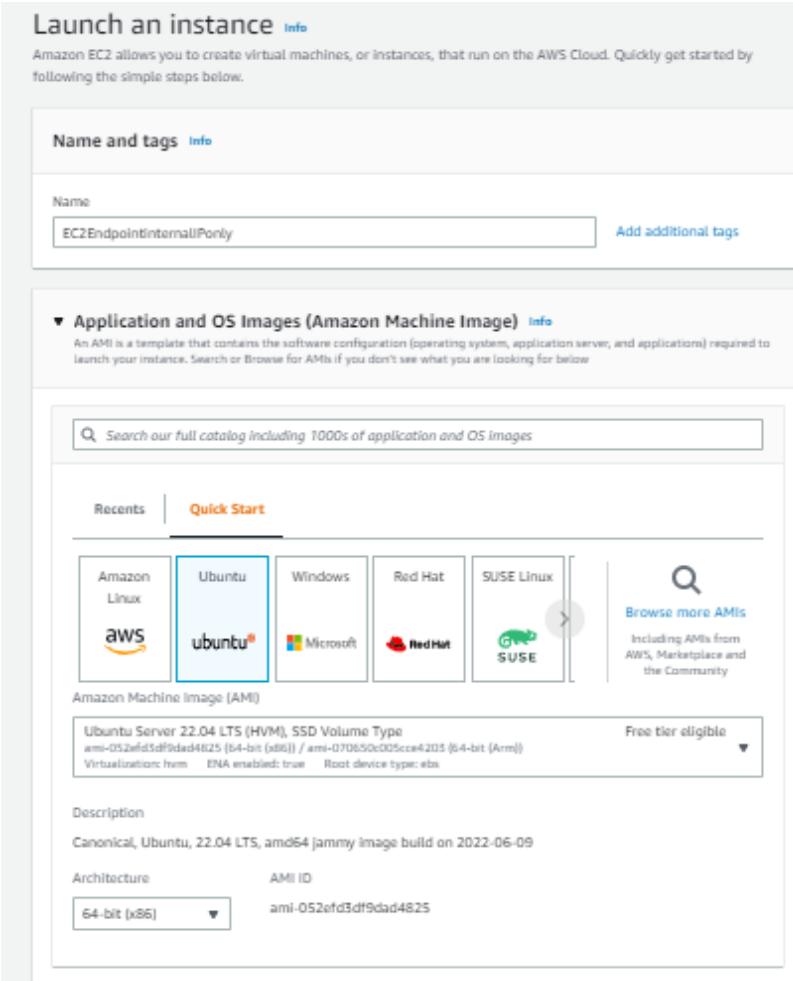
**Amazon Linux** **Ubuntu** **Windows** **Red Hat** **SUSE Linux**  Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type  
ami-052efd3df9dad4825 (64-bit (x86)) / ami-070650c005cce4203 (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

**Description**  
Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2022-06-09

**Architecture** 64-bit (x86) **AMI ID** ami-052efd3df9dad4825



**Instance type Info**

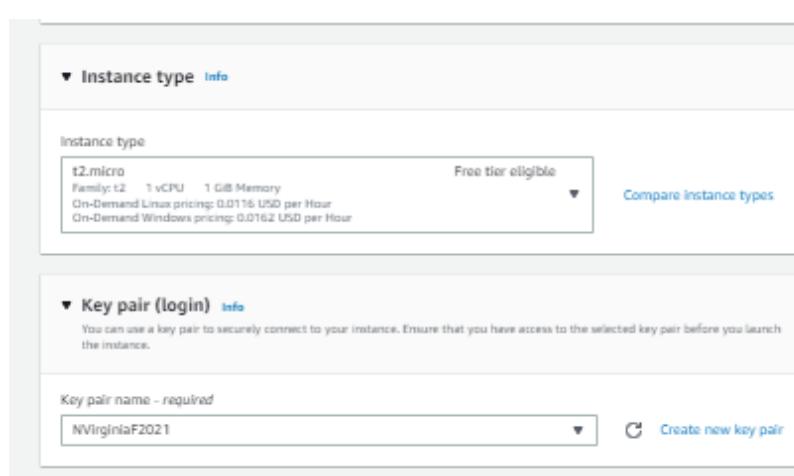
Instance type: t2.micro  
Family: t2 1 vCPU 1 GiB Memory  
On-Demand Linux pricing: 0.0116 USD per Hour  
On-Demand Windows pricing: 0.0162 USD per Hour

Free tier eligible

**Key pair (login) Info**

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required: NVirginiaF2021



**Network settings Get guidance**

**VPC - required Info**  
vpc-07a224409df181773 (VPCEndpoint)  
10.0.0.0/16

**Subnet Info**  
subnet-02fefb994fe45ec7  
VPC: vpc-07a224409df181773 Owner: 183451715204 Availability Zone: us-east-1a IP addresses available: 250

**Auto-assign public IP Info**  
Disable

**Firewall (security groups) Info**  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group  Select existing security group

**Common security groups Info**

EC2endpoint-StackSecurityGroup-2929YC6R8C2B sg-00409d39456a1a52f   
VPC: vpc-07a224409df181773

Security groups that you add or remove here will be added to or removed from all your network interfaces.

**Advanced network configuration**

**Configure storage Info**

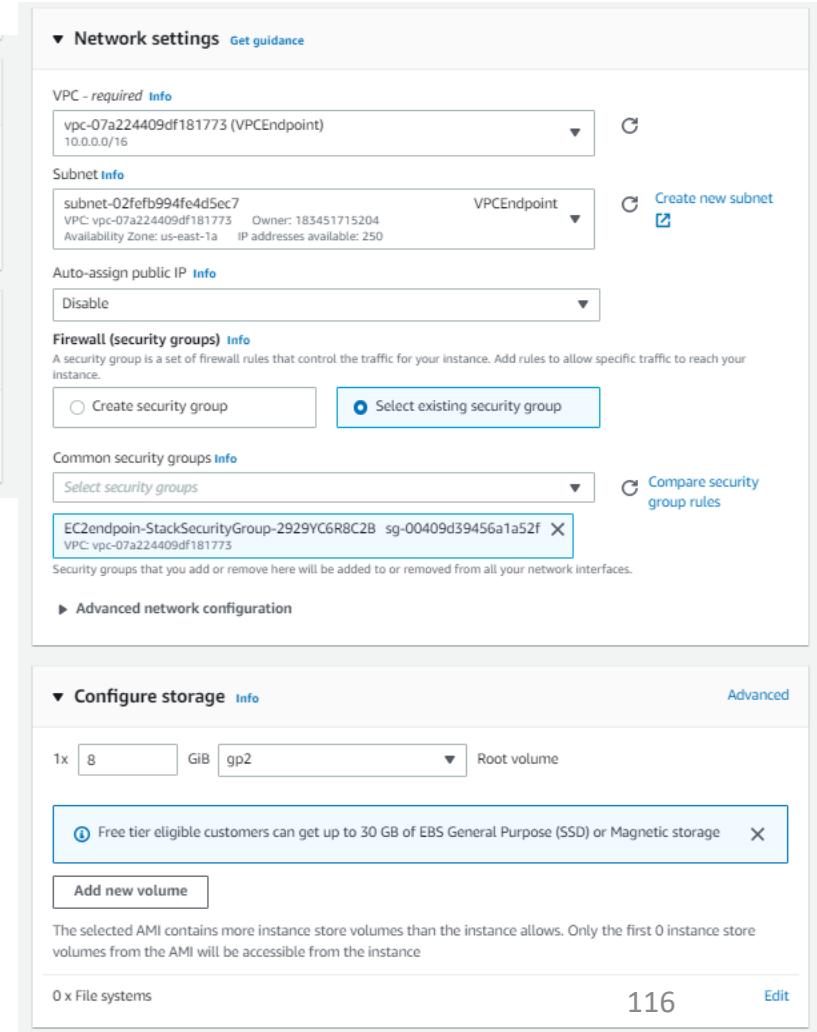
Advanced

1x 8 GiB gp2

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems



New EC2 Experience [Tell us what you think](#)

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

Instances

Instances [New](#)

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances [New](#)

Dedicated Hosts

Scheduled Instances

Capacity Reservations

Images

AMIs [New](#)

## Instances (1/2) [Info](#)

Search  Instance state = running [X](#) Clear filters

C Connect Instance state Actions Launch instances

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
EC2Endpoint	i-0a1623040a73b9b36	<span>Running</span> <a href="#">View details</a>	t2.micro	<span>2/2 checks passed</span> <a href="#">View details</a>	No alarms <a href="#">+ Create alarm</a>	us-east-1a	ec2-44-203-34-173
EC2EndpointinternalIPonly	i-01282462a41837efc	<span>Running</span> <a href="#">View details</a>	t2.micro	<span>2/2 checks passed</span> <a href="#">View details</a>	No alarms <a href="#">+ Create alarm</a>	us-east-1a	-

### Instance: i-01282462a41837efc (EC2EndpointinternalIPonly)

Details Security Networking Storage Status checks Monitoring Tags

Instance summary [Info](#)

Instance ID <a href="#">i-01282462a41837efc (EC2EndpointinternalIPonly)</a>	Public IPv4 address -	Private IPv4 addresses <a href="#">10.0.0.193</a>
IPv6 address -	Instance state <span>Running</span>	Public IPv4 DNS -

# SSH to EC2EndpointinternalIPonly from EC2Endpoint (don't forget the .pem file)

Then do the wget in EC2EndpointinternalIPonly EC2. Remember this EC2 does not have a public IP so it must go to through the Gateway endpoint.

```
ubuntu@ip-10-0-0-193:~$ wget https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Status_0748.jpg
--2022-08-03 01:40:28-- https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Status_0748.jpg
Resolving santoso1demo.s3.amazonaws.com (santoso1demo.s3.amazonaws.com) ... 54.231.228.169
Connecting to santoso1demo.s3.amazonaws.com (santoso1demo.s3.amazonaws.com)|54.231.228.169|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 15707507 (15M) [image/jpeg]
Saving to: '2019_UTA_COB_Status_0748.jpg'

2019_UTA_COB_Status_0748.jpg 100%[=====] 14.98M --.-KB/s in 0.1s

2022-08-03 01:40:29 (101 MB/s) - '2019_UTA_COB_Status_0748.jpg' saved [15707507/15707507]

ubuntu@ip-10-0-0-193:~$ ls
2019_UTA_COB_Status_0748.jpg
ubuntu@ip-10-0-0-193:~$
```

Proof it does not go out to the internet since it doesn't have public ip and must go through the gateway endpoint to s3.

```
ubuntu@ip-10-0-0-193:~$ ping -c3 google.com
PING google.com (172.253.62.138) 56(84) bytes of data.

--- google.com ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2036ms

ubuntu@ip-10-0-0-193:~$
```

But the other EC2 with Public IP can go out of the internet.

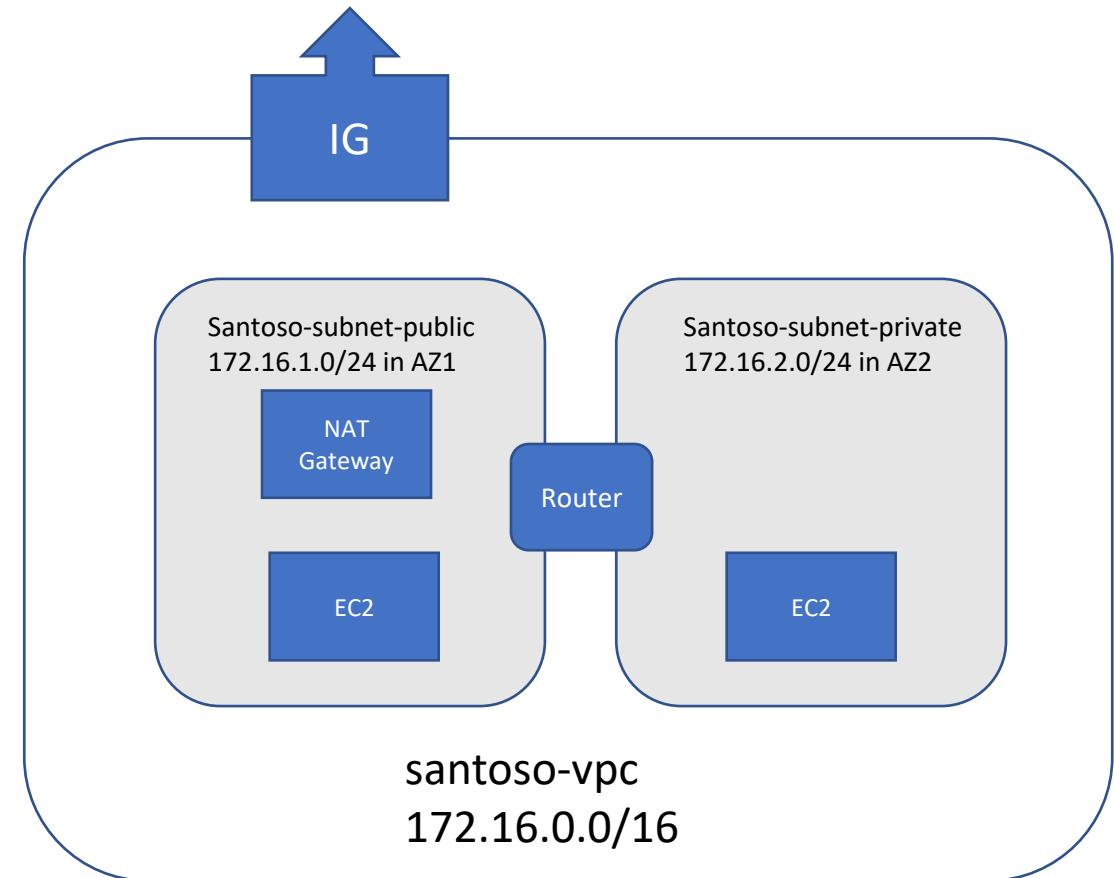
```
ubuntu@ip-10-0-0-235:~$ ping -c3 google.com
PING google.com (172.253.115.102) 56(84) bytes of data.
64 bytes from bg-in-f102.1e100.net (172.253.115.102): icmp_seq=1 ttl=95 time=1.87 ms
64 bytes from bg-in-f102.1e100.net (172.253.115.102): icmp_seq=2 ttl=95 time=1.93 ms
64 bytes from bg-in-f102.1e100.net (172.253.115.102): icmp_seq=3 ttl=95 time=1.98 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.866/1.925/1.980/0.046 ms
ubuntu@ip-10-0-0-235:~$
```

# Demo 4

# VPC Demo 4

- VPC Name: santoso-vpc
- CIDR: 172.16.0.0/16
- Subnet 1: santoso-subnet-public
- Subnet 1 CIDR: 172.16.1.0/24
- Subnet 2: santoso-subnet-private
- Subnet 2 CIDR: 172.16.2.0/24



# Steps

- Create VPC
- Create IG and attach to the VPC
- Create Public Subnet in AZ1
- Create Route Table for the public subnet
- Create EC2 in the public subnet – allow ssh and ping from anywhere
- Create Private Subnet in AZ2
- Create Route Table for the private subnet
- Create EC2 in the private subnet – allow ssh and ping from anywhere
- Create NAT Gateway in the public subnet
- Update the Route Table for the Private Subnet

# Create VPC

aws Services Search [Alt+S] N. Virginia ▾

VPC > Your VPCs > Create VPC

## Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

**VPC settings**

Resources to create Info  
Create only the VPC resource or the VPC and other networking resources.

VPC only  VPC and more

Name tag - *optional*  
Creates a tag with a key of 'Name' and a value that you specify.  
santoso-vpc

IPv4 CIDR block Info  
 IPv4 CIDR manual input  IPAM-allocated IPv4 CIDR block

IPv4 CIDR  
172.16.0.0/16

IPv6 CIDR block Info  
 No IPv6 CIDR block  IPAM-allocated IPv6 CIDR block  Amazon-provided IPv6 CIDR block  IPv6 CIDR owned by me

Tenancy Info  
Default ▾

123

# Create IG and Attach to the VPC

aws Services  X

VPC > Internet gateways > Create internet gateway

### Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

**Internet gateway settings**

**Name tag**  
Creates a tag with a key of 'Name' and a value that you specify.

**Tags - optional**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="santoso-vpc"/> X
<button>Remove</button>	
<button>Add new tag</button>	

You can add 49 more tags.

Cancel **Create internet gateway**

aws Services  X

VPC > Internet gateways > Attach to VPC (igw-0a21722f78468c752)

### Attach to VPC (igw-0a21722f78468c752) Info

**VPC**  
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

**Available VPCs**  
Attach the internet gateway to this VPC.

vpc-074eb80faa111b54b - santoso-vpc  
▶ AWS Command Line Interface command

Cancel **Attach internet gateway**

# Create Public Subnet

aws Services  X

VPC > Subnets > Create subnet

### Create subnet Info

**VPC**

VPC ID  
Create subnets in this VPC.

Associated VPC CIDRs

IPv4 CIDRs  
172.16.0.0/16

**Subnet settings**  
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

Subnet name  
Create a tag with a key of 'Name' and a value that you specify.  
  
The name can be up to 256 characters long.

Availability Zone Info  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 CIDR block Info  
 X

▼ Tags - optional

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="Santoso-public"/> X
<input type="button" value="Remove"/>	

You can add 49 more tags.

VPC > Subnets >  Edit subnet settings

### Edit subnet settings Info

**Subnet**

Subnet ID	Name
<input type="text" value="subnet-0f66f4c55f7c1e6c3"/>	<input type="text" value="Santoso-public"/>

**Auto-assign IP settings Info**  
Enable the auto-assign IP settings to automatically request a public IPv4 or IPv6 address for a new network interface in this subnet.

Enable auto-assign public IPv4 address Info

Enable auto-assign customer-owned IPv4 address Info  
Option disabled because no customer owned pools found.

# Create Route Table for the Public Subnet

aws | Services  X

VPC > Route tables > Create route table

### Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

#### Route table settings

Name - optional  
Create a tag with a key of 'Name' and a value that you specify.

VPC  
The VPC to use for this route table.

#### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="Santoso-Public-RT"/>

You can add 49 more tags.

## Add route to IG

VPC > Route tables > rtb-0166a8fdc5eec2608 > Edit routes

### Edit routes

Destination	Target	Status	Propagated
172.16.0.0/16	<input type="text" value="Q local"/> <input checked="" type="radio"/> Active	No	<input type="button" value="Remove"/>
Q 0.0.0.0	<input type="text" value="Q igw-0321722f78468c754"/>	-	<input type="button" value="Remove"/>

## Associate to the public subnet

VPC dashboard X  
EC2 Global View New  
Filter by VPC:

Virtual private cloud  
Your VPCs  
Subnets  
**Route tables**  
Internet gateways  
Egress-only Internet gateways  
Carrier gateways  
DHCP option sets  
Elastic IPs  
Managed prefix lists  
Endpoints  
Endpoint services  
NAT gateways  
Peering connections  
Security  
Network ACLs  
Security groups

You have successfully updated subnet associations for rtb-0166a8fdc5eec2608 / Santoso-Public-RT.

VPC > Route tables > rtb-0166a8fdc5eec2608  
**rtb-0166a8fdc5eec2608 / Santoso-Public-RT**

You can now check network connectivity with Reachability Analyzer

### Details Info

Route table ID <input type="text" value="rtb-0166a8fdc5eec2608"/>	Main <input checked="" type="checkbox"/> No	Explicit subnet associations <a href="#">subnet-0f66f4c55f7c1e6c3 / Santoso-public</a>
VPC <input type="text" value="vpc-074eb80faa111b54b   santoso-vpc"/>	Owner ID <input type="text" value="183451715204"/>	

#### Explicit subnet associations (1)

Find subnet association
<input type="text" value="subnet-0f66f4c55f7c1e6c3 / Santoso-public"/>

Subnet ID:  IPv4 CIDR:  IPv6 CIDR:

# Create EC2 in the public subnet – allow ssh and ping from anywhere

The image consists of three side-by-side screenshots from the AWS Management Console.

**Screenshot 1: Launch an instance (Step 1)**

This screenshot shows the initial steps of launching an EC2 instance:

- Name and tags:** Instance name is "EC2Public1".
- Application and OS Images (Amazon Machine Image):** Search bar for "ubuntu".
- Recent AMIs:** Includes Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and Microsoft.
- AMI Selection:** Ubuntu Server 22.04 LTS (HVM), SSD Volume Type, ami-0574da719dca65348 (64-bit (x86)) / ami-0e2b332e6356bc05 (64-bit (Arm)).
- Description:** Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2022-12-01.
- Architecture:** 64-bit (x86).
- AMI ID:** ami-0574da719dca65348.
- Verified provider:** Verified.

**Screenshot 2: Instance type and network settings (Step 2)**

This screenshot shows the configuration of the instance type and network settings:

- Instance type:** t2.micro (Free tier eligible). Comparison link: [Compare instance types](#).
- Key pair (login):** NVirginiaF2021.
- Network settings:** VPC: vpc-074eb80faa111b54b (santoso-vpc), Subnet: subnet-0f66f4c55f7c1e6c3 (Santoso-public).
- Auto-assign public IP:** Enabled.

**Screenshot 3: Security group configuration (Step 3)**

This screenshot shows the configuration of the security group:

- Enable:** Firewall (security groups) Info.
- Security group name - required:** EC2Public1.
- Inbound security groups rules:**
  - Security group rule 1 (TCP, 22, 0.0.0.0/0):** Type: ssh, Protocol: TCP, Port range: 22, Source type: Anywhere, Description: e.g. SSH for admin desktop.
  - Security group rule 2 (ICMP, All, 0.0.0.0/0):** Type: All ICMP - IPv4, Protocol: ICMP, Port range: All, Source type: Anywhere, Description: e.g. SSH for admin desktop.

# Test EC2Public1

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with 'New EC2 Experience' (on), 'EC2 Dashboard', and 'EC2 Global View'. The main area has a title 'Instances (4) Info' with a search bar 'Find instance by attribute or tag (case-sensitive)'. Below is a table with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv..., and Public IPv4... (with dropdown arrows). One row is selected: 'EC2Public1' (Name), 'i-066babaa395b019cd' (Instance ID), 'Running' (InstanceState), 't2.micro' (Instance type), 'No alarms' (Alarm status), 'us-east-1a' (Availability Zone), '3.231.213.130' (Public IPV4), and '3.231.213.130' (Public IPV4). There are 'Edit' and 'Delete' buttons at the top right of the table.

```
C:\Windows\System32>ping 3.231.213.130

Pinging 3.231.213.130 with 32 bytes of data:
Reply from 3.231.213.130: bytes=32 time=46ms TTL=39
Reply from 3.231.213.130: bytes=32 time=45ms TTL=39
Reply from 3.231.213.130: bytes=32 time=47ms TTL=39
Reply from 3.231.213.130: bytes=32 time=45ms TTL=39

Ping statistics for 3.231.213.130:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 45ms, Maximum = 47ms, Average = 45ms

C:\Windows\System32>
```

# Create Private Subnet and the route table

VPC > Subnets > Create subnet

## Create subnet Info

**VPC**

VPC ID  
Create subnets in this VPC.  
vpc-074eb80faa111b54b (santoso-vpc)

Associated VPC CIDRs  
IPv4 CIDRs  
172.16.0.0/16

**Subnet settings**  
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

Subnet name  
Create a tag with a key of 'Name' and a value that you specify.  
Santoso-Private

The name can be up to 256 characters long.

Availability Zone Info  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.  
US East (N. Virginia) / us-east-1b

IPv4 CIDR block Info  
Q 172.16.2.0/24 X

▼ Tags - optional

Key	Value - optional
Q Name X	Q Santoso-Private X Remove

Add new tag

You can add 49 more tags.

# Create The route table

You have successfully updated subnet associations for rtb-01d00db988e6fc9fe / Santoso-Private-RT.

VPC dashboard EC2 Global View New Filter by VPC: Select a VPC Virtual private cloud Your VPCs Subnets Route tables Internet gateways Egress-only internet gateways Carrier gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services NAT gateways Peering connections Security Network ACLs Security groups Network Analysis

VPC > Route tables > rtb-01d00db988e6fc9fe

rtb-01d00db988e6fc9fe / Santoso-Private-RT

You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer Actions

**Details** Info

Route table ID rtb-01d00db988e6fc9fe	Main No	Explicit subnet associations subnet-0330c85cde23f9917 / Santoso-Private	Edge associations -
VPC vpc-074eb80faa111b54b   santoso-vpc	Owner ID 183451715204		

Routes Subnet associations Edge associations Route propagation Tags

**Routes (1)**

Destination	Target	Status	Propagated
172.16.0.0/16	local	Active	No

Edit routes < 1 > ⚙

# Create EC2Private

The screenshot illustrates the process of creating a private Amazon EC2 instance. It shows the 'Launch an instance' wizard, the instance configuration details, and the resulting EC2 instance summary.

**EC2 > Instances > Launch an instance**

**Launch an instance**

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags**

Name: EC2Private1

**Application and OS Images (Amazon Machine Image)**

Search our full catalog including 1000s of application and OS images

Recent AMIs: My AMIs, Quick Start

Available AMIs:

- Amazon Linux
- macOS
- Ubuntu** (selected)
- Windows
- Red Hat

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

**Ubuntu Server 22.04 LTS (HVM), SSD Volume Type**

ami-0574da719dca65548 (64-bit (x86)) / ami-0e2b332e63c56bc5 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description: Canonical, Ubuntu, 22.04 LTS, amd64 Jammy image build on 2022-12-01

Architecture: 64-bit (x86)

AMI ID: ami-0574da719dca65548

Verified provider

**Instance type**

t2.micro

Family: t2 1 vCPU 1 GiB Memory

On-Demand Linux pricing: 0.0116 USD per Hour

On-Demand Windows pricing: 0.0162 USD per Hour

Free tier eligible

Compare instance types

**Key pair (login)**

Key pair name - required: NVirginiaF2021

**Create new key pair**

**Network settings**

VPC - required

vpc-074eb80faa111b54b (santoso-vpc)

172.16.0.16

Subnet info

subnet-0330c85cde23f9917

Santoso-Private

VPC: vpc-074eb80faa111b54b Owner: 183451715204

Availability Zone: us-east-1b IP addresses available: 251 CIDR: 172.16.2.0/24

**Create new subnet**

Auto-assign public IP

Disable

**Firewall (security groups) Info**

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

**Security group name - required**

EC2Private

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_.-:/@!+=;&|\$^

**Description - required**

launch-wizard-1 created 2022-12-10T15:27:20.458Z

**Inbound security groups rules**

**Security group rule 1 (TCP, 22, 0.0.0.0/0)**

Type: ssh Protocol: TCP Port range: 22

Source type: Anywhere Source: 0.0.0.0/0 Description: e.g. SSH for admin desktop

**Remove**

**Security group rule 2 (ICMP, All, 0.0.0.0/0)**

Type: All ICMP - IPv4 Protocol: ICMP Port range: All

Source type: Anywhere Source: 0.0.0.0/0 Description: e.g. SSH for admin desktop

**Remove**

**Instances (5) Info**

Find instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability zone	Public IPv4	Public IPv4	Elastic IP	IPv6 I
EC2Public1	i-066babaa395b019cd	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	-	3.231.213.130	-	-
EC2Private1	i-0a1ba6684fb25aeeb	Running	t2.micro	-	No alarms	us-east-1b	-	-	-	-

**New EC2 Experience**

Tell us what you think

EC2 Dashboard

EC2 Global View

Events

131

# SSH to public EC2 and ping private EC2

```
ubuntu@ip-172-16-1-68:~$ #this is public EC2
ubuntu@ip-172-16-1-68:~$ #ping to private EC2
ubuntu@ip-172-16-1-68:~$ ping -c3 172.16.2.61
PING 172.16.2.61 (172.16.2.61) 56(84) bytes of data.
64 bytes from 172.16.2.61: icmp_seq=1 ttl=64 time=1.18 ms
64 bytes from 172.16.2.61: icmp_seq=2 ttl=64 time=0.761 ms
64 bytes from 172.16.2.61: icmp_seq=3 ttl=64 time=0.788 ms

--- 172.16.2.61 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.761/0.910/1.181/0.191 ms
ubuntu@ip-172-16-1-68:~$
```

# SSH to private EC2 from public EC2

```
ubuntu@ip-172-16-1-68:~$ chmod 700 NVirginiaF2021.pem  
ubuntu@ip-172-16-1-68:~$ ssh -i "NVirginiaF2021.pem" ubuntu@172.16.2.61
```

```
ubuntu@ip-172-16-2-61:~$ #we are in private EC2  
ubuntu@ip-172-16-2-61:~$ #try ping google.com  
ubuntu@ip-172-16-2-61:~$ ping -c3 www.google.com  
PING www.google.com (142.251.163.106) 56(84) bytes of data.  
  
--- www.google.com ping statistics ---  
3 packets transmitted, 0 received, 100% packet loss, time 2032ms  
  
ubuntu@ip-172-16-2-61:~$
```

# Create NAT Gateway in the public subnet

The screenshot shows the AWS VPC dashboard with the 'NAT gateways' section selected. On the left, a sidebar lists various VPC-related services: VPC dashboard, EC2 Global View (New), Filter by VPC (Select a VPC dropdown), Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services), NAT gateways (selected), and Peering connections.

The main content area is titled 'NAT gateways' with an 'Info' link. It features a search bar labeled 'Filter NAT gateways'. A table header includes columns: Name, NAT gateway ID, Connectivity..., State, State message, Elastic IP address, Primary private ..., Network interface ID, VPC, Subnet, and Created. Below the table, a message reads 'No NAT gateways found'. At the bottom of the page, there is a button labeled 'Select a NAT gateway'.

# Must be in Public subnet and must allocate Elastic IP

Elastic IP address 23.21.102.70 (eipalloc-0418b968da3dd5986) allocated.

VPC > NAT gateways > Create NAT gateway

### Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

#### NAT gateway settings

Name - *optional*  
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Subnet  
Select a subnet in which to create the NAT gateway.

Connectivity type  
Select a connectivity type for the NAT gateway.

Public  
 Private

Elastic IP allocation ID Info  
Assign an Elastic IP address to the NAT gateway.

► Additional settings

#### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key  Value - *optional*

You can add 49 more tags.

# Update private subnet Routing Table

VPC > Route tables > rtb-01d00db988e6fc9fe > Edit routes

## Edit routes

Destination	Target	Status	Propagated
172.16.0.0/16	local	Active	No
0.0.0.0/0	nat nat-0c14805d66ebd4da2 (NAT-santosoVPC)	-	No
<a href="#">Add route</a>			

Cancel   Preview   **Save changes**

# Test outbound traffic (may take sometimes to set up)

```
ubuntu@ip-172-16-2-61:~$ ping -c3 google.com
PING google.com (172.253.115.113) 56(84) bytes of data.

--- google.com ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2043ms

ubuntu@ip-172-16-2-61:~$ ping -c3 google.com
PING google.com (172.253.115.138) 56(84) bytes of data.
64 bytes from bg-in-f138.1e100.net (172.253.115.138): icmp_seq=1 ttl=102 time=2.90 ms
64 bytes from bg-in-f138.1e100.net (172.253.115.138): icmp_seq=2 ttl=102 time=2.47 ms
64 bytes from bg-in-f138.1e100.net (172.253.115.138): icmp_seq=3 ttl=102 time=2.41 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 2.410/2.592/2.902/0.220 ms

ubuntu@ip-172-16-2-61:~$ ping -c3 www.google.com
PING www.google.com (142.251.163.147) 56(84) bytes of data.
64 bytes from wv-in-f147.1e100.net (142.251.163.147): icmp_seq=1 ttl=98 time=2.41 ms
64 bytes from wv-in-f147.1e100.net (142.251.163.147): icmp_seq=2 ttl=98 time=2.02 ms
64 bytes from wv-in-f147.1e100.net (142.251.163.147): icmp_seq=3 ttl=98 time=2.08 ms

--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 2.019/2.170/2.412/0.172 ms

ubuntu@ip-172-16-2-61:~$
```

```
ubuntu@ip-172-16-2-61:~$ #ping publicEC2 using its public IP
ubuntu@ip-172-16-2-61:~$ ping -c3 3.231.213.130
PING 3.231.213.130 (3.231.213.130) 56(84) bytes of data.
64 bytes from 3.231.213.130: icmp_seq=1 ttl=62 time=1.35 ms
64 bytes from 3.231.213.130: icmp_seq=2 ttl=62 time=0.994 ms
64 bytes from 3.231.213.130: icmp_seq=3 ttl=62 time=0.911 ms

--- 3.231.213.130 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.911/1.086/1.353/0.191 ms

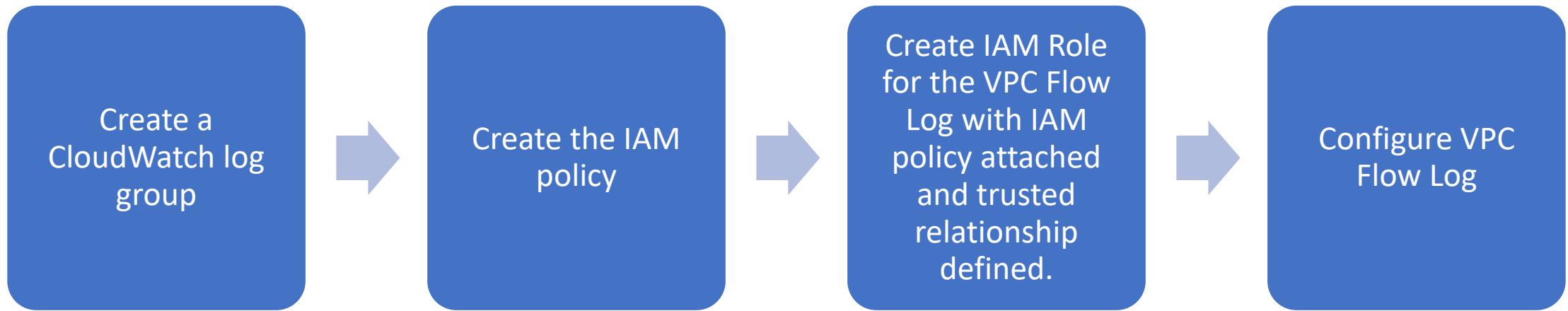
ubuntu@ip-172-16-2-61:~$
```

# Demo 5 (VPC Flow Logs)

# Demo steps

- Create a VPC
  - 2 subnets
  - IG
  - 2EC2s with their Security Groups
- Set up VPC Flow log to publish to a CloudWatch log
  - Create a CloudWatch log group
  - Create an IAM policy to allow necessary access to CloudWatch
  - Create IAM role for the VPC Flow log
    - Attach the IAM policy
    - Define the trusted relationship (allows the flow logs service to assume the role)

# Steps to setup VPC Flow Log to publish in CloudWatch logs



<https://docs.aws.amazon.com/vpc/latest/tgw/flow-logs-cwl.html>

# Demo shortcut

- We have not talked about IAM and the process of creating IAM role and policy is a bit involved.
- Since the purpose of the demo is to show a way to see IP traffic going to and from network interfaces in your VPC and to speed up the demo, I have created and will use the below in the demo.
  - IAM Policy: BudimanVPCflowlog
  - IAM Role : BudimanVPCflowlog
- So demo order:
  - Create CloudWatch log group
  - Configure VPC Flow Log

Create a VPC + 2 subnets + 2  
EC2s

# Create a VPC with 2 subnets

The screenshot shows the AWS VPC dashboard interface. At the top, there's a navigation bar with tabs for 'VPC dashboard' (selected), 'EC2 Global View' (New), and 'Virtual private cloud'. Below the navigation is a search bar labeled 'Filter VPCs'.

**Your VPCs (1/2) Info**

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main
VPCFlowLogs	vpc-0a717923dd34ac49c	Available	10.0.0.0/16	-	dopt-ebbe3d91	rtb-0

**Subnets (2) Info**

Search filters: search: VPCFlowLogs X | Clear filters

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	Available IP
-	subnet-016f272071f2880d2	Available	vpc-0a717923dd34ac49c   VP...	10.0.64.0/18	-	16378
-	subnet-00f8bbf617303aef5	Available	vpc-0a717923dd34ac49c   VP...	10.0.0.0/18	-	16378

# Spin up an EC2 in each subnet: 1<sup>st</sup> EC2

New EC2 Experience  Tell us what you think

EC2 Dashboard  
EC2 Global View  
Events  
Tags  
Limits

Instances  Instances [New](#)  
Instance Types  
Launch Templates  
Spot Requests  
Savings Plans  
Reserved Instances [New](#)  
Dedicated Hosts  
Scheduled Instances  
Capacity Reservations

### Instances (1/2) Info

Find instance by attribute or tag (case-sensitive)

Instance state = running  Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Avail...
EC2Test1	i-03c393d29a7c6e289	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1
EC2Test2	i-0add2b12bf77c0f55	Running	t2.micro	Initializing	No alarms	+ us-east-1

Instance: i-03c393d29a7c6e289 (EC2Test1)

Use RBN as guest OS hostname  Disabled Answer RBN DNS hostname IPv4  Enabled

Network Interfaces (1) Info

Interface ID	Description	IPv4 Prefixes	IPv6 Prefixes	Public IPv4 address	Private IPv4 address	Private IPv4 DNS	IPv6
eni-06905c9c5ad38a...	-	-	-	100.27.17.204	10.0.53.78	ip-10-0-53-78.ec2.int...	-

Instances  Instances [New](#)  
Instance Types  
Launch Templates  
Spot Requests  
Savings Plans  
Reserved Instances [New](#)  
Dedicated Hosts  
Scheduled Instances  
Capacity Reservations

Instance: i-03c393d29a7c6e289 (EC2Test1)

Security groups  sg-0a0f43d571fd4c57a (EC2Test1)

Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Des...
-	sgr-0c74066bdd94c638b	All	ICMP	0.0.0.0/0	EC2Test1 <input type="checkbox"/>	-
-	sgr-0434b3ba7b60897a5	22	TCP	0.0.0.0/0	EC2Test1 <input type="checkbox"/>	-

# Spin up an EC2 in each subnet: 2nd EC2

The screenshot shows the AWS EC2 Instances page with two running instances listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability zone
EC2Test1	i-03c393d29a7c6e289	Running	t2.micro	2/2 checks passed	No alarms	us-east-1
EC2Test2	i-0add2b12bf77c0f55	Running	t2.micro	2/2 checks passed	No alarms	us-east-1

The details for EC2Test2 are expanded, showing its network interfaces:

Interface ID	Description	IPv4 Prefixes	IPv6 Prefixes	Public IPv4 address	Private IPv4 address	Private IPv4 DNS	IPv6 address
eni-0f5da3bf9a79f7caa	-	-	-	3.83.8.162	10.0.84.32	ip-10-0-84-32.ec2.int...	-

A modal window for EC2Test2 displays its configuration details:

IAM Role	Owner ID	Launch time
-	183451715204	Sun Dec 11 2022 08:00:43 GMT-0600 (Central Standard Time)

The modal also shows the inbound rules for the instance:

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Des...
-	sgr-0fe0cf0cf690f0998	22	TCP	0.0.0.0/0	EC2Test2	-

# Create VPC Flow Log

# Create CloudWatch group

# Create CloudWatch Log

The screenshot shows the AWS CloudWatch interface for creating a new log group. The left sidebar includes links for Favorites and recents, Dashboards, Alarms (with 2 alerts), Logs (selected, showing Log groups New), Logs Insights, Metrics, X-Ray traces, Events, Application monitoring, and Insights. The main content area shows the 'Create log group' page with the breadcrumb 'CloudWatch > Log groups > Create log group'. The 'Log group details' section contains fields for 'Log group name' (set to 'VPCflowlogdemo'), 'Retention setting' (set to 'Never expire'), and 'KMS key ARN - optional' (empty). The 'Tags' section explains what tags are and provides an 'Add new tag' button, noting a limit of up to 50 tags. At the bottom are 'Cancel' and 'Create' buttons.

CloudWatch

Favorites and recents

Dashboards

Alarms

Logs

Log groups New

Logs Insights

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings New

Getting Started

CloudWatch > Log groups > Create log group

## Create log group

**Log group details**

Log group name

VPCflowlogdemo

Retention setting

Never expire

KMS key ARN - optional

**Tags**

A tag is a label that you assign to an Amazon Web Services resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your Amazon Web Services costs.

No tags are associated with this log group.

Add new tag

You can add up to 50 more tag(s).

Cancel

Create

# Create an IAM Role for the VPC Flow log

## To create an IAM role for flow logs

1. Open the IAM console at <https://console.aws.amazon.com/iam/>.
  2. In the navigation pane, choose **Roles**, **Create role**.
  3. For **Select type of trusted entity**, choose **AWS service**. For **Use case**, choose **EC2**. Choose **Next: Permissions**.
  4. On the **Attach permissions policies** page, choose **Next: Tags** and optionally add tags. Choose **Next: Review**.
  5. Enter a name for your role and optionally provide a description. Choose **Create role**.
  6. Select the name of your role. For **Permissions**, choose **Add inline policy, JSON**.
  7. Copy the first policy from [IAM roles for publishing flow logs to CloudWatch Logs](#) and paste it in the window. Choose **Review policy**.
  8. Enter a name for your policy, and choose **Create policy**.
  9. Select the name of your role. For **Trust relationships**, choose **Edit trust relationship**. In the existing policy document, change the service from `ec2.amazonaws.com` to `vpc-flow-logs.amazonaws.com`. Choose **Update Trust Policy**.
  10. On the **Summary** page, note the ARN for your role. You need this ARN when you create your flow log.
- <https://docs.aws.amazon.com/vpc/latest/tgw/flow-logs-cwl.html>

# Create IAM Policy

## Create policy

1 2 3

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor

JSON

Import managed policy

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Action": [  
7         "logs:CreateLogGroup",  
8         "logs:CreateLogStream",  
9         "logs:PutLogEvents",  
10        "logs:DescribeLogGroups",  
11        "logs:DescribeLogStreams"  
12      ],  
13      "Resource": "*"  
14    }  
15  ]  
16 }
```

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

Character count: 199 of 6,144.

Cancel

Next: Tags

# Create Policy

Create policy

Review policy

Name\* BudimanVPCflowlog

Description to attach to VPC flow log role

Summary

Service	Access level	Resource	Request condition
CloudWatch Logs	Limited: List, Write	All resources	None

Tags

Key	Value
No tags associated with the resource.	

\* Required

Cancel Previous Create policy

# Create Role

IAM > Roles > Create role

Step 1

Select trusted entity

Step 2

Add permissions

Step 3

Name, review, and create

## Select trusted entity Info

### Trusted entity type

AWS service

Allow AWS services like EC2, Lambda, or others to perform actions in this account.

AWS account

Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

Web identity

Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

SAML 2.0 federation

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

Custom trust policy

Create a custom trust policy to enable others to perform actions in this account.

### Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

#### Common use cases

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

#### Use cases for other AWS services:

Choose a service to view use case ▾

Cancel

Next

# Create Role

IAM > Roles > Create role

Step 1  
Select trusted entity

Step 2  
**Add permissions**

Step 3  
Name, review, and create

### Add permissions Info

**Permissions policies (Selected 1/826) Info**

Choose one or more policies to attach to your new role.

Filter policies by property or policy name and press enter. 1 match

"Budiman"

<input checked="" type="checkbox"/>	Policy name <small>Info</small>	Type	Description
<input checked="" type="checkbox"/>	+ BudimanVPCflowlog	Custom...	to attach to VPC flow log role

**► Set permissions boundary - optional Info**

Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting, but you can use it to delegate permission management to others.

Cancel  Previous  Next

**Identity and Access Management (IAM)**

Search IAM

Dashboard

**Access management**

- User groups
- Users

**Roles**

- Policies
- Identity providers
- Account settings

**Access reports**

- Access analyzer
- Archive rules
- Analyzers
- Settings
- Credential report
- Organization activity
- Service control policies (SCPs)

Related consoles

IAM Identity Center New

**IAM > Roles > Create role**

Step 1 Select trusted entity

Step 2 Add permissions

Step 3 Name, review, and create

## Name, review, and create

### Role details

Role name  
Enter a meaningful name to identify this role.

Maximum 64 characters. Use alphanumeric and '+,-,@,\_' characters.

Description  
Add a short explanation for this role.

Maximum 1000 characters. Use alphanumeric and '+,-,@,\_' characters.

### Step 1: Select trusted entities

**Edit**

```
1 [{}]
2 "Version": "2012-10-17",
3 "Statement": [
4     {
5         "Effect": "Allow",
6         "Action": [
7             "sts:AssumeRole"
8         ],
9         "Principal": {
10             "Service": [
11                 "ec2.amazonaws.com"
12             ]
13         }
14     }
15 ]
```

# IAM Role is created, now edit the Trusted Relationship

Screenshot of the AWS IAM Roles page showing the configuration of the 'BudimanVPCflowlog' role.

**BudimanVPCflowlog**  
Allows VPC flow log to publish to CloudWatch

**Summary**

Creation date	ARN	Instance profile ARN
December 11, 2022, 18:10 (UTC-06:00)	<a href="#">arn:aws:iam::183451715204:role/BudimanVPCflowlog</a>	<a href="#">arn:aws:iam::183451715204:instance-profile/BudimanVPCflowlog</a>
Last activity	Maximum session duration	
None	1 hour	

**Permissions**   **Trust relationships**   **Tags**   **Access Advisor**   **Revoke sessions**

**Permissions policies (1) Info**  
You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter.

<input type="checkbox"/>	Policy name	Type	Description
<input type="checkbox"/>	<a href="#">BudimanVPCflowlog</a>	Customer managed	to attach to VPC flow log role

**Permissions boundary - (not set) Info**  
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others.

**Related consoles** [IAM Identity Center](#) New

155

# Edit Trust Policy

Identity and Access Management (IAM) X

IAM > Roles > BudimanVPCflowlog > Edit trust policy

### Edit trust policy

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles**
- Policies
- Identity providers
- Account settings

Access reports

- Access analyzer
- Archive rules
- Analyzers
- Settings
- Credential report
- Organization activity
- Service control policies (SCPs)

Related consoles

IAM Identity Center New

```
1 ~ {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Principal": {  
7         "Service": "vpc-flow-logs.amazonaws.com"  
8       },  
9       "Action": "sts:AssumeRole"  
10     }  
11   ]  
12 }
```

**Edit statement**

Select a statement  
Select an existing statement in the policy or add a new statement.

**Add new statement**

**Add new statement**

JSON Ln 12, Col 2

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

Preview external access

Cancel Update policy

Trust policy updated.

IAM > Roles > BudimanVPCflowlog

## BudimanVPCflowlog

Allows VPC flow log to publish to CloudWatch

Delete

### Summary

Edit

Creation date

December 11, 2022, 18:10 (UTC-06:00)

ARN

arn:aws:iam::183451715204:role/BudimanVPCflowlog

Instance profile ARN

arn:aws:iam::183451715204:instance-profile/BudimanVPCflowlog

Last activity

None

Maximum session duration

1 hour

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

### Trusted entities

Edit trust policy

Entities that can assume this role under specified conditions.

```
1 - [ {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Principal": {  
7         "Service": "vpc-flow-logs.amazonaws.com"  
8       },  
9       "Action": "sts:AssumeRole"  
10     }  
11   ]  
12 } ]
```

# Create VPC flow log

The screenshot shows the AWS VPC dashboard with the following details:

**VPC ID:** `vpc-0a717923dd34ac49c`

**State:** Available

**Tenancy:** Default

**DHCP option set:** `dopt-ebbe3d91`

**Default VPC:** No

**IPv4 CIDR:** `10.0.0.0/16`

**Network Address Usage metrics:** Disabled

**DNS hostnames:** Enabled

**Main route table:** `rtb-0465fb559da4b1ca7`

**IPv6 pool:** -

**Owner ID:** `183451715204`

**Actions:** `Actions ▾`

**Flow logs:** `Create flow log`

**Filter flow logs:** `Filter flow logs`

**Table Headers:** Name, Flow log ID, Filter, Destination type, Destination name

**Left sidebar:**

- VPC dashboard X
- EC2 Global View New
- Filter by VPC:  
Select a VPC ▾
- Virtual private cloud ▼
- Your VPCs
- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- Endpoints
- Endpoint services
- NAT gateways
- Peering connections
- Security ▼
- Network ACLs

VPC > Your VPCs > Create flow log

## Create flow log [Info](#)

Flow logs can capture IP traffic flow information for the network interfaces associated with your resources. You can create multiple flow logs to send traffic to different destinations.

**Selected resources [Info](#)**

Name	Resource ID	State
VPCFlowLogs	vpc-0a717923dd34ac49c	<span style="color: green;">Available</span>

**Flow log settings**

Name - *optional*

Filter  
 Accept  
 Reject  
 All

Maximum aggregation interval [Info](#)  
The maximum interval of time during which a flow of packets is captured and aggregated into a flow log record.  
 10 minutes  
 1 minute

**Filter**  
The type of traffic to capture (accepted traffic only, rejected traffic only, or all traffic).  
 Accept  
 Reject  
 All

**Maximum aggregation interval [Info](#)**  
The maximum interval of time during which a flow of packets is captured and aggregated into a flow log record.  
 10 minutes  
 1 minute

**Destination**  
The destination to which to publish the flow log data.  
 Send to CloudWatch Logs  
 Send to an Amazon S3 bucket  
 Send to Kinesis Firehose in the same account  
 Send to Kinesis Firehose in a different account

**Destination log group [Info](#)**  
The name of the Amazon CloudWatch log group to which the flow log is published. A new log stream is created for each monitored network interface.  
 X C

**IAM role [Info](#)**  
The IAM role that has permission to publish to the Amazon CloudWatch log group. [Set up permissions](#) D  
 ▼ C

**Log record format**  
Specify the fields to include in the flow log record.  
 AWS default format  
 Custom format

Format preview

VPC dashboard X

EC2 Global View New

Filter by VPC:  
Select a VPC ▾

Virtual private cloud

Your VPCs

- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- Endpoints
- Endpoint services
- NAT gateways
- Peering connections

Security

Network ACLs

Security groups

Network Analysis

Successfully created flow log for vpc-0a717923dd34ac49c.

VPC > Your VPCs > vpc-0a717923dd34ac49c

## vpc-0a717923dd34ac49c / VPCFlowLogs

Actions ▾

Details		Info	
VPC ID	vpc-0a717923dd34ac49c	State	Available
Tenancy	Default	DHCP option set	dopt-ebbe3d91
Default VPC	No	IPv4 CIDR	10.0.0.0/16
Network Address Usage metrics	Disabled	Route 53 Resolver DNS Firewall rule groups	-
		DNS hostnames	Enabled
		Main route table	rtb-0465fb559da4b1ca7
		IPv6 pool	-
		Owner ID	183451715204
		DNS resolution	Enabled
		Main network ACL	acl-08ee3a9a9bbebf47b
		IPv6 CIDR (Network border group)	-

CIDRs Flow logs Tags

### Flow logs (1/1) Info

Filter flow logs Actions ▾ Create flow log

<input checked="" type="checkbox"/> Name	Flow log ID	Filter	Destination type	Destination name	IAM role ARN
BudimanFlowlog	fl-02aa207e64ed1420a	ALL	cloud-watch-logs	VPCflowlogdemo <span style="color: blue;">Edit</span>	arn:aws:iam::183451715204:role/VPCflowlogdemo

◀ ▶ ⏪ ⏩ ⏴ ⏵ 160

# CloudWatch

CloudWatch X

Favorites and recents ▶

Dashboards

Alarms ⚠ 2 ✔ 7 ⋯ 0

In alarm

All alarms

Billing

Logs

Log groups New

Logs Insights

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings New

Getting Started

CloudWatch > Log groups > VPCflowlogdemo

## VPCflowlogdemo

Actions ▾ View in Logs Insights Search log group

▼ Log group details

ARN	Metric filters	Data protection - new
arn:aws:logs:us-east-1:183451715204:log-group:VPCflowlogdemo:*	0	Inactive
Creation time	Subscription filters	Sensitive data found - new
35 minutes ago	0	-
Retention	Contributor Insights rules	KMS key ID
Never expire	-	-
Stored bytes		
-		

Log streams Metric filters Subscription filters Contributor Insights Tags Data protection - new

Log streams (2)

Filter log streams or try prefix search   Exact match

C Delete Create log stream Search all log streams

<input type="checkbox"/> Log stream	Last event time
<input type="checkbox"/> eni-0f5da3bf9a79f7caa-all	2022-12-11 18:18:42 (UTC-06:00)
<input type="checkbox"/> eni-06905c9c5ad38a79d-all	2022-12-11 18:18:22 (UTC-06:00)

# CloudWatch creates a log stream per ENI

Instance: i-03c393d29a7c6e289 (EC2Test1)

▼ Network Interfaces (1) [Info](#)

Filter network interfaces	
Interface ID	Description
<a href="#">eni-06905c9c5ad38a79d</a>	-

Instance: i-0add2b12bf77c0f55 (EC2Test2)

▼ Network Interfaces (1) [Info](#)

Filter network interfaces	
Interface ID	Description
<a href="#">eni-0f5da3bf9a79f7caa</a>	-

# Flow log syntax

Flow log syntax:

```
<version> <account-id> <interface-id> <srcaddr> <dstaddr> <srcport> <dstport> <protocol> <packets> <bytes>  
<start> <end> <action> <log-status>
```

# CloudWatch

CloudWatch X

Favorites and recents ▶

Dashboards

Alarms ▾ ⚠ 2 ✓ 7 ⏪ 0

- In alarm
- All alarms
- Billing

Logs ▾

- Log groups New
- Logs Insights

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings New

Getting Started

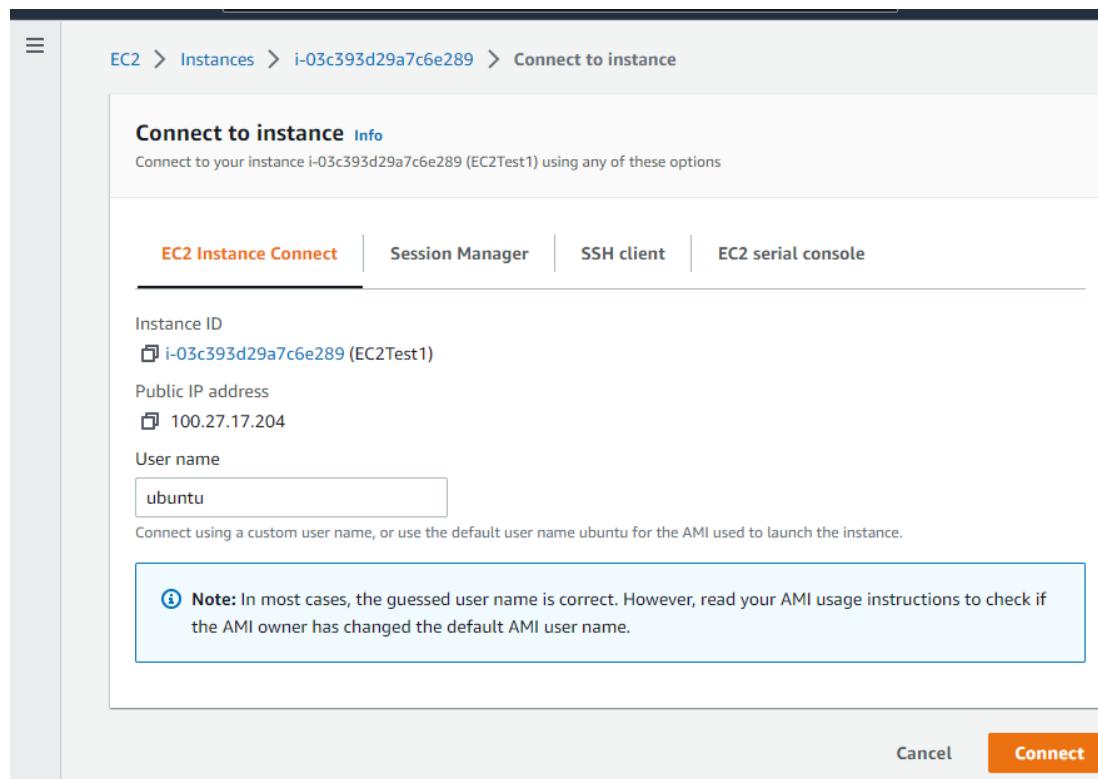
CloudWatch > Log groups > VPCflowlogdemo > eni-06905c9c5ad38a79d-all

**Log events**  
You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events Clear 1m 30m 1h 12h Custom Display ⚙

▶	Timestamp	Message
There are older events to load. <a href="#">Load more</a> .		
▶	2022-12-11T18:21:21.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 64.142.54.12 10.0.53.78 123 35046 17 1 76 1670804481 1670804541 ACCEPT OK
▶	2022-12-11T18:21:21.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 10.0.53.78 64.142.54.12 35046 123 17 1 76 1670804481 1670804541 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 91.240.118.77 10.0.53.78 49691 5616 6 1 40 1670804549 1670804601 REJECT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 61.177.173.36 10.0.53.78 51119 22 6 1 923 1670804549 1670804601 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 10.0.53.78 61.177.173.36 22 51119 6 9 9108 1670804549 1670804601 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 89.248.165.51 10.0.53.78 50561 5969 6 1 40 1670804549 1670804601 REJECT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 61.177.173.51 10.0.53.78 12053 22 6 3 180 1670804549 1670804601 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 10.0.53.78 61.177.173.51 22 12053 6 8 480 1670804549 1670804601 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 91.189.94.4 10.0.53.78 123 59828 17 1 76 1670804549 1670804601 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 10.0.53.78 91.189.94.4 59828 123 17 1 76 1670804549 1670804601 ACCEPT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 152.32.201.23 10.0.53.78 51065 9348 6 1 44 1670804549 1670804601 REJECT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 92.63.196.154 10.0.53.78 58354 4400 6 1 40 1670804549 1670804601 REJECT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 80.82.77.234 10.0.53.78 41739 9993 6 1 40 1670804549 1670804601 REJECT OK
▶	2022-12-11T18:22:29.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 78.128.113.254 10.0.53.78 44240 55681 6 1 40 1670804549 1670804601 REJECT OK
▶	2022-12-11T18:23:23.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 152.32.141.108 10.0.53.78 36952 3363 6 1 60 1670804603 1670804661 REJECT OK
▶	2022-12-11T18:23:23.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 94.102.61.46 10.0.53.78 38299 1193 6 1 44 1670804603 1670804661 REJECT OK
▶	2022-12-11T18:23:23.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 61.177.173.52 10.0.53.78 64613 22 6 3 180 1670804603 1670804661 ACCEPT OK
▶	2022-12-11T18:23:23.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 10.0.53.78 61.177.173.52 22 64613 6 7 420 1670804603 1670804661 ACCEPT OK
▶	2022-12-11T18:23:23.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 155.94.208.21 10.0.53.78 55942 1849 6 1 40 1670804603 1670804661 REJECT OK
▶	2022-12-11T18:23:23.000-06:00	2 183451715204 eni-06905c9c5ad38a79d 162.142.125.92 10.0.53.78 43116 12378 6 1 44 1670804603 1670804661 REJECT OK

# SSH with EC2 Instance Connect



# Delete Demo resources

- Delete IAM Roles and Policy
- Delete CloudWatch log group
- Delete NAT Gateway
- Terminate EC2s
- Delete VPC
- Release Elastic IP address of the NAT Gateway

# HW pointers

- Use What's my IP to find the external IP address of your laptop
- Use SSH Client when only your laptop is allowed (EC2 Instance Connect will not work since the IP address is not your laptop)

# End of Lecture