INSY 5345 & INSY 4307

Dr. Santoso Budiman

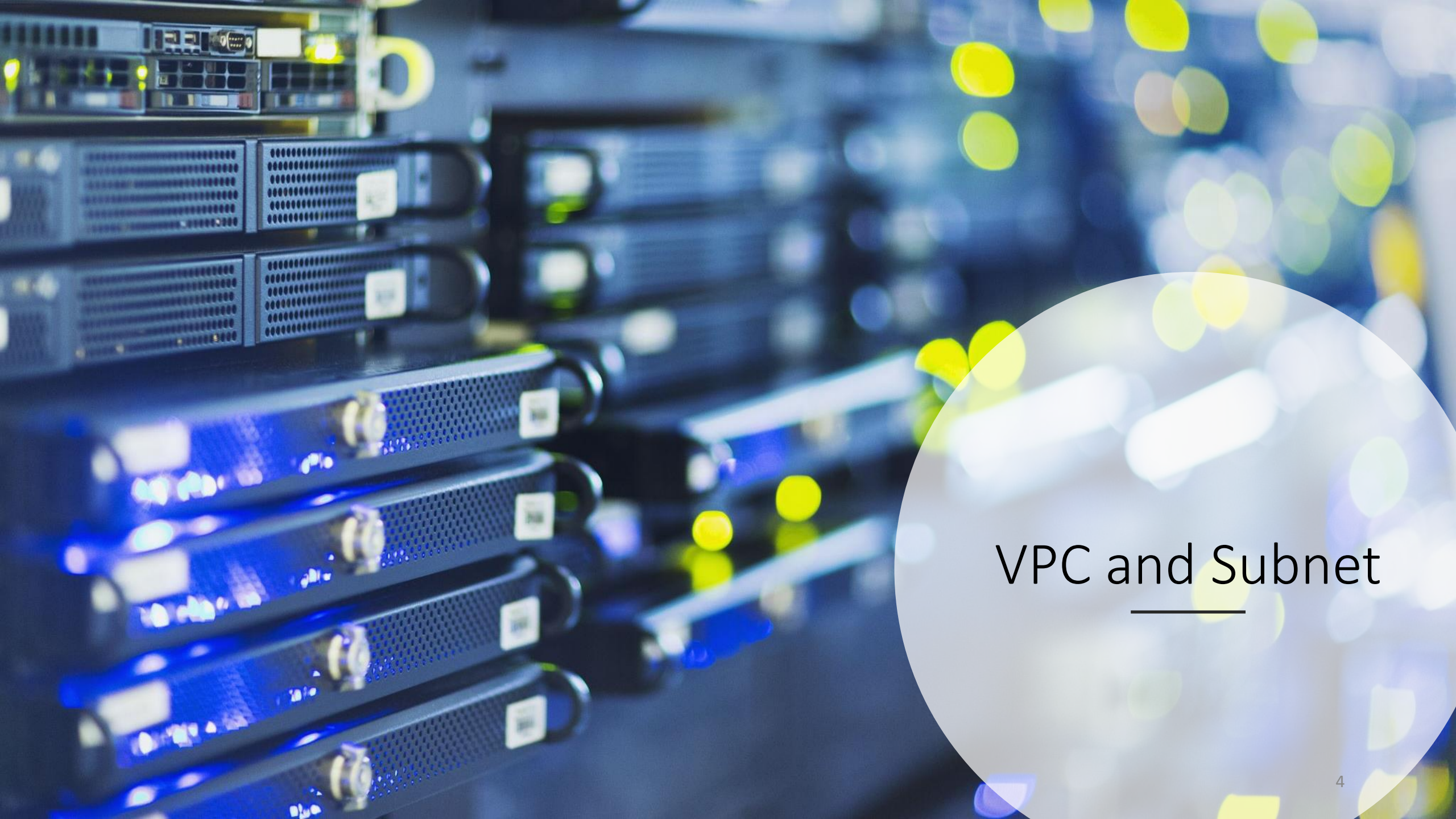
Cloud Computing Theory and Practice

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



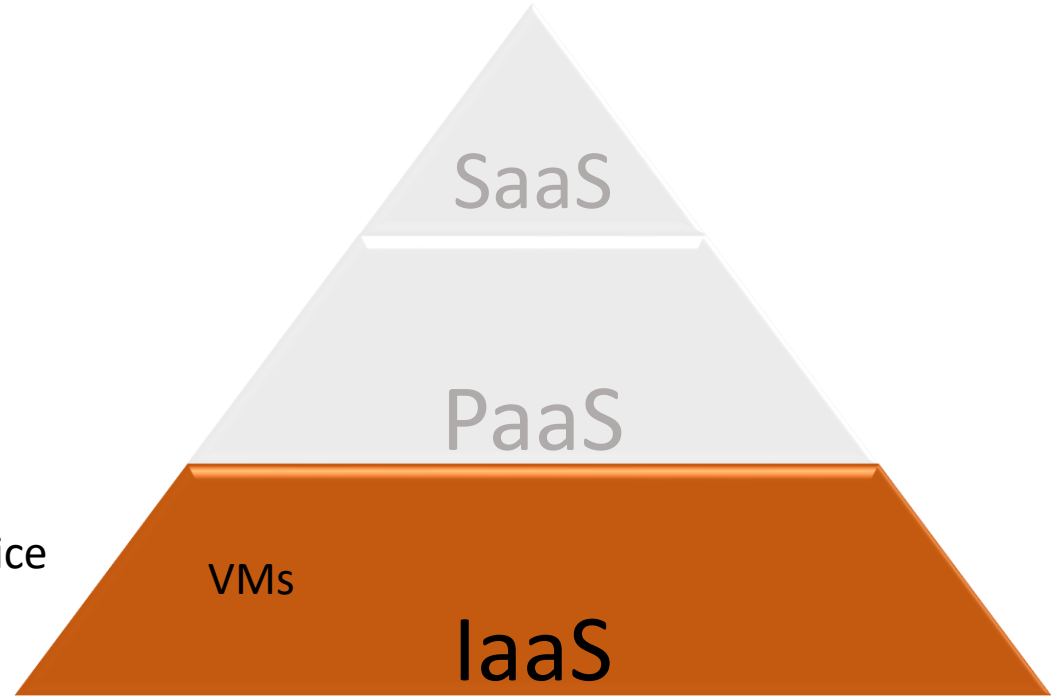
VPC and Subnet

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

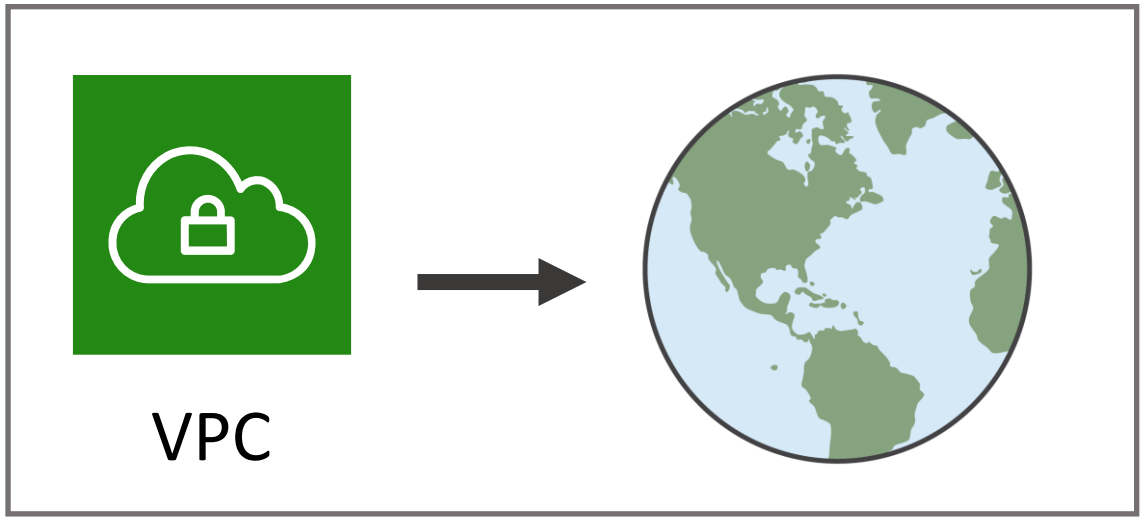
AWS Virtual Private Cloud

Default (DON'T DELETE default VPC and its SUBNETS)	Customized
• 1 VPC per AWS Region • CIDR - /16 • A subnet per AZ (/20) • An internet Gateway is attached	• 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	= 10.22.33.44
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**
- 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

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)

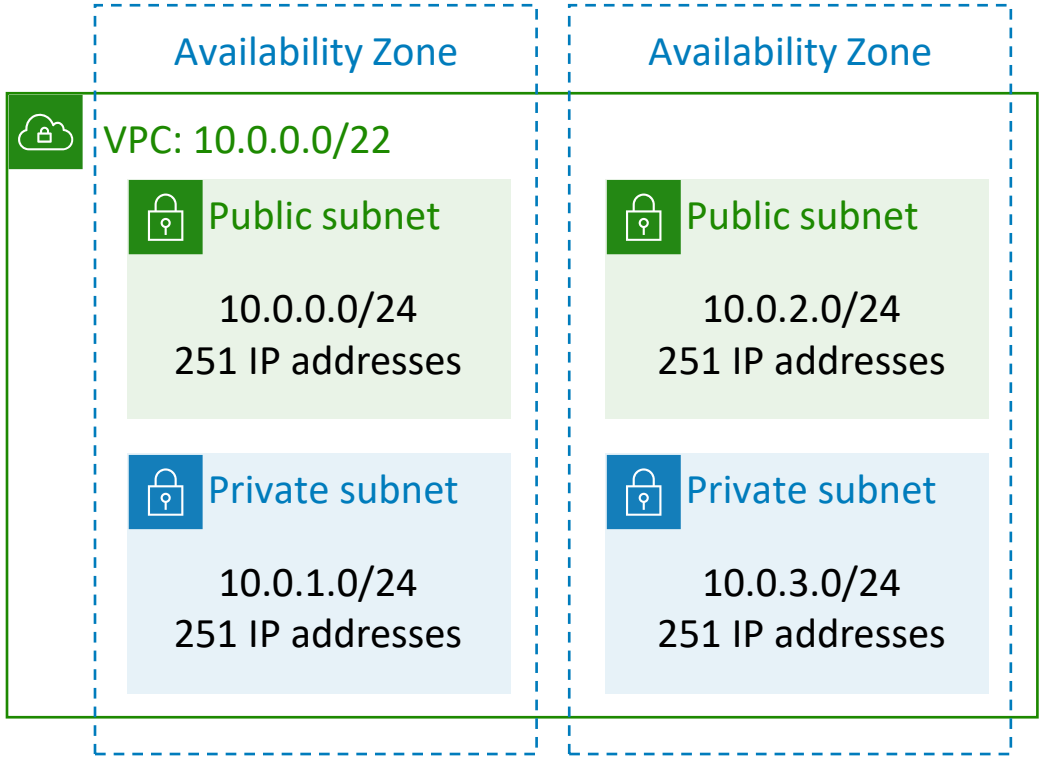
The CIDR block of a subnet can be:

- 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.

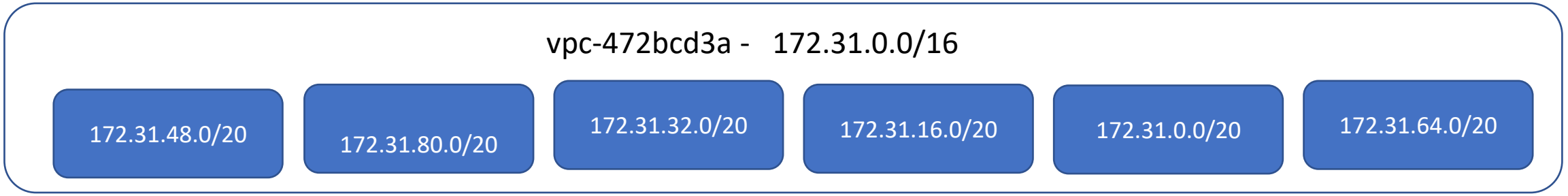
Subnets: Dividing your VPC

- 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



		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	0	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	0	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	0	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	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	172.31.64.0/20	1	0	1	0	1	1	0	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)

		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
AWS	x.x.x.x/16																																
AWS	x.x.x.x/28																																
IANA	x.x.x.x/8																																
IANA	x.x.x.x/12																																
IANA	x.x.x.x/16																																

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

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

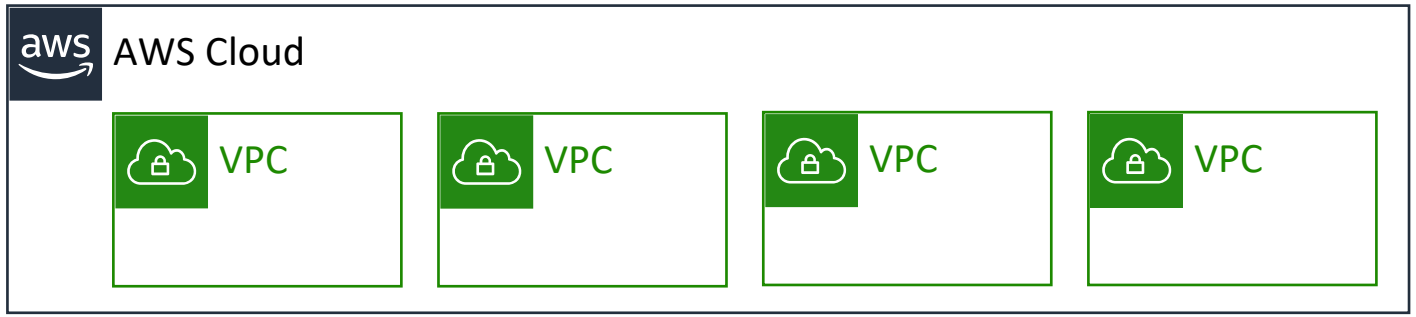
There are limited use cases where deploying one VPC might be appropriate:

- 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.

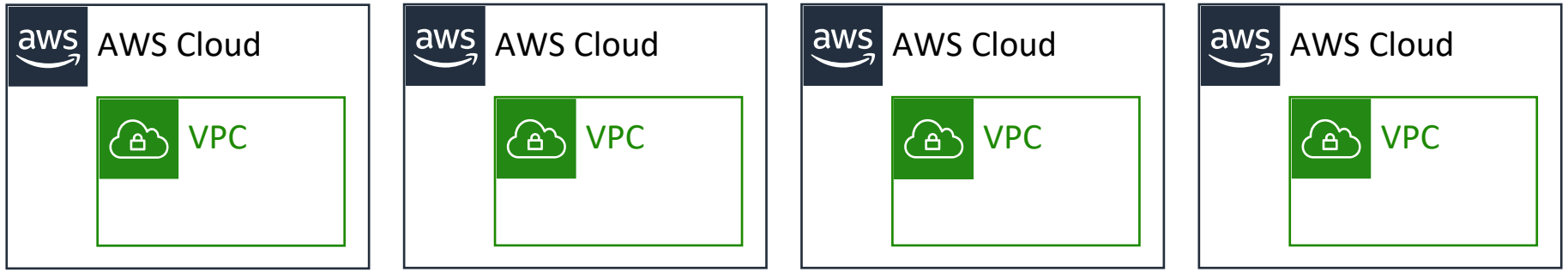
Multiple VPCs

- 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



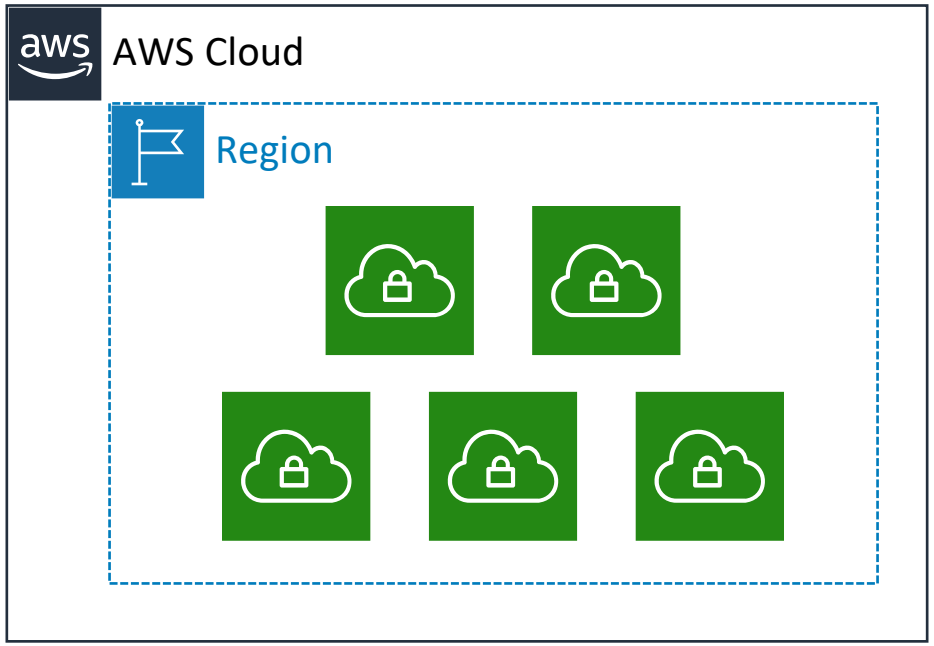
Multiple accounts

- 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 *



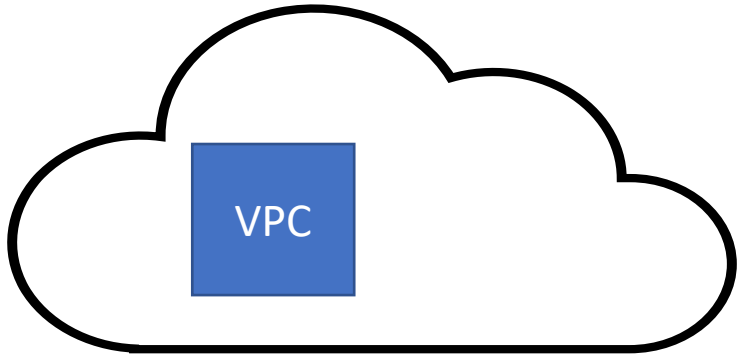
* The default quota is 5 VPCs per Region, but you can request a quota increase.

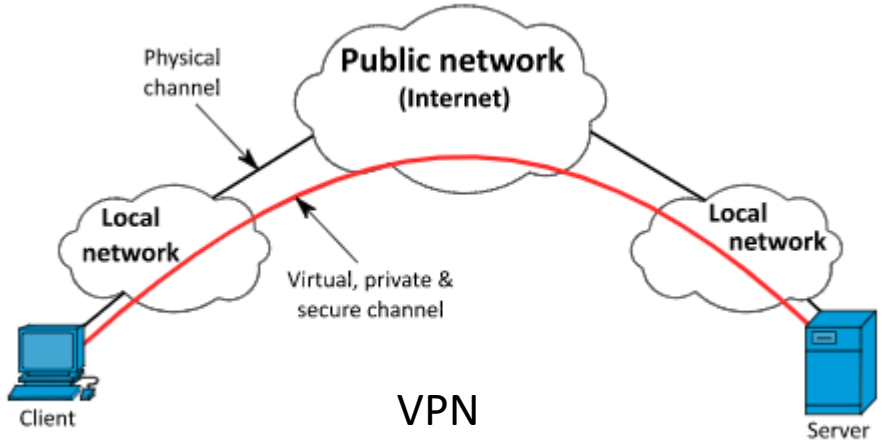
AWS Domain Name System

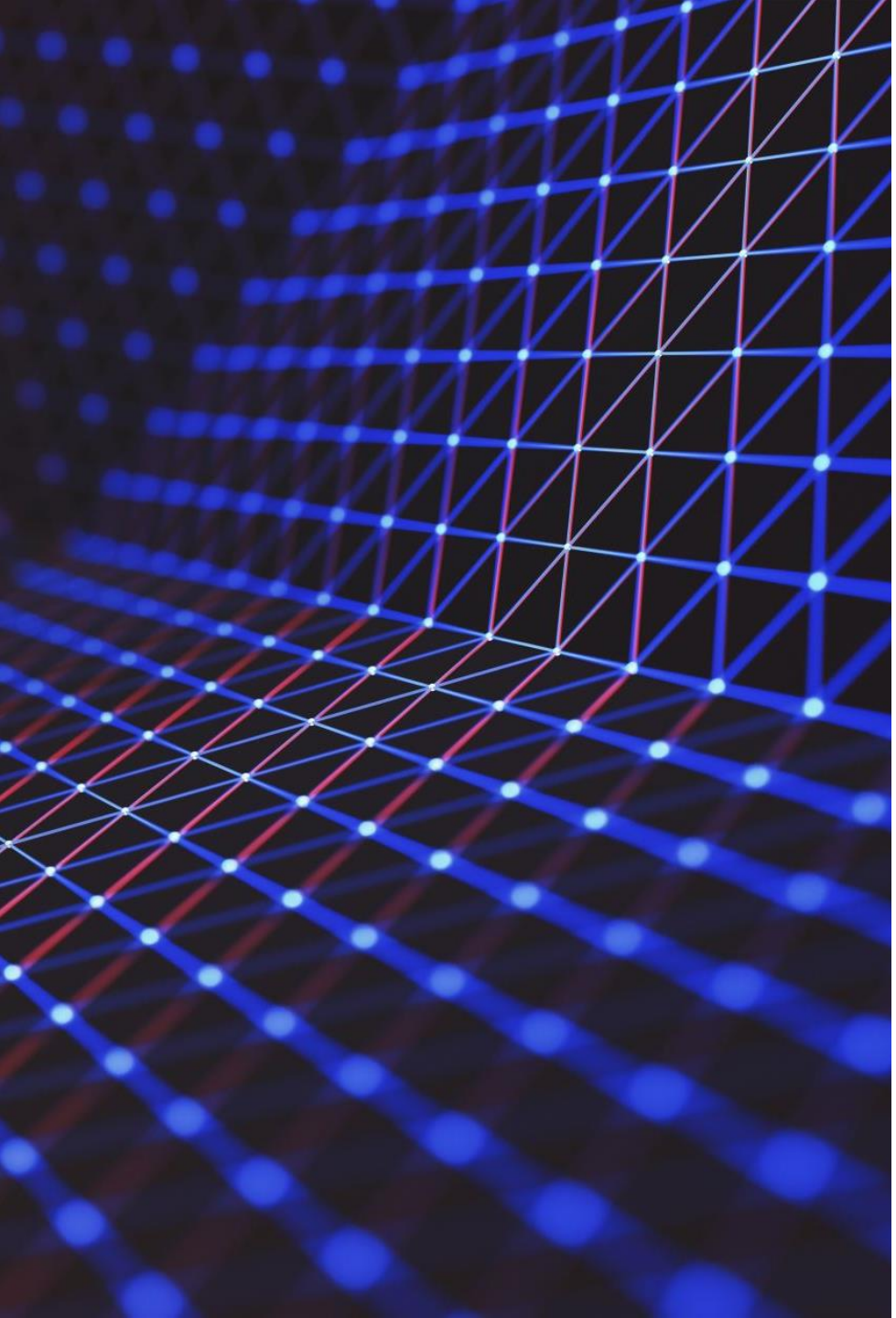
- An instance launched into a **default VPC** has
  - a public DNS hostname (corresponds to the public IPv4 address)
  - a private DNS hostname (corresponds to the private IPv4 address)
- An instance launched into a **custom VPC**:
  - a private DNS hostname
  - may have a public DNS hostname depending 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.

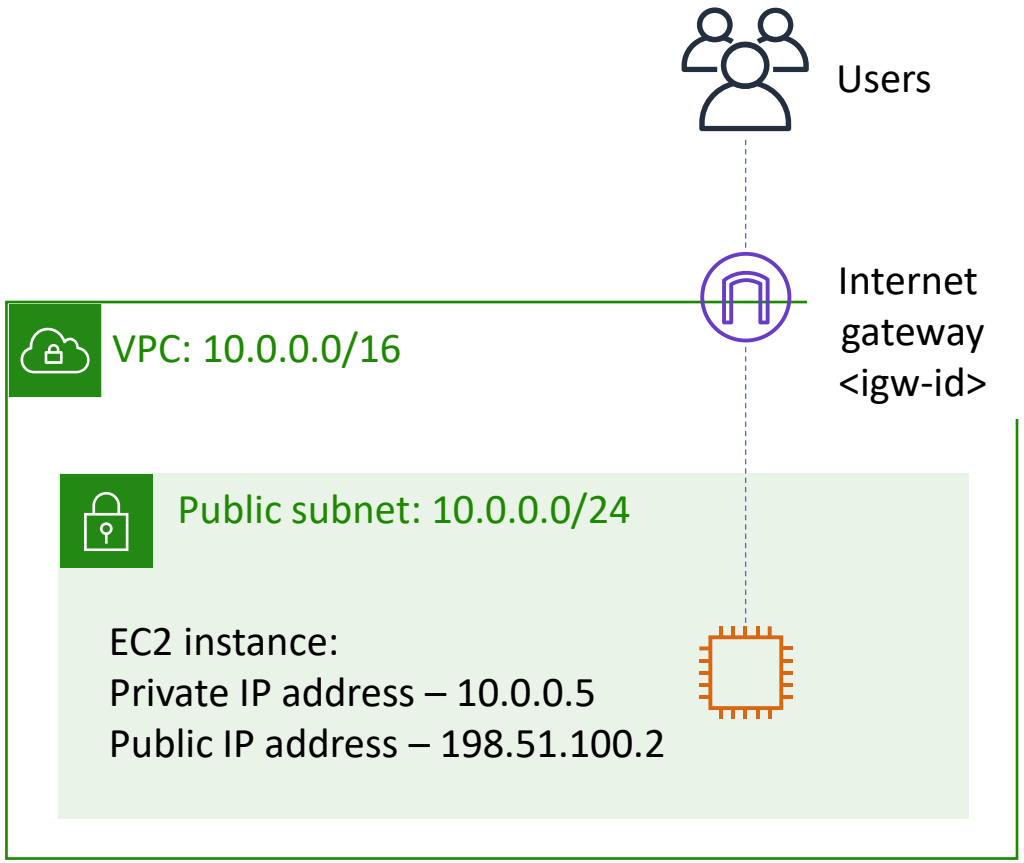
Follow best practices when you design your VPC.

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

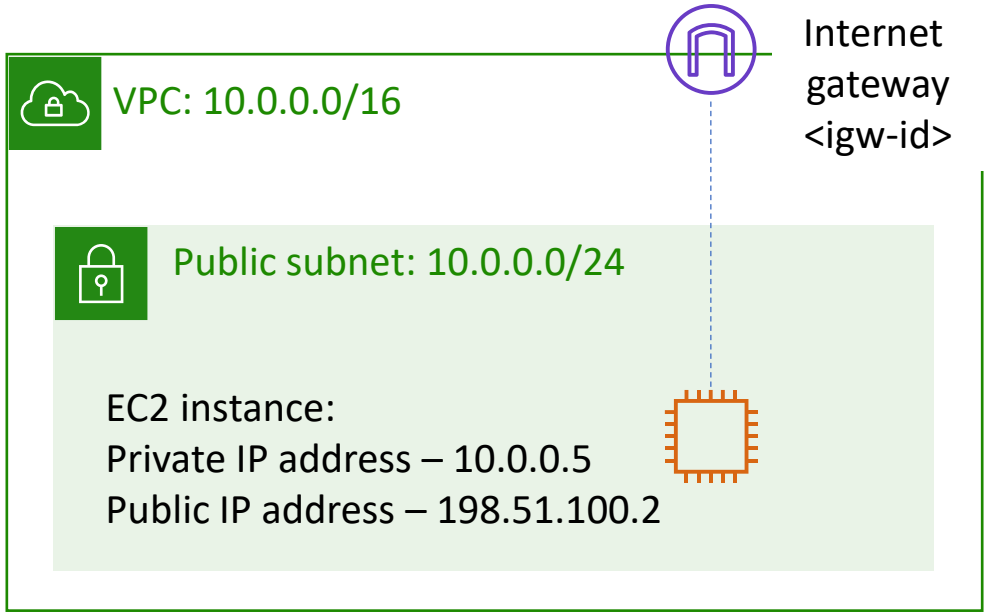


Directing traffic between VPC resources

- 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

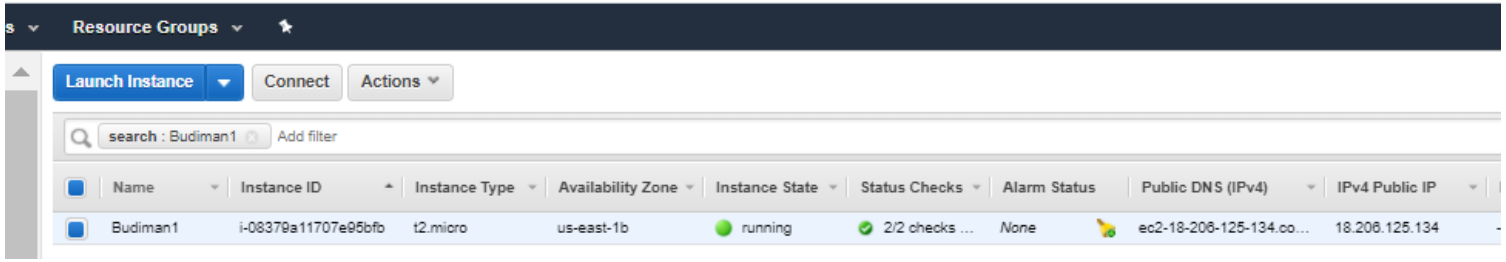
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

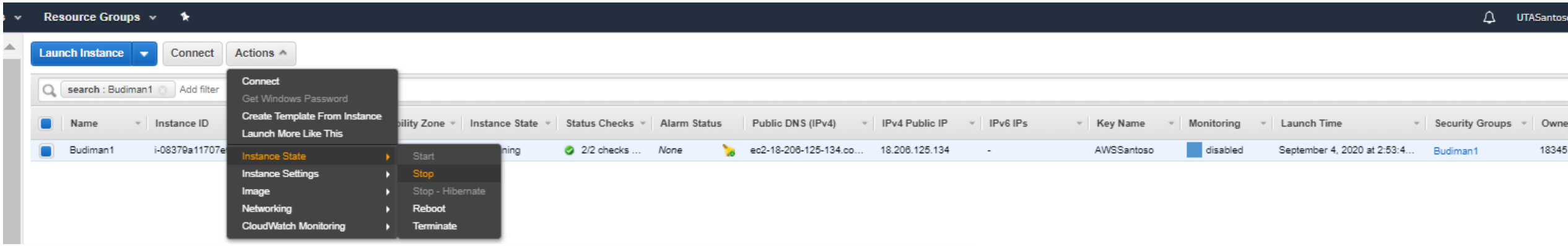
Elastic IP Address & Elastic Network Interface

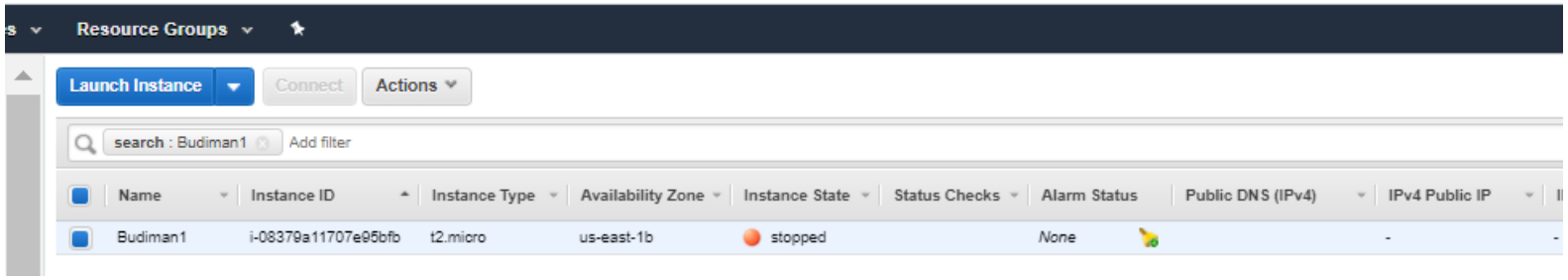
Elastic IP Address (EIP)

- 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

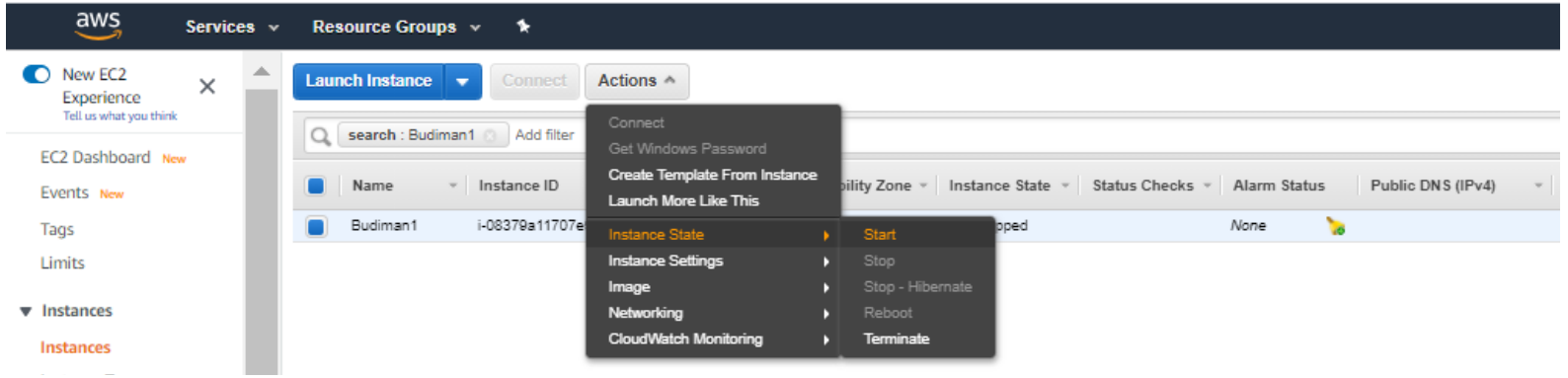
Compare the Public IP addresses before and after the “stop” - Before

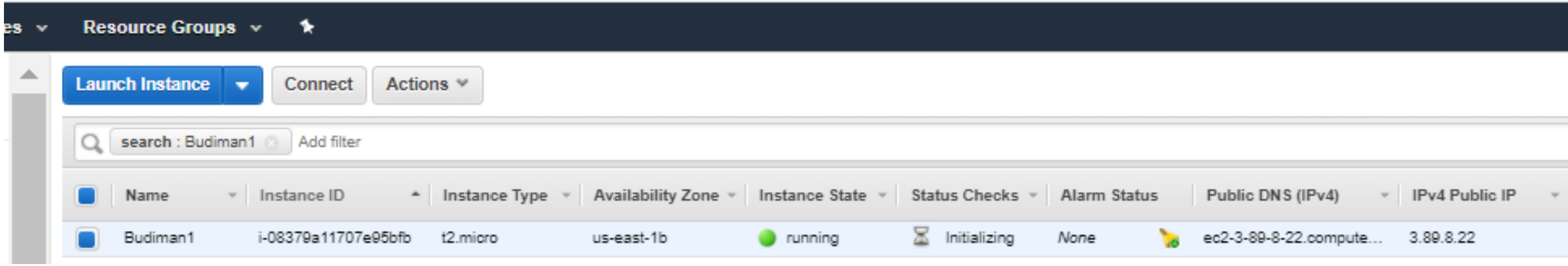






Compare the Public IP addresses before and after the “stop” - After





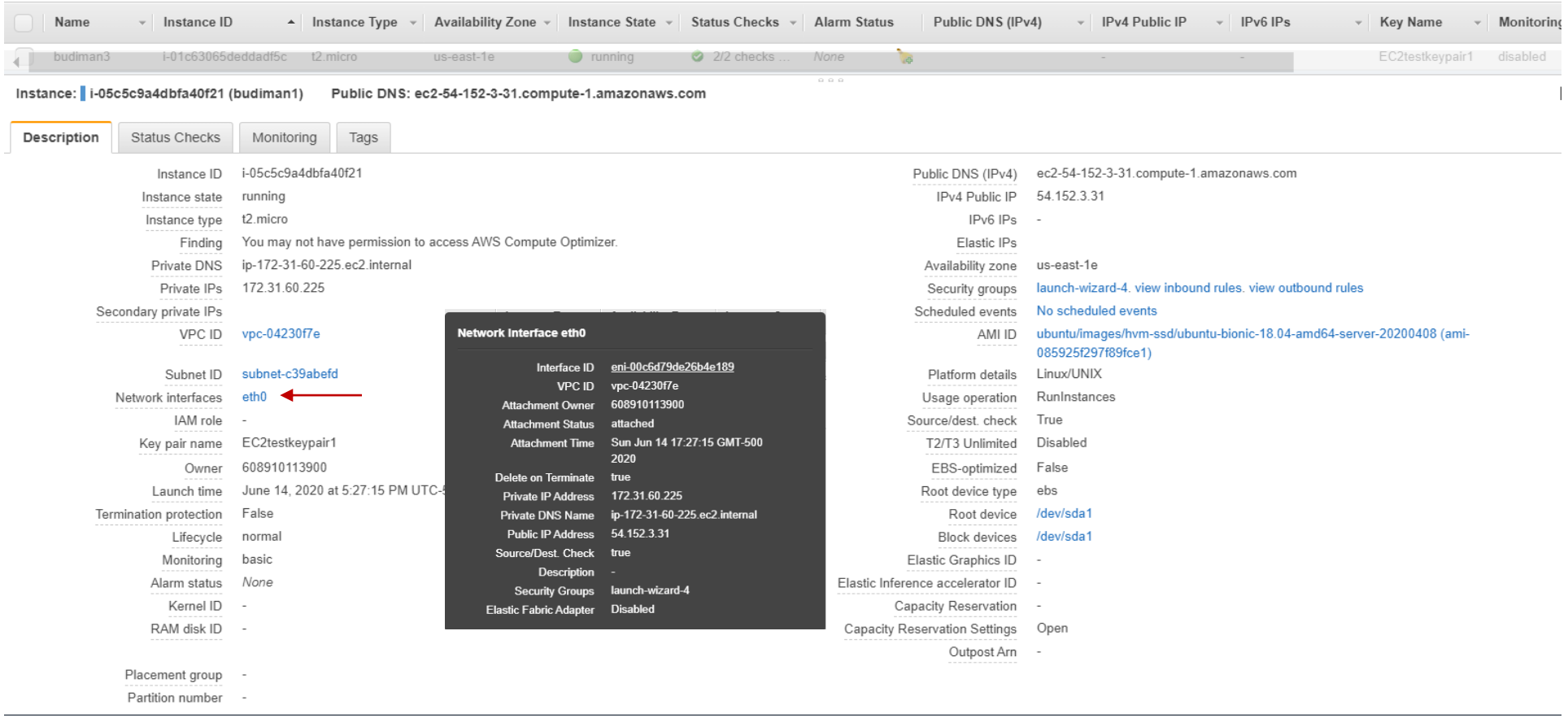
Elastic Network Interface

- 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 of more security groups
  - A source/destination check flag and description
- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html

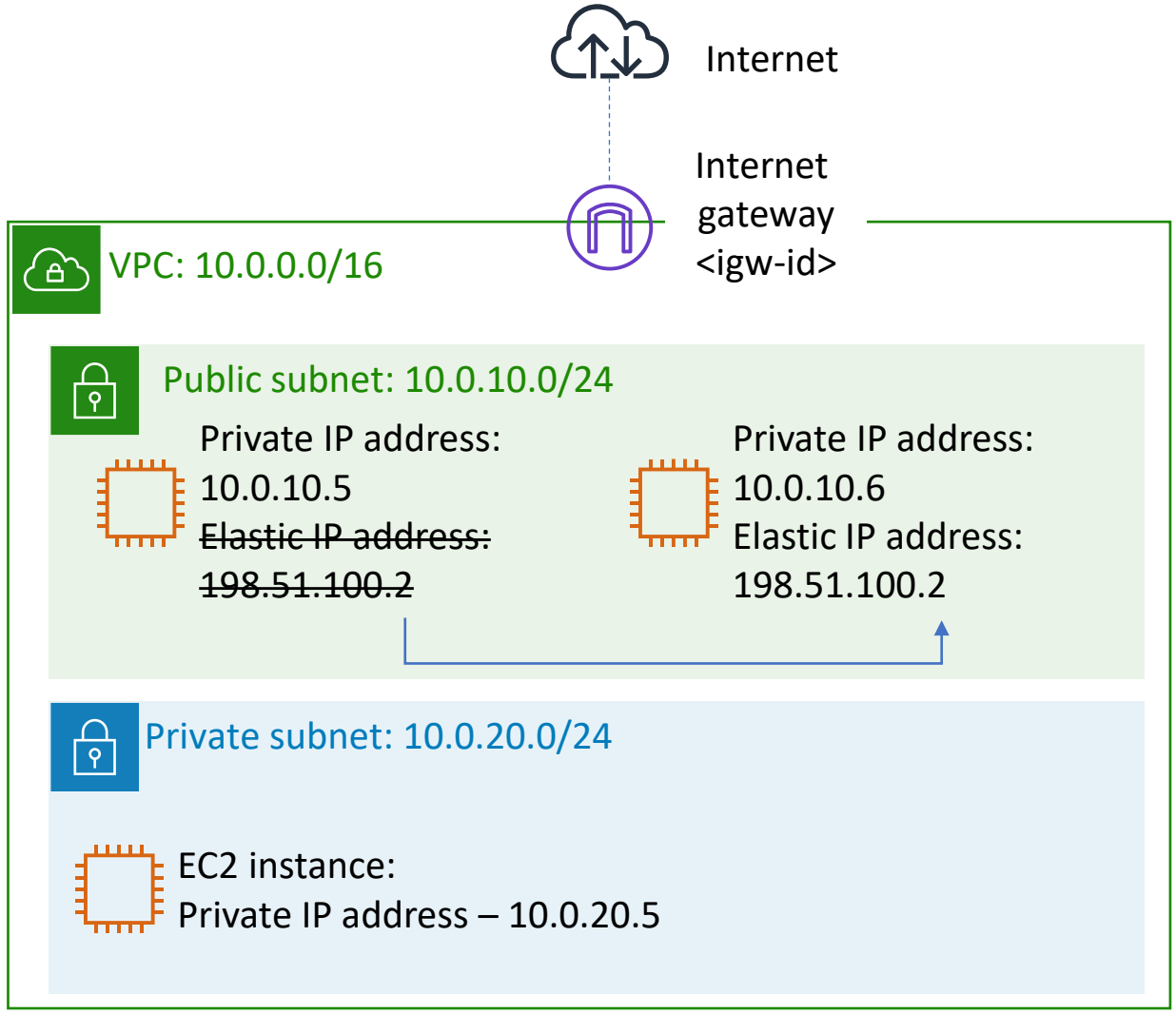
EC2-Network Interface



Remapping an IP address from one instance to another

Elastic IP addresses

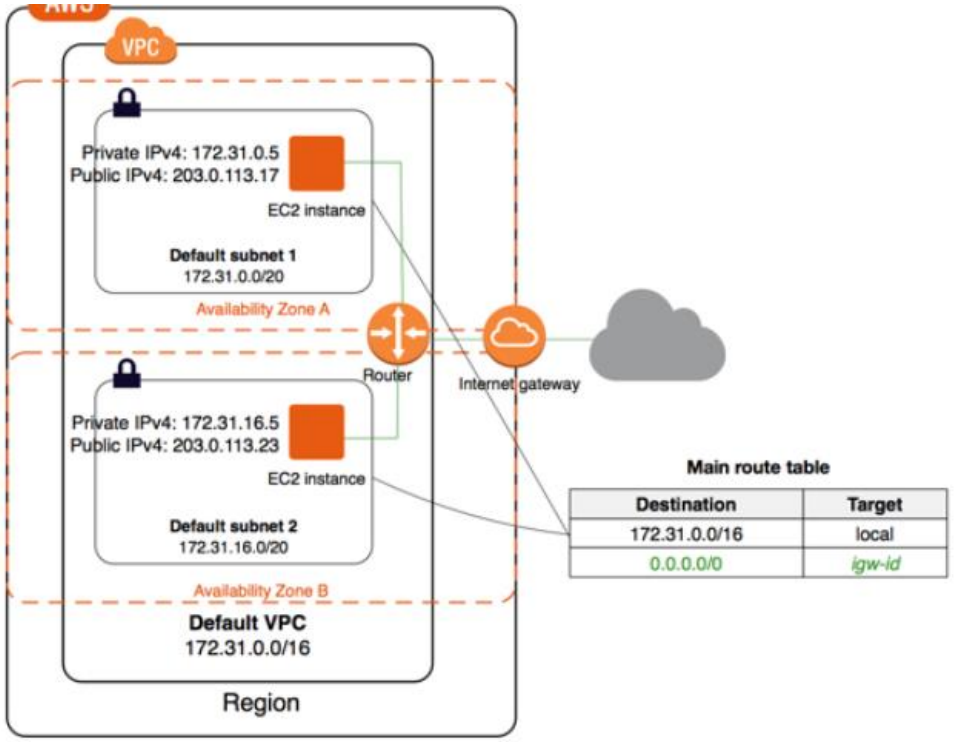
- Are static, public IPv4 addresses associated with your AWS account
- Can be associated with an instance or elastic network interface
- Can be remapped to another instance in your account
- Are useful for redundancy when load balancers are not an option



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

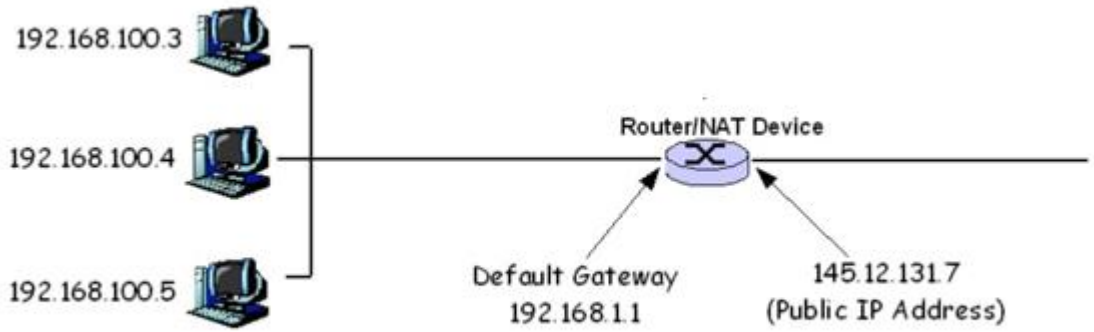
NAT (Network Address Translation)

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

NAT TYPES	DESCRIPTION	FOR	
Static NAT (SNAT)	**1 external** public address is mapped to **1 internal** 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 **NAT** pool.	Outgoing	Difficult external attack, keep changing public IP address.
Port Address Translation (PAT)	Use combination of ip address and non-reserved TCP/IP ports. **One public IP address – many internal hosts**.	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 translate 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:

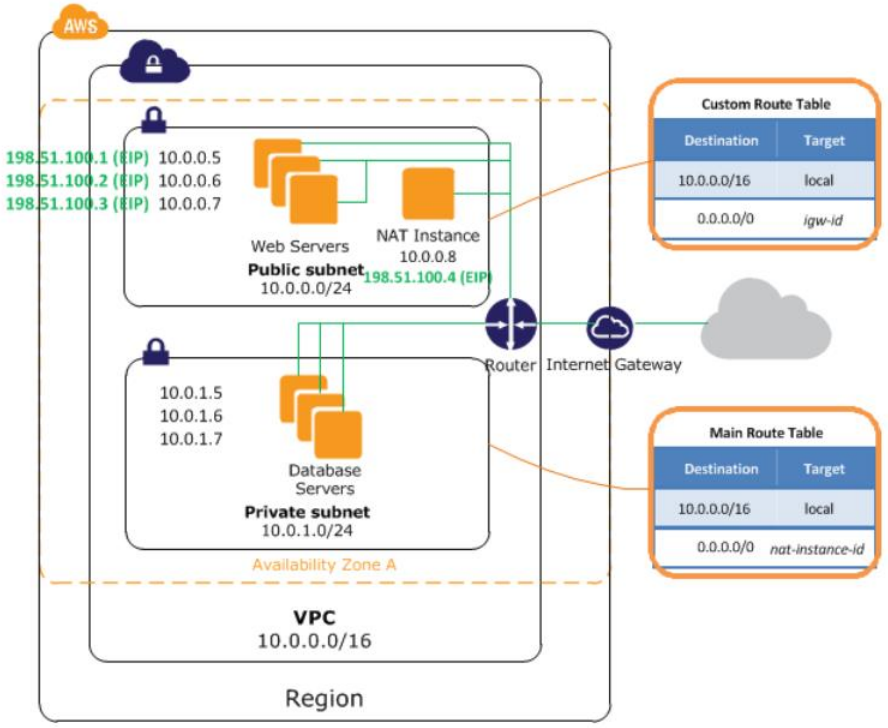
- A NAT device forwards traffic from instances in the private subnet to the Internet (**Outbound**) and then sends response to the instances – (similar to router)
    - No inbound
    - Purpose: to protect instances in Private subnet from the internet while allowing the instances to access it.
- When 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.

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

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.

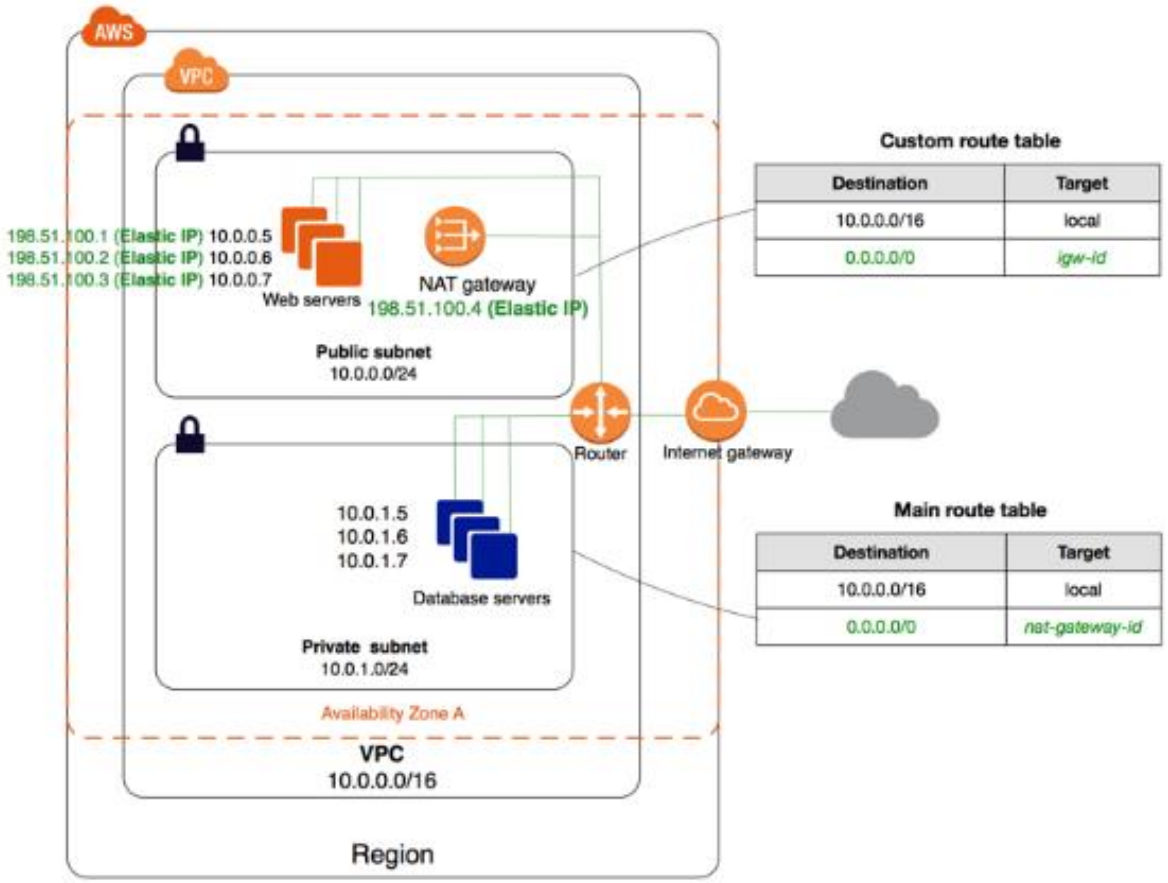


NAT Gateway

- 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

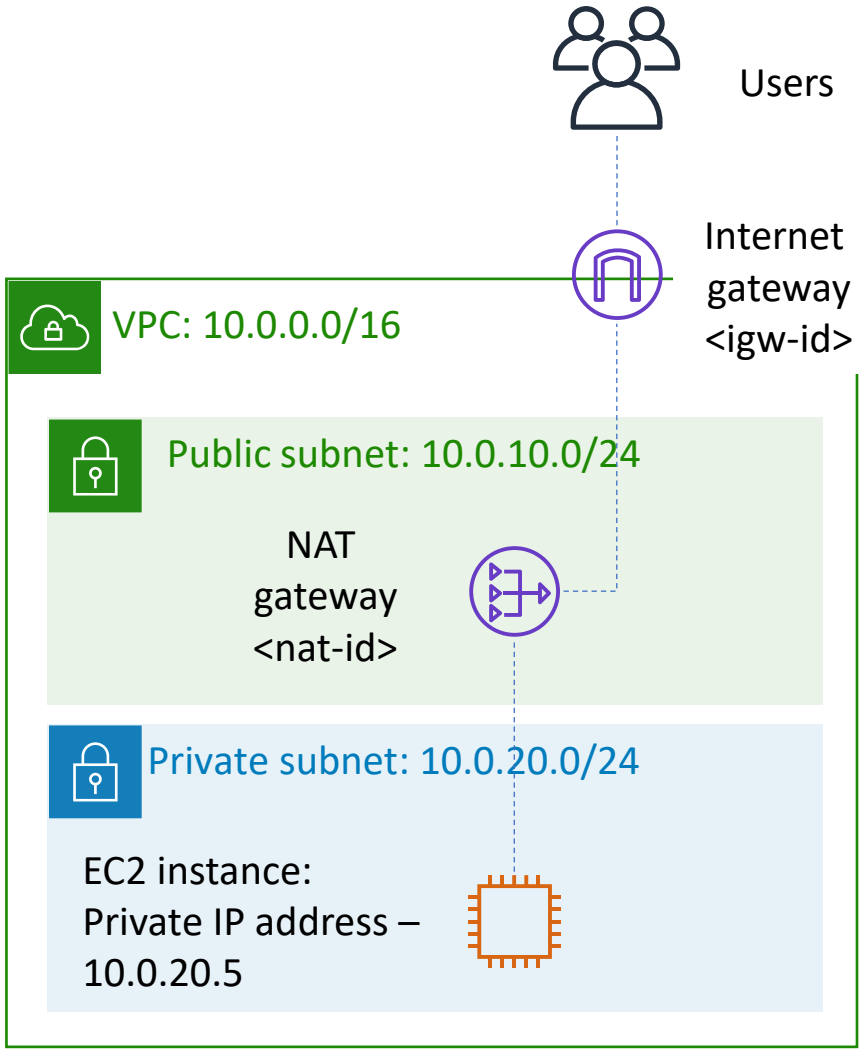
- Enable instances in a private subnet to initiate outbound traffic to the internet or other AWS services
- **Prevent** private instances from receiving **inbound** connection requests from the internet

Public route table

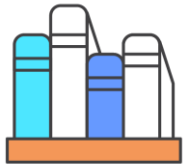
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Private route table

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



Subnet use case examples

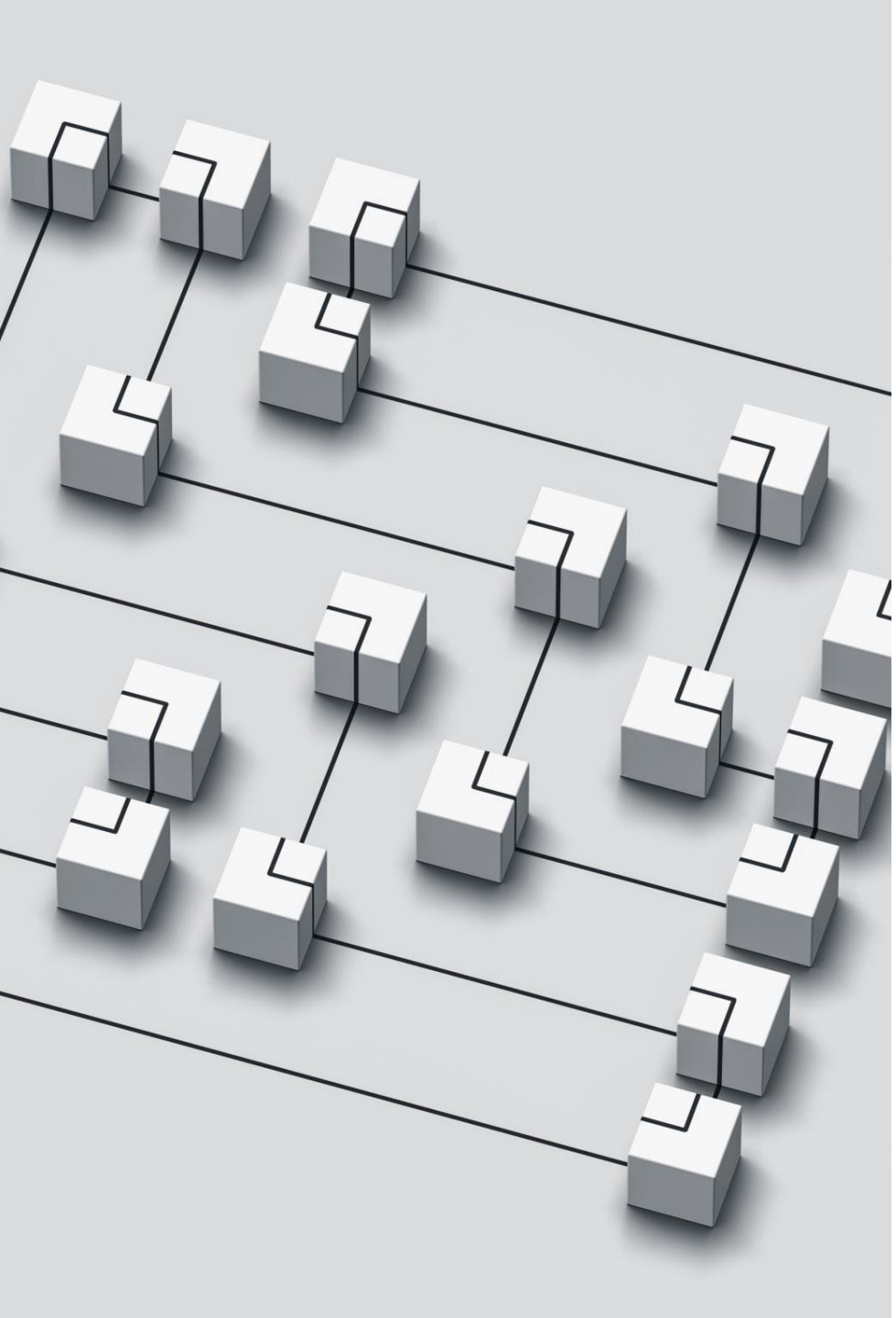


Data store instances → Private subnet

Batch-processing instances → Private subnet

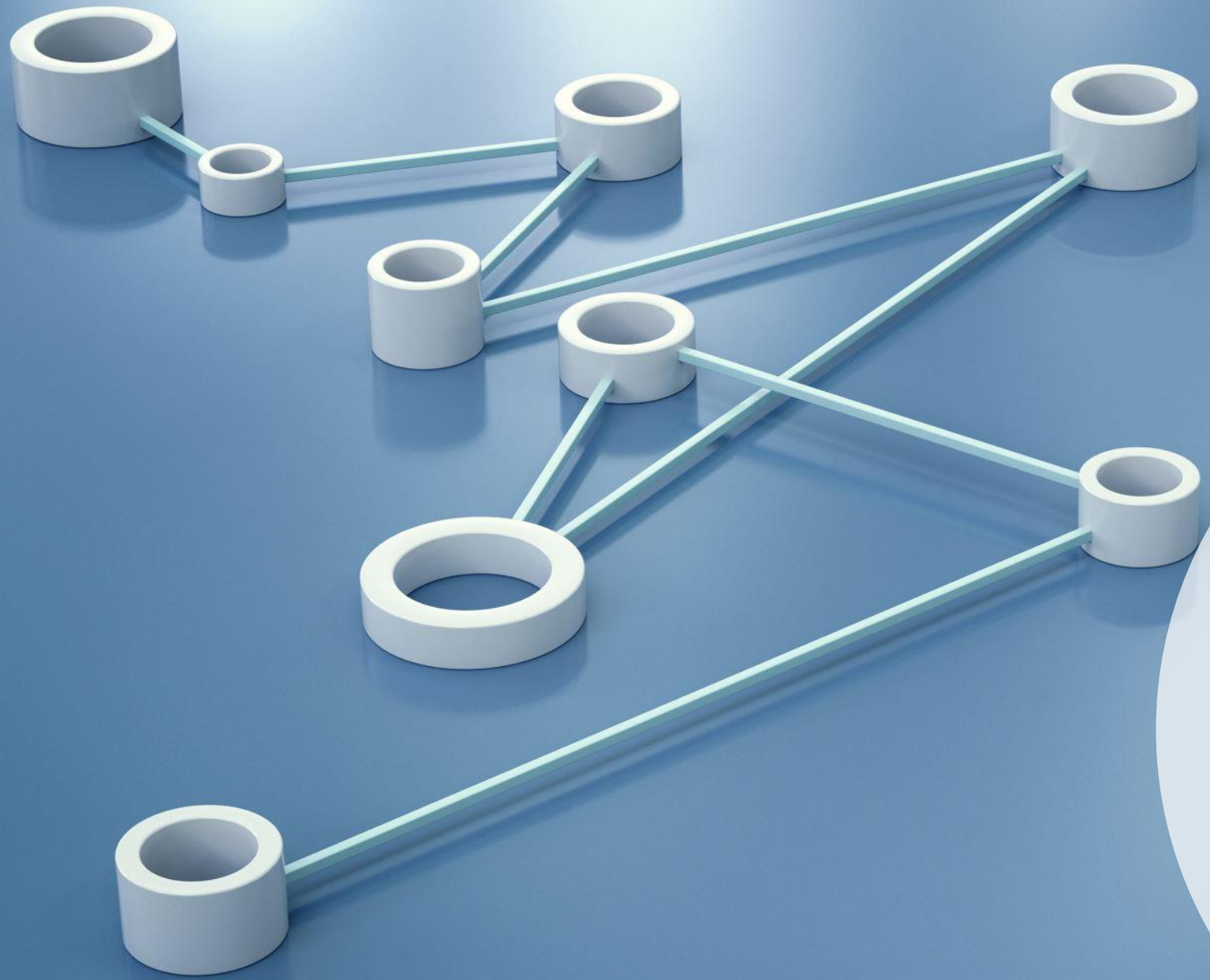
Backend instances → Private subnet

Web application instances → Public or private subnet



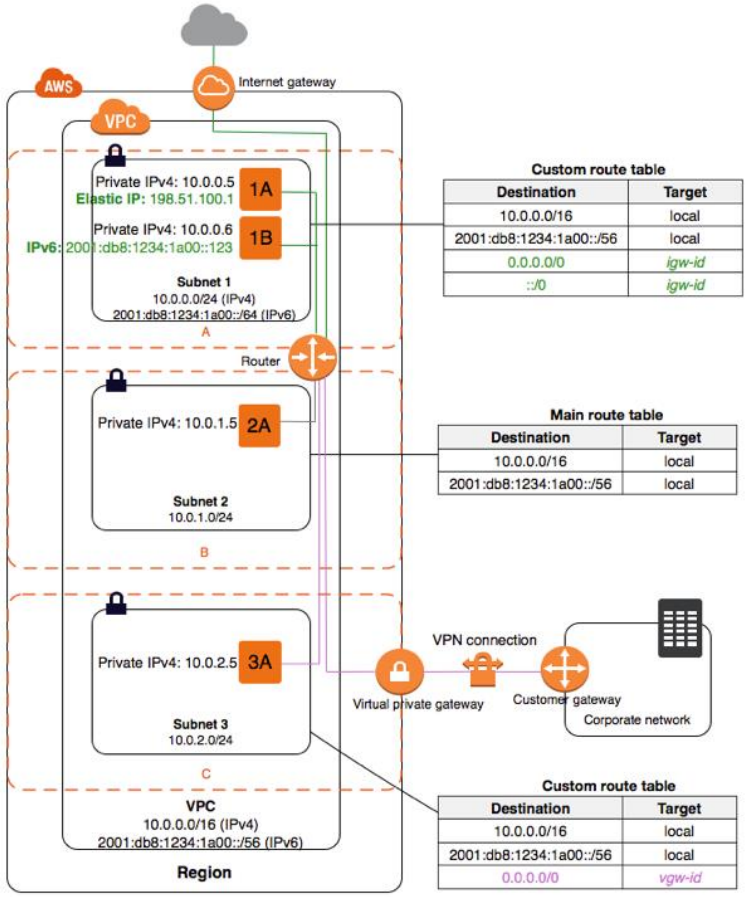
key takeaways

- 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

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-aabbccddee1122334

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

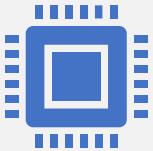
Network Security

Network Security

Security Groups

Firewall

Per Instance



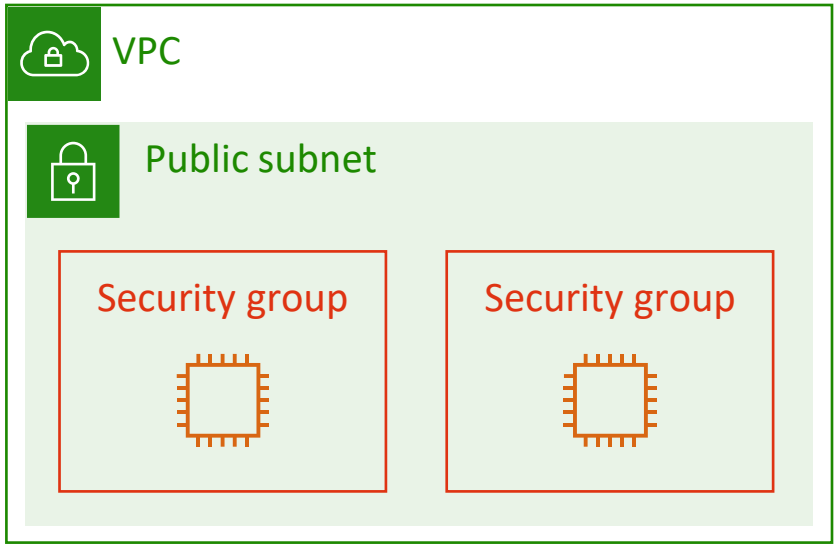
NACL

Network Access Control List

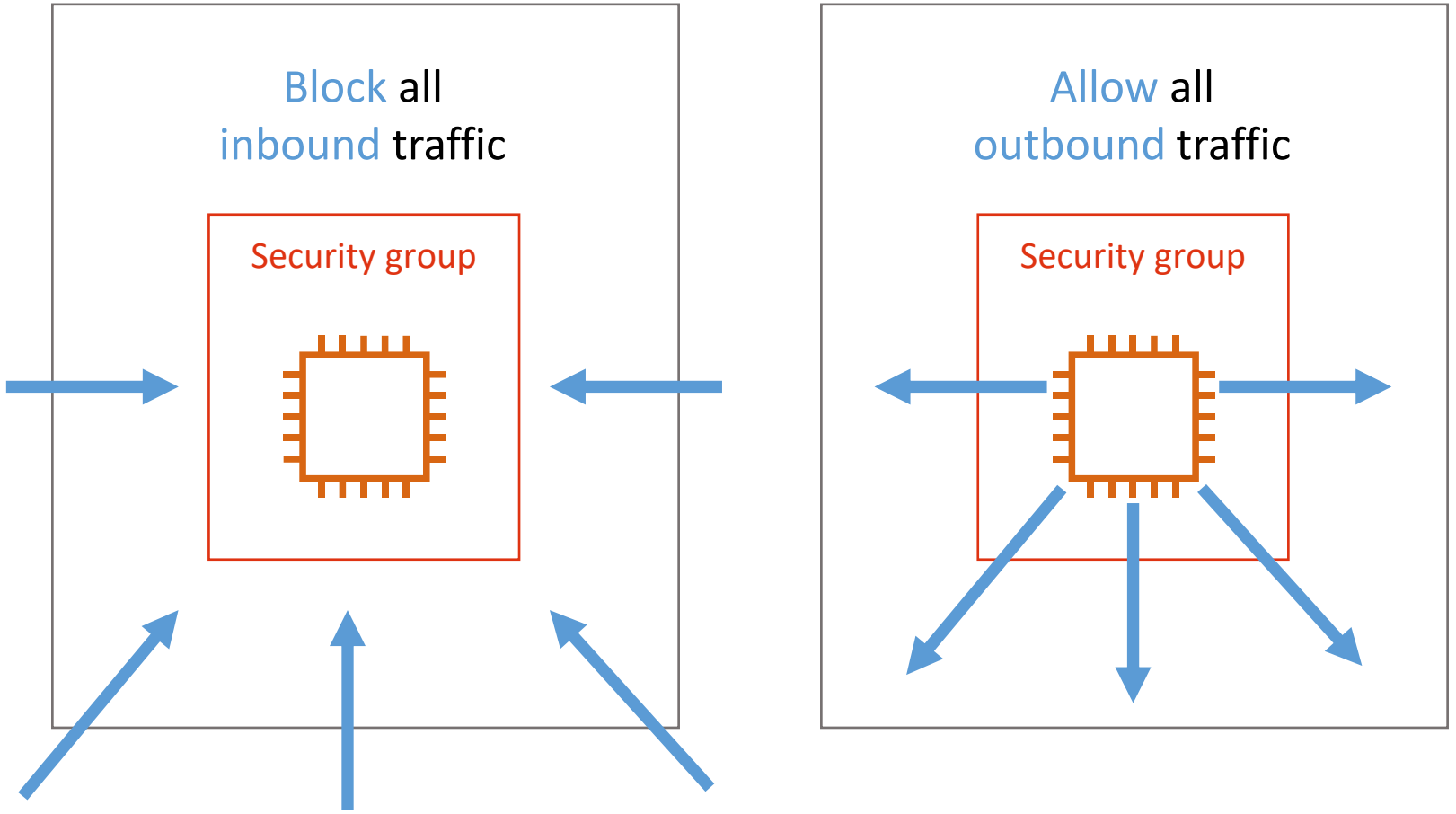
Subnet level

Security groups

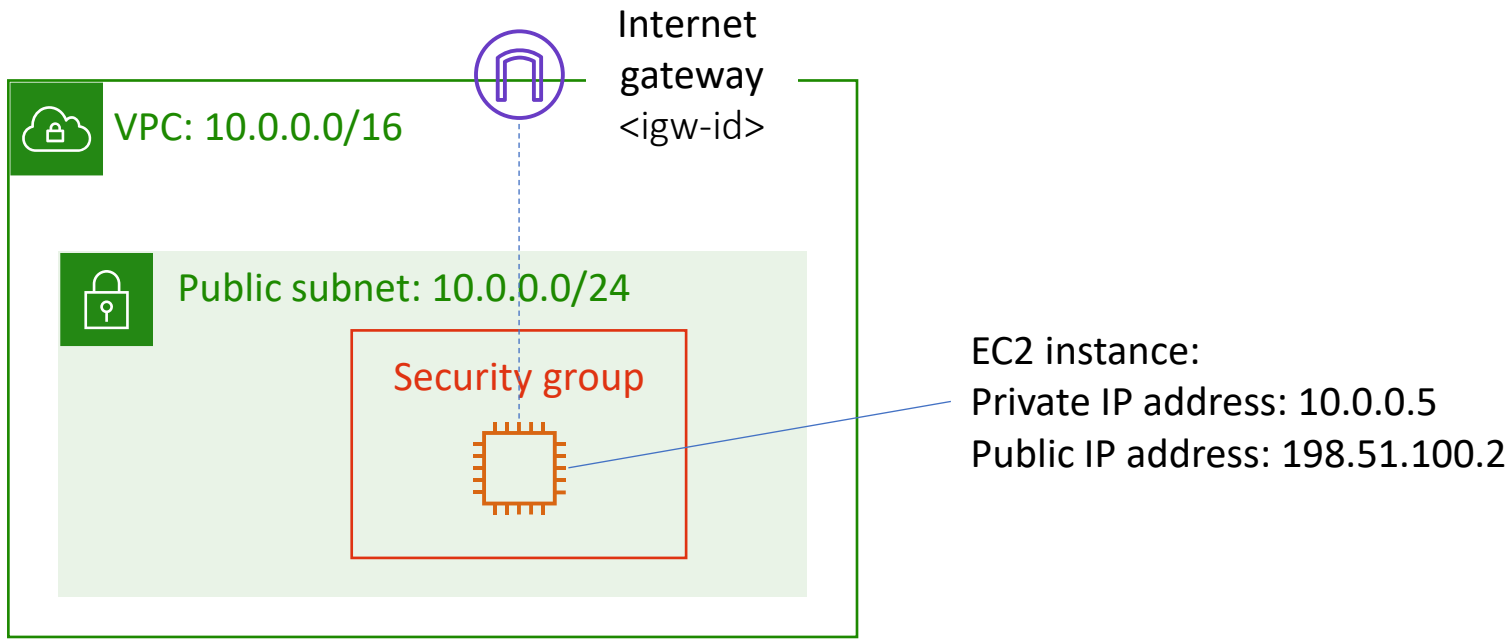
- Are stateful firewalls that control inbound and outbound traffic to AWS resources
- Act at the level of the instance or network interface



Default for new custom security groups



Custom security groups



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

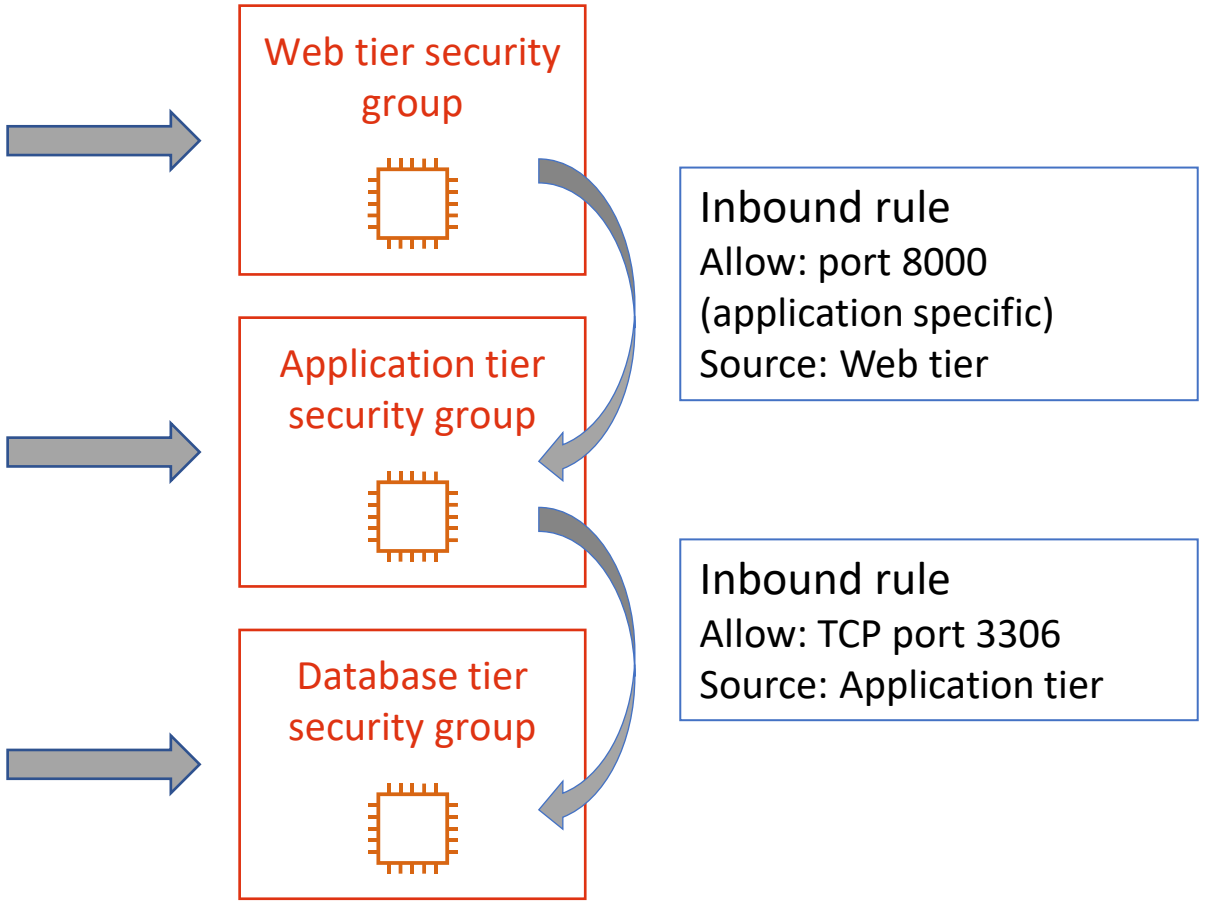
Chaining security groups

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

Inbound rule
Allow: SSH (port 22) to Application tier
Source: Corporate IP range

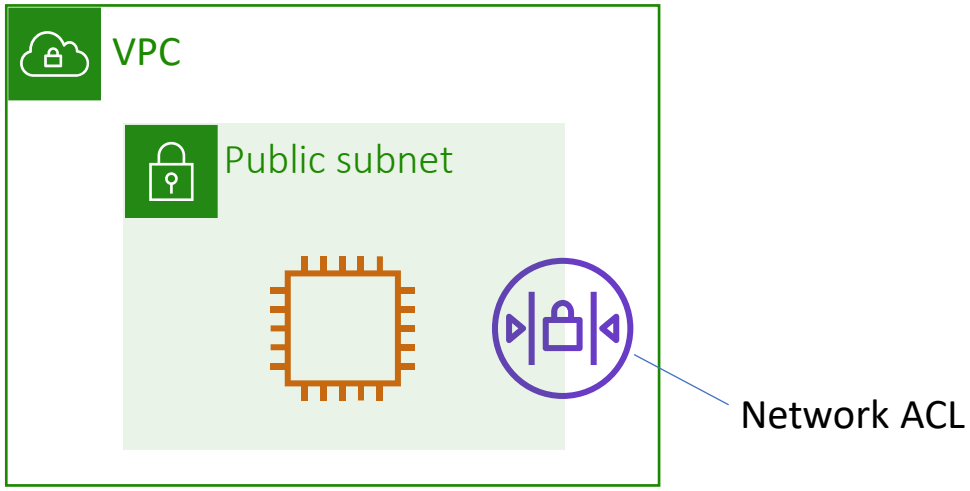
Inbound rule
Allow: SSH (port 22) to Database tier
Source: Corporate IP range

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

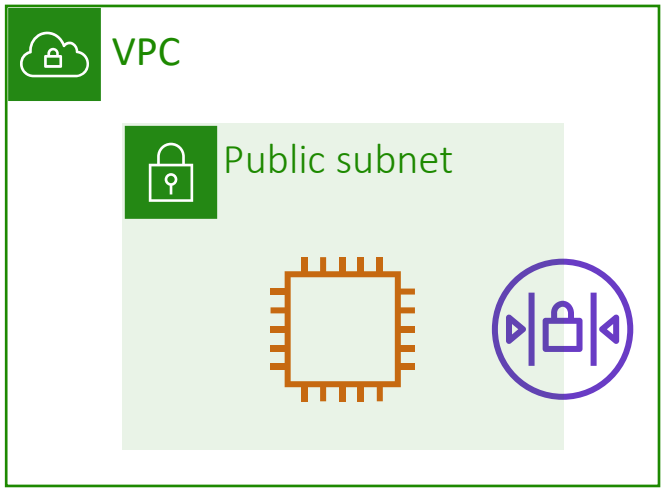


Network Access Control List

- 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

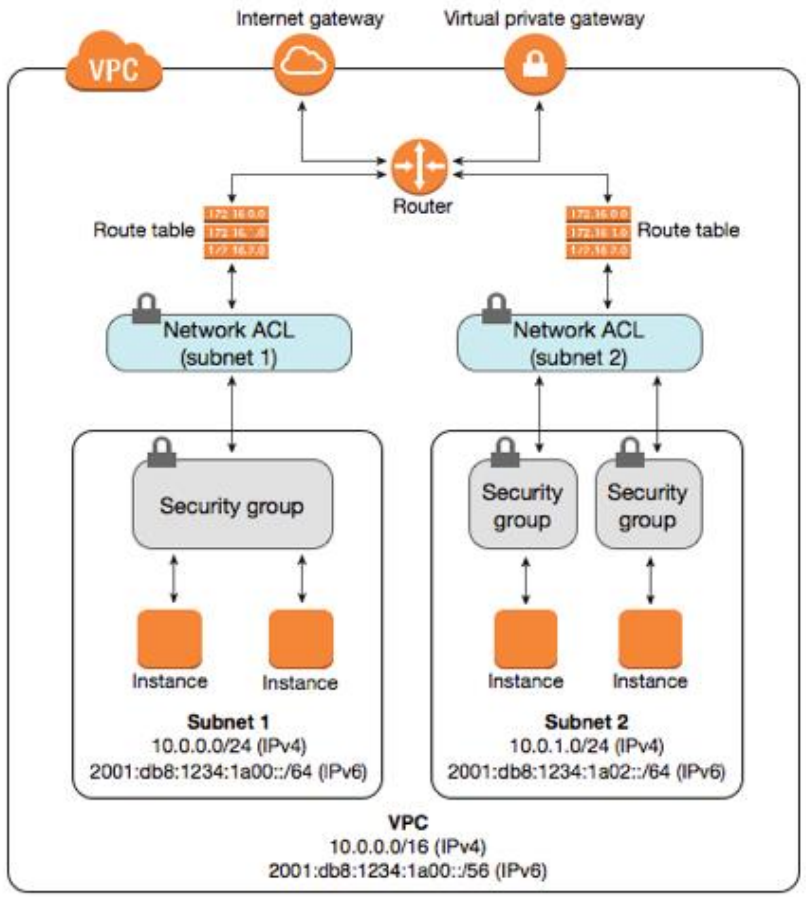
The default network ACL is configured to allow all traffic to flow in and out of the subnets with which it is associated.

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

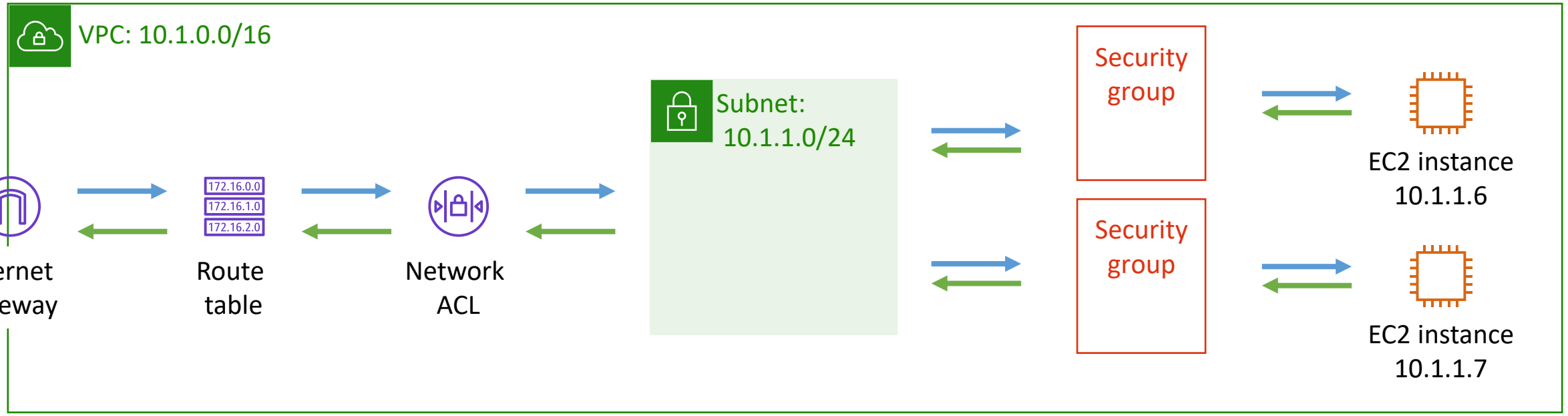
Inbound					
Rule #	**Type**	**Protocol**	**Port Range**	**Source**	**Allow/Deny**
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0	DENY
Outbound					
Rule #	**Type**	**Protocol**	**Port Range**	**Destination**	**Allow/Deny**
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW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

Security Group vs. NACL



Security Group	NACL
Applied only at **instance level**	**Subnet level**
Stateful (return traffic is allowed by default)	Stateless (return traffic is not allowed by default)
Allow rules only; can't specify a deny rule explicitly	**Allow** and **Deny** 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)

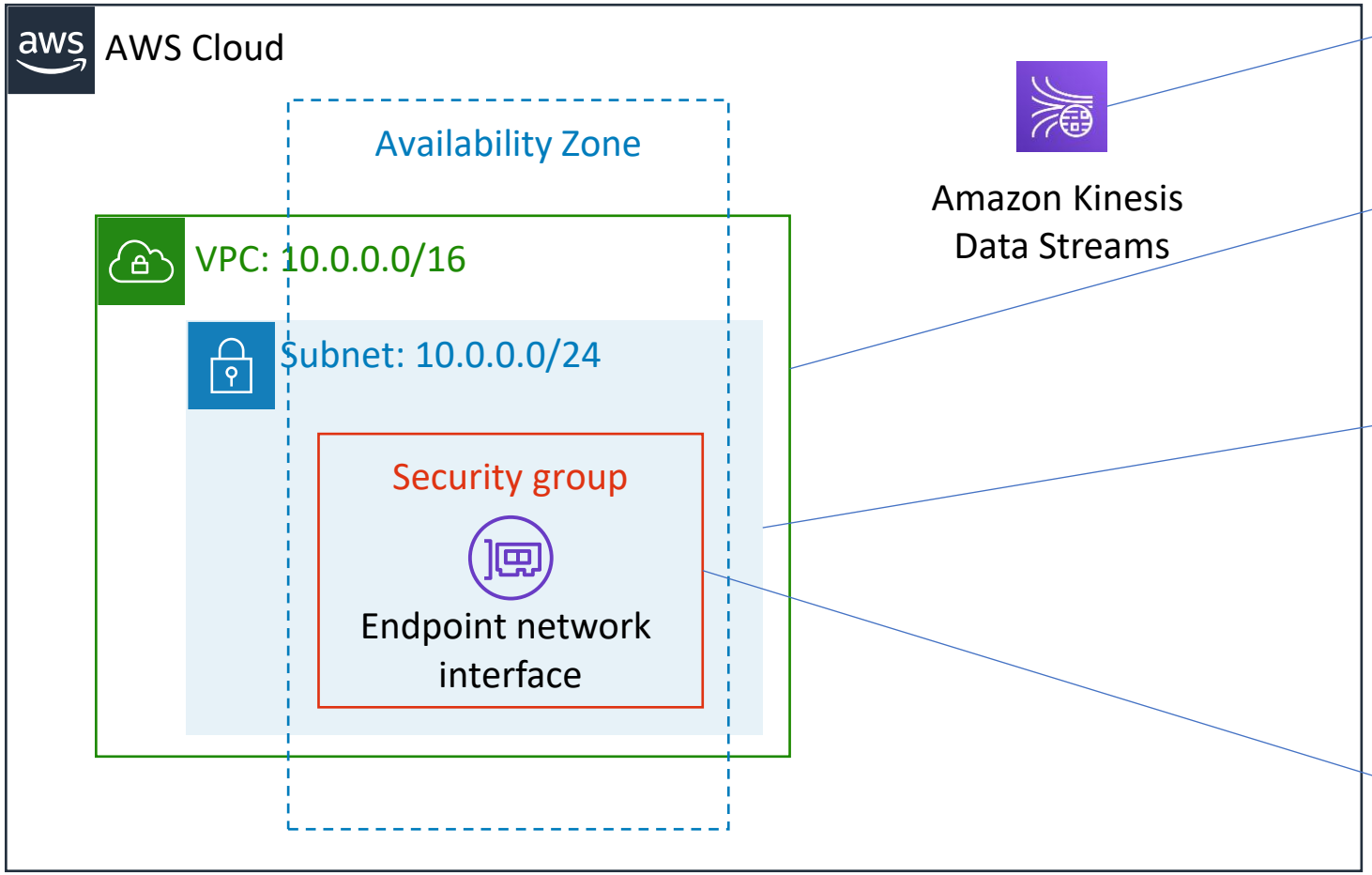
VPC endpoints

- 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.

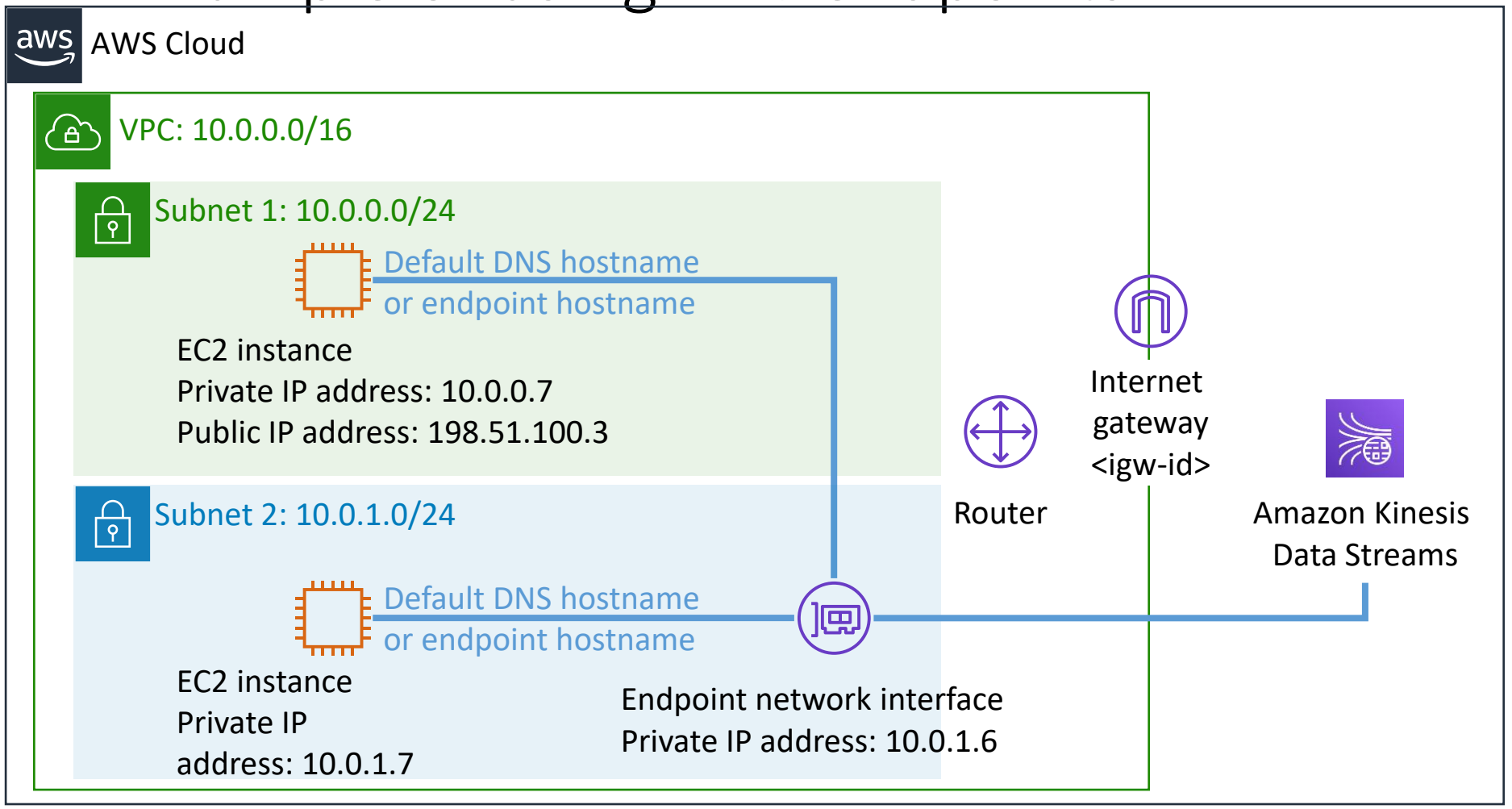
2. Choose the VPC where you want to create the interface endpoint.

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



Subnet 1 route table

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

Internet

Subnet 2 route table

Destination	Target
10.0.0.0/16	local

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 79-10, July 1979; also in IEEE Transactions

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

- One Region

- A subnet in an AZ (select or no preference)
- An AZ may have multiple subnets
- Can set public/private subnets (no public IP for EC2s connected)
- Set auto-assign public IP address for public subnet (“Default” network is set).

- Create an Internet Gateway
- Attach to the VPC.

- Target Internet Gateway
- Associate with public subnets

- Connect to subnet
- Define Security Groups

AWS VPC Quotas

VPC and subnets

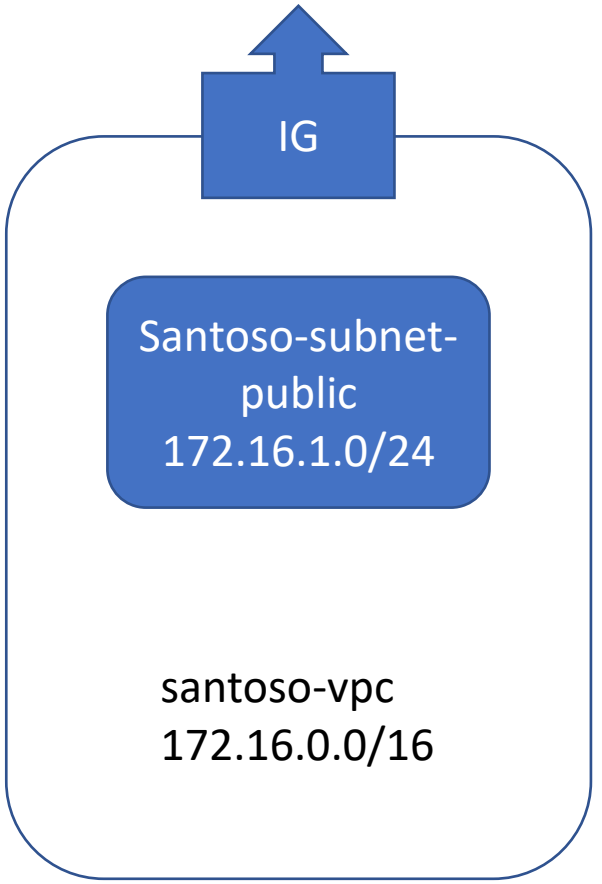
Resource	Default	Comments
VPCs per Region	5	The quota for internet gateways per Region is directly correlated to this one. Increasing this quota increases the quota on internet gateways per Region by the same amount. You can have 100s of VPCs per Region for your needs even though the default quota is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

https://docs.amazonaws.cn/en_us/vpc/latest/userguide/amazon-vpc-limits.html

VPC Demo

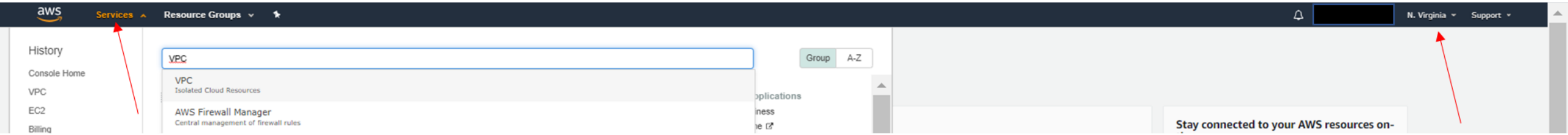
- VPC Name: santoso-vpc
- CIDR: 172.16.0.0/16
- Subnet 1: santoso-subnet-public
- Subnet 1 CIDR: 172.16.1.0/24

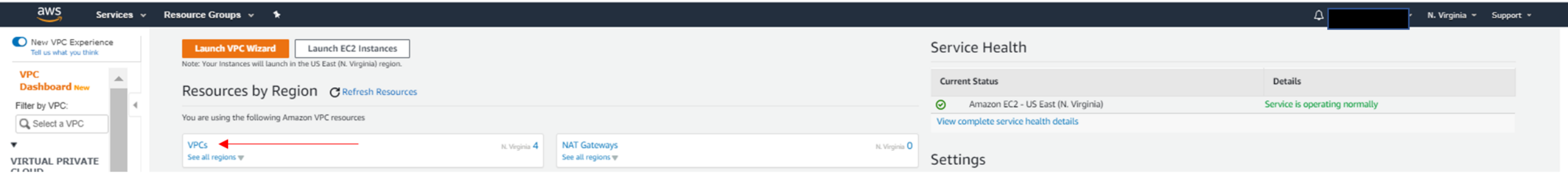
AWS allows block sizes between **/16** and **/28**
RFC 1918 private IP addresses (IANA)
10.0. 0.0 - 10.255. 255.255 (10/8 prefix)
172.16. 0.0 - 172.31. 255.255 (172.16/12 prefix)
192.168. 0.0 - 192.168. 255.255 (192.168/16 prefix)

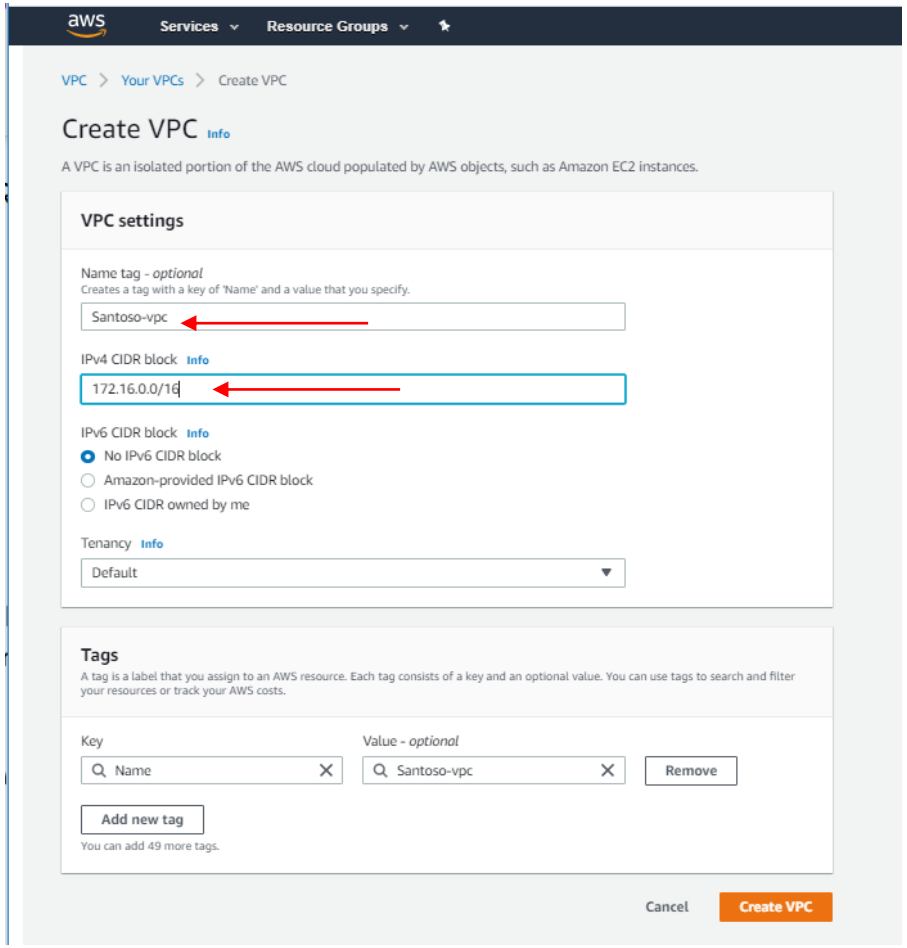


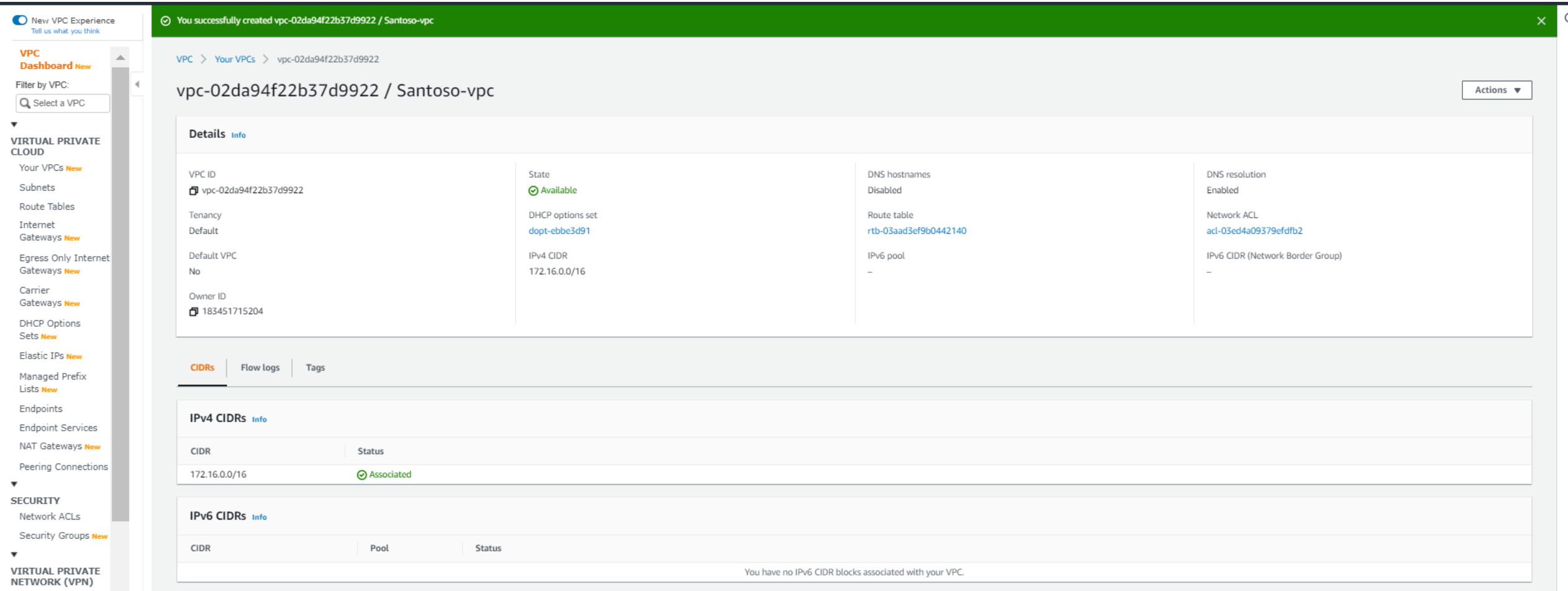
After you are done:
1. Terminate EC2s
2. Delete VPC

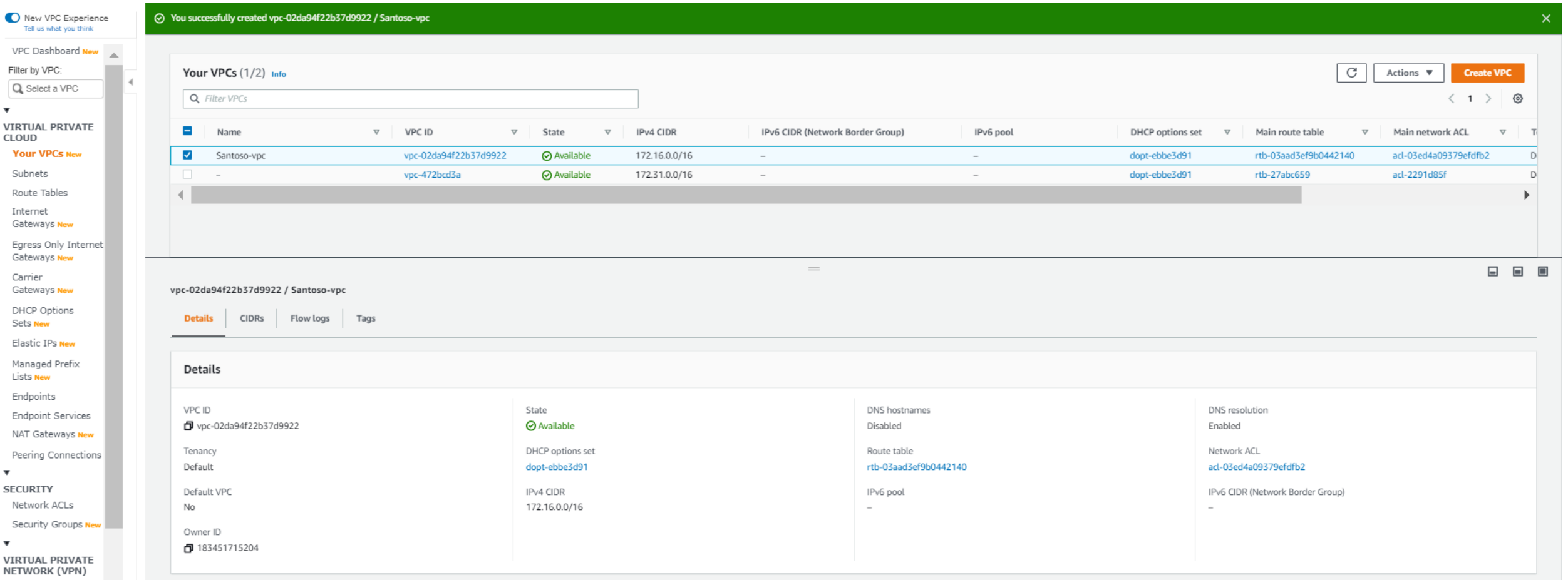
We need to create a VPC



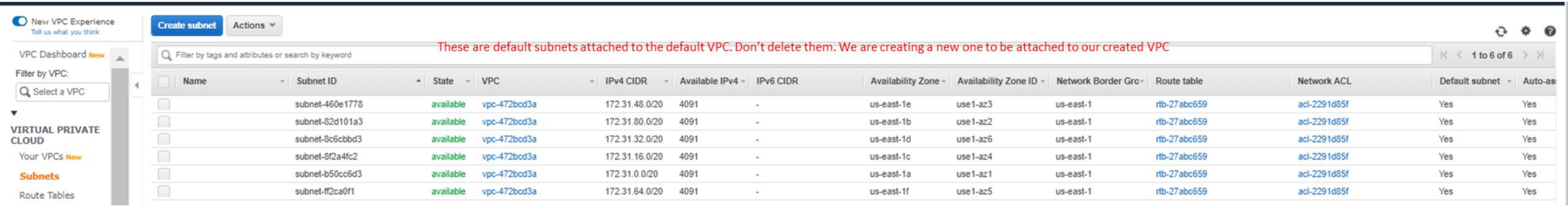


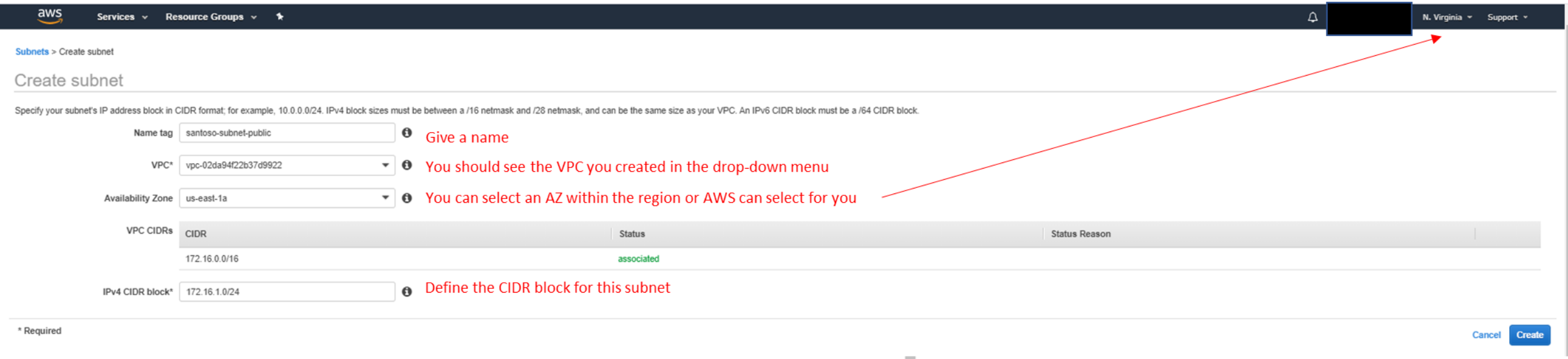

aws
Services
Resource Groups
VPC > Your VPCs > Create VPC
Create VPC Info
A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances.
VPC settings
Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.
Santoso-vpc
IPv4 CIDR block Info
172.16.0.0/16
IPv6 CIDR block Info
No IPv6 CIDR block
Amazon-provided IPv6 CIDR block
IPv6 CIDR owned by me
Tenancy Info
Default
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
Name
Santoso-vpc
Remove
Add new tag
You can add 49 more tags.
Cancel
Create 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
VPC > Your VPCs > vpc-02da94f22b37d9922
vpc-02da94f22b37d9922 / Santoso-vpc
Actions
Details Info
VPC ID
vpc-02da94f22b37d9922
Tenancy
Default
Default VPC
No
Owner ID
183451715204
State
Available
DHCP options set
dopt-ebbe3d91
IPv4 CIDR
172.16.0.0/16
DNS hostnames
Disabled
Route table
rtb-03aad3ef9b0442140
IPv6 pool
–
DNS resolution
Enabled
Network ACL
acl-03ed4a09379efdfb2
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.

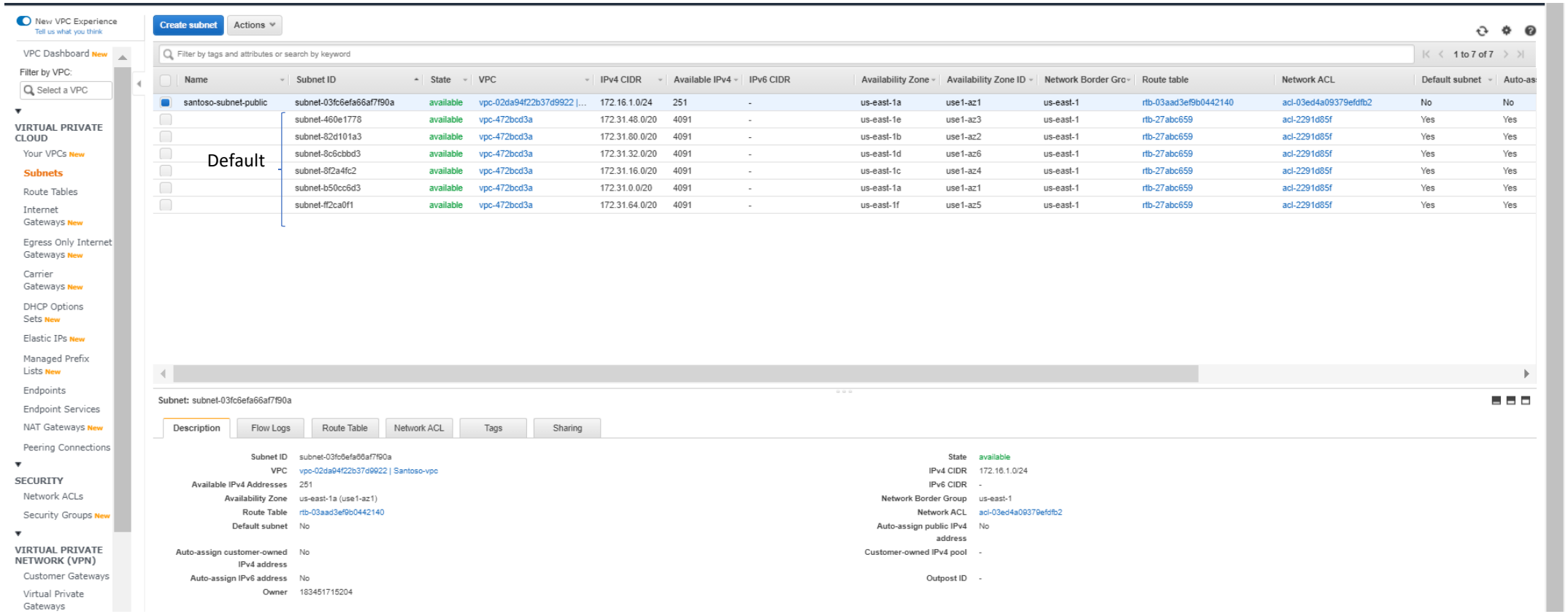

New VPC Experience
Tell us what you think
VPC Dashboard New
Filter by VPC:
Select a VPC
VIRTUAL PRIVATE CLOUD
Your VPCs New
Subnets
Route Tables
Internet Gateways New
Egress Only Internet Gateways New
Carrier Gateways New
DHCP Options Sets New
Elastic IPs New
Managed Prefix Lists New
Endpoints
Endpoint 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
Actions
Create VPC
1
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)

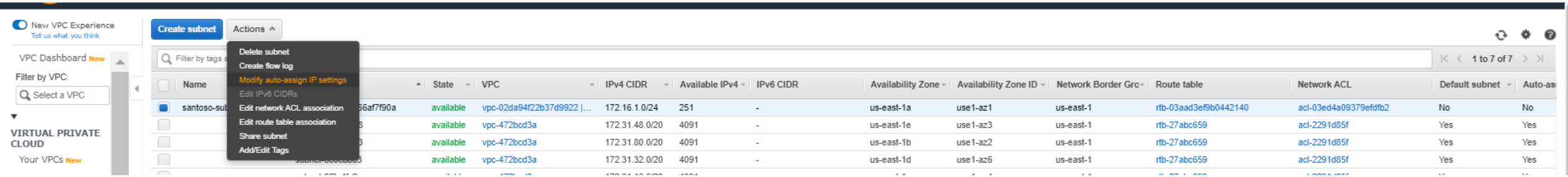


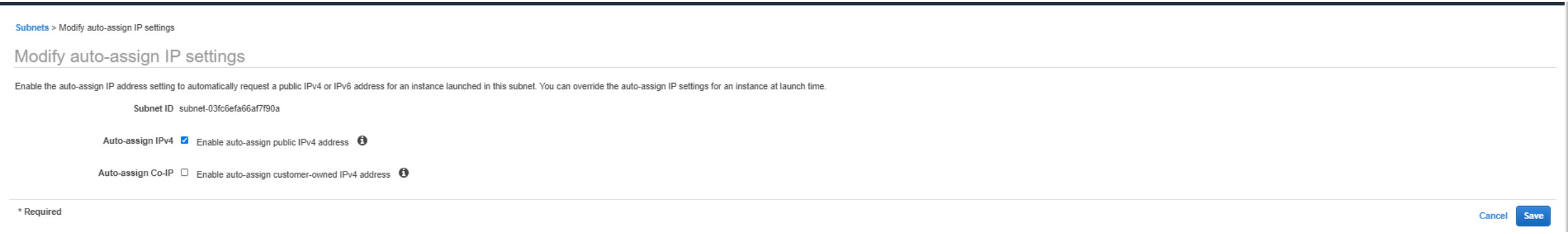


Check if subnet is created properly

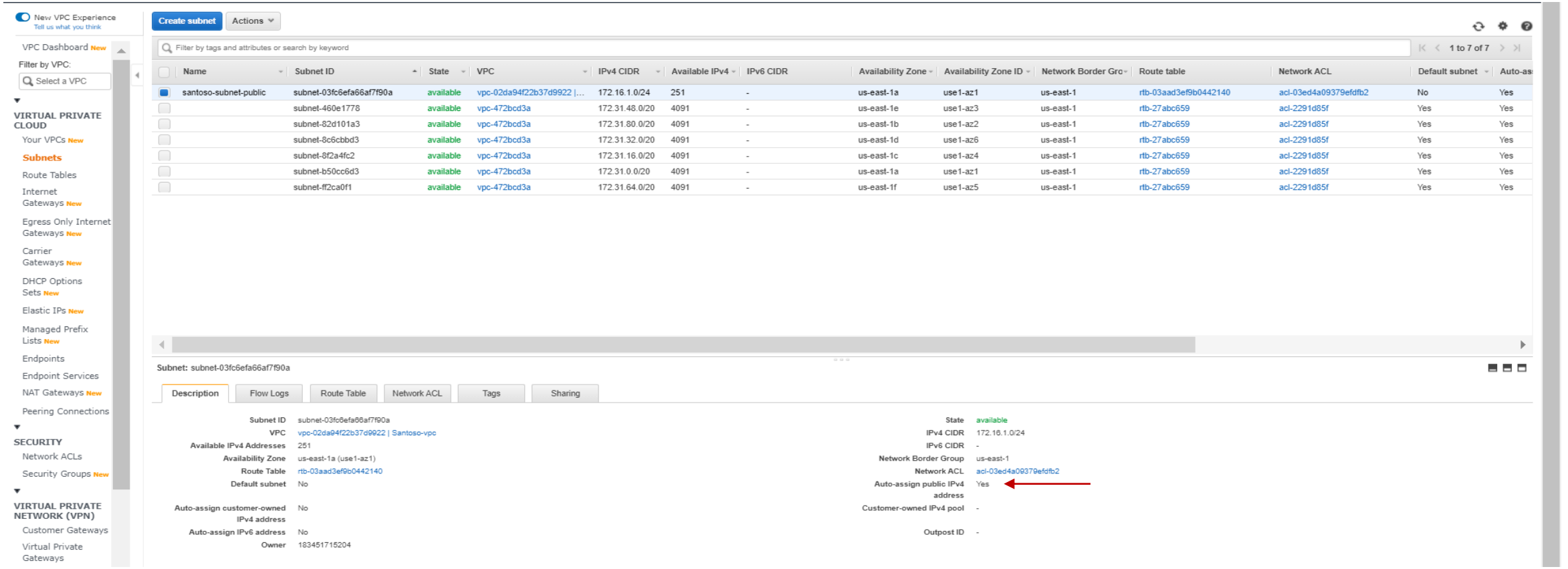


Subnet -Enable auto-assign IP settings (Assign a Public IP for all EC2s spinned up)

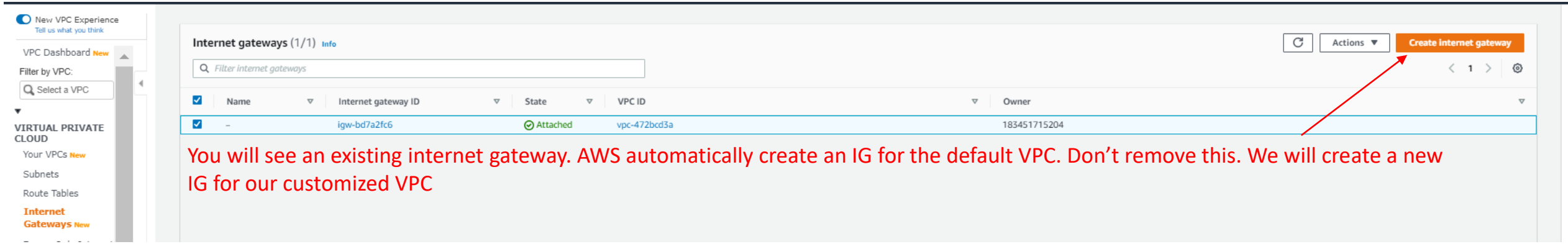


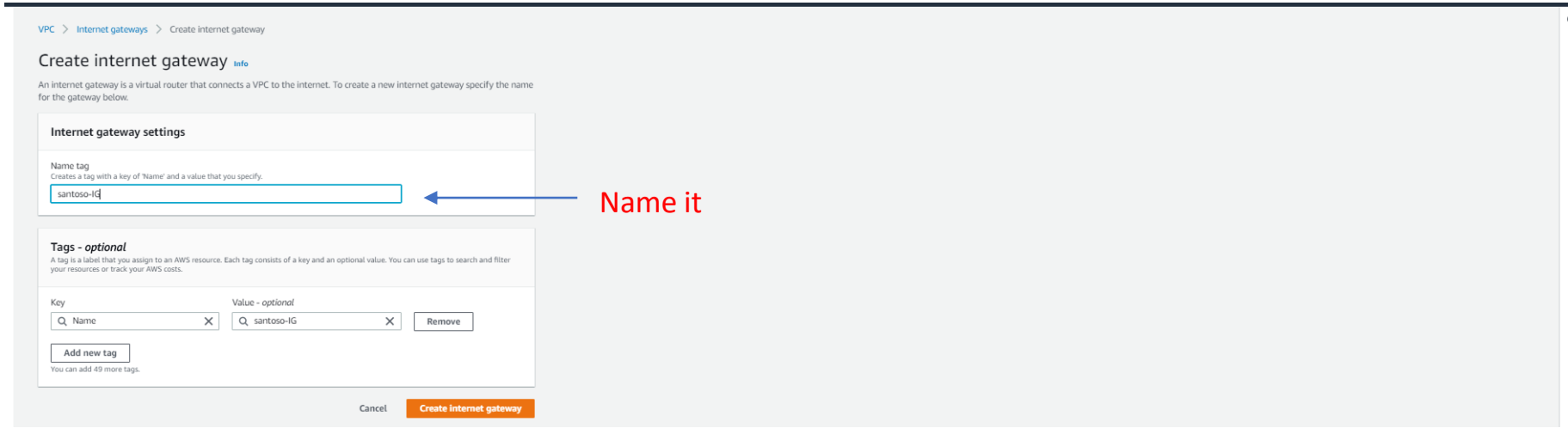


Check

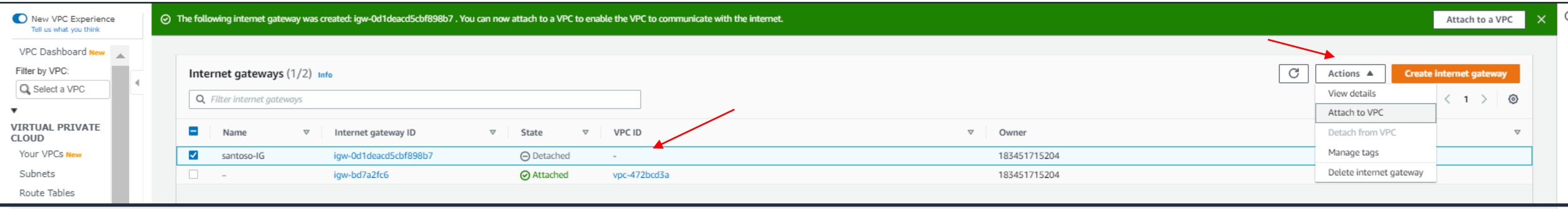

New VPC Experience
Tell us what you think
VPC Dashboard New
Filter by VPC:
Select a VPC
VIRTUAL PRIVATE CLOUD
Your VPCs New
Subnets
Route Tables
Internet Gateways New
Egress Only Internet Gateways New
Carrier Gateways New
DHCP Options Sets New
Elastic IPs New
Managed Prefix Lists New
Endpoints
Endpoint Services
NAT Gateways New
Peering Connections
SECURITY
Network ACLs
Security Groups New
VIRTUAL PRIVATE NETWORK (VPN)
Customer Gateways
Virtual Private Gateways
Create subnet
Actions
Filter by tags and attributes or search by keyword
1 to 7 of 7
Name | Subnet ID | State | VPC | IPv4 CIDR | Available IPv4 | IPv6 CIDR | Availability Zone | Availability Zone ID | Network Border Gro | Route table | Network ACL | Default subnet | Auto-as
santoso-subnet-public | subnet-03fc6efa66af7f90a | available | vpc-02da94f22b37d9922 |... | 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-ff2ca0f1 | 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
VPC vpc-02da94f22b37d9922 | Santoso-vpc
Available IPv4 Addresses 251
Availability Zone us-east-1a (use1-az1)
Route Table rtb-03aad3ef9b0442140
Default subnet No
Auto-assign customer-owned IPv4 address No
Auto-assign IPv6 address No
Owner 183451715204
State available
IPv4 CIDR 172.16.1.0/24
IPv6 CIDR -
Network Border Group us-east-1
Network ACL acl-03ed4a09379efdfb2
Auto-assign public IPv4 address Yes
Customer-owned IPv4 pool -
Outpost ID -

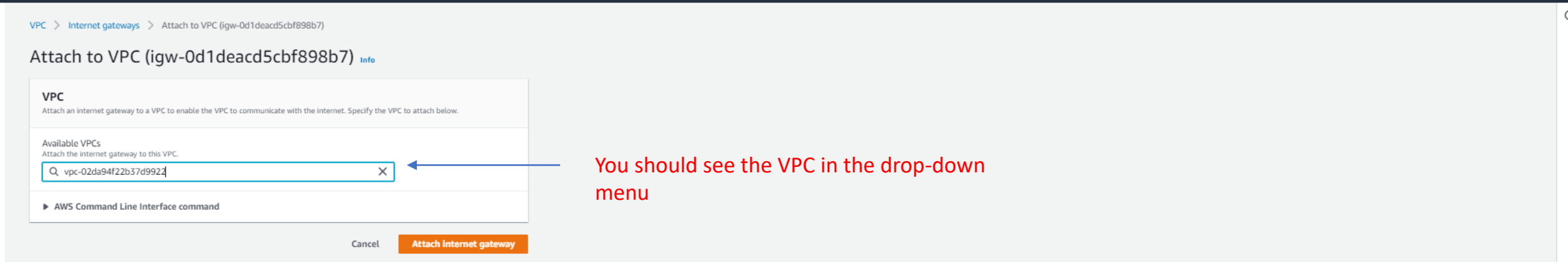
Create Internet Gateway



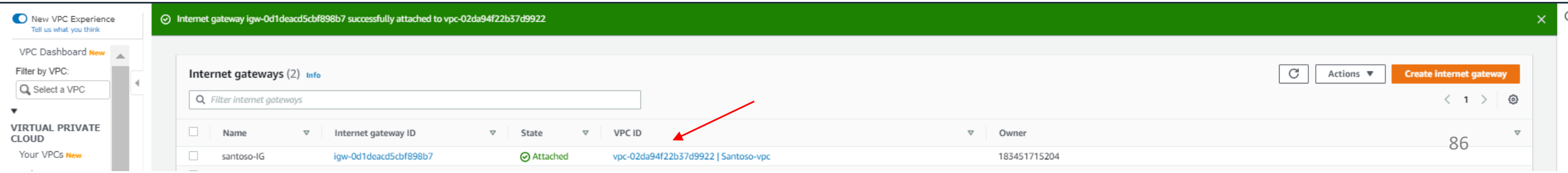


Attach IG to VPC

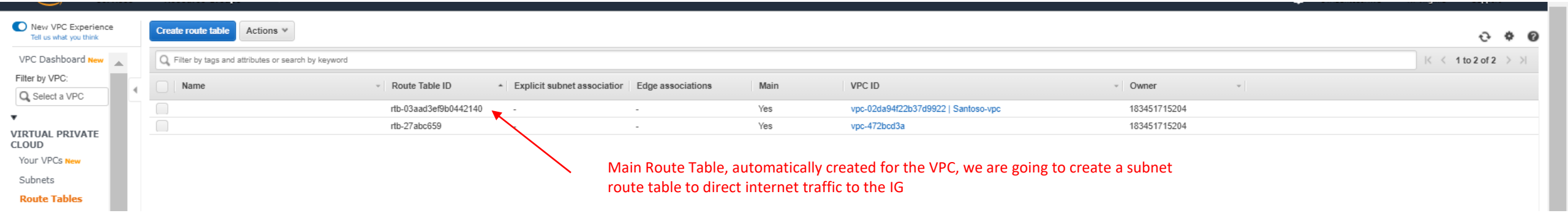


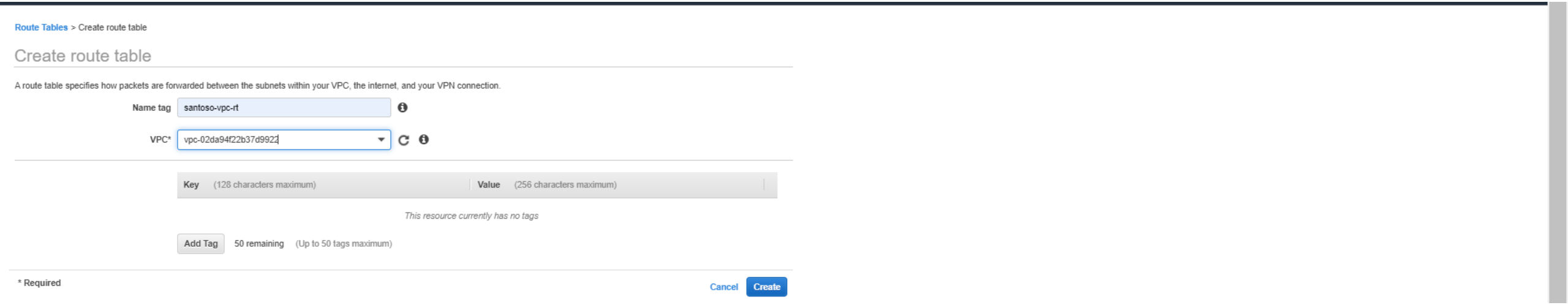


You should see the VPC in the drop-down menu

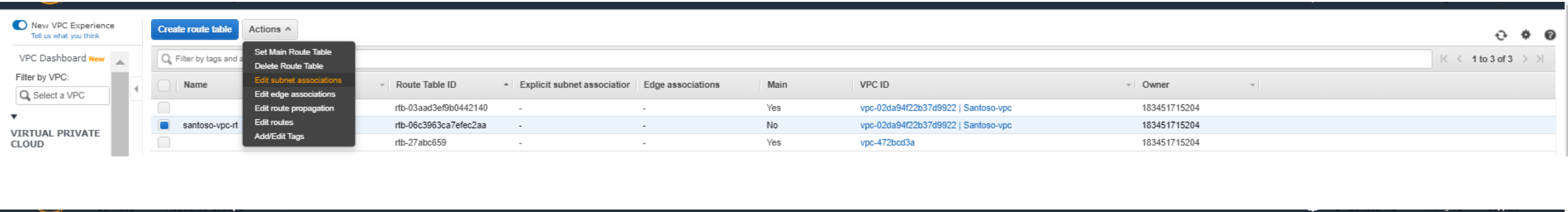


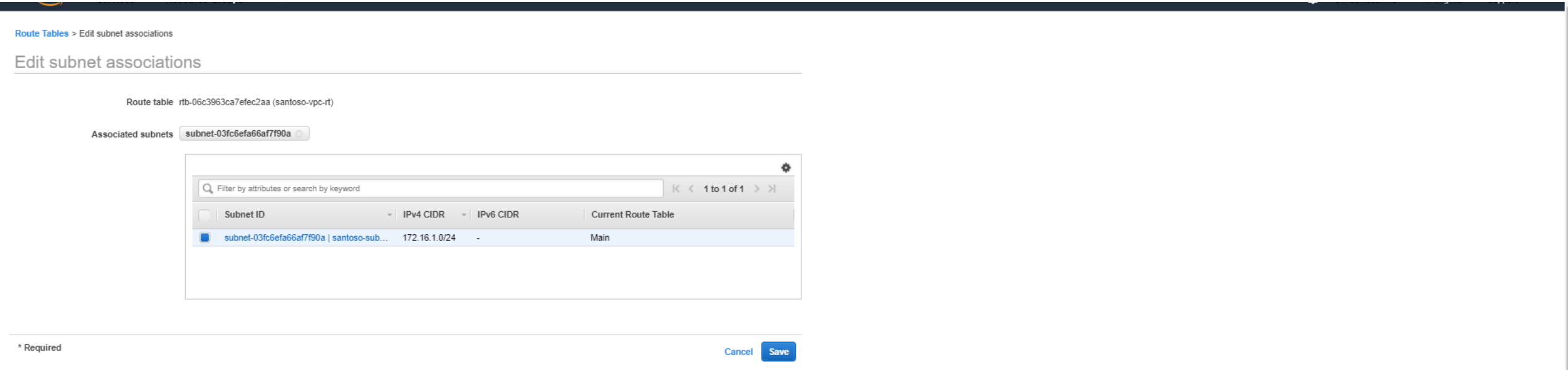
Create a Route Table for the subnet



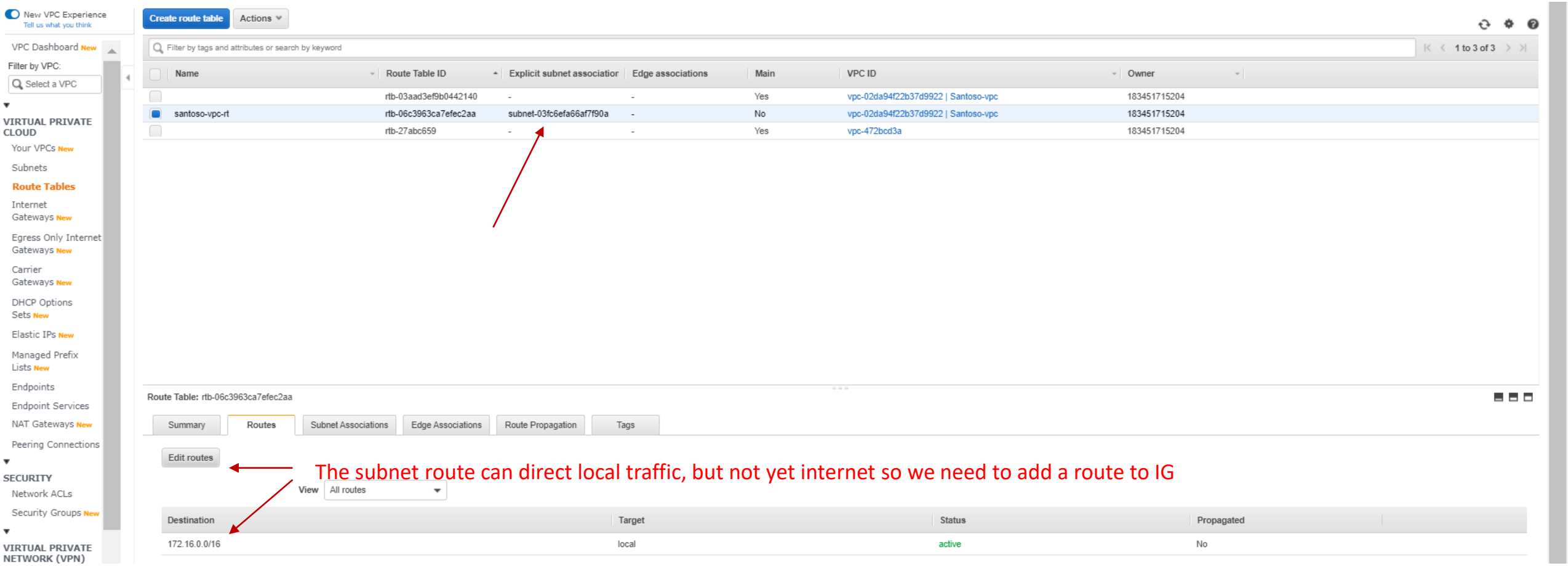


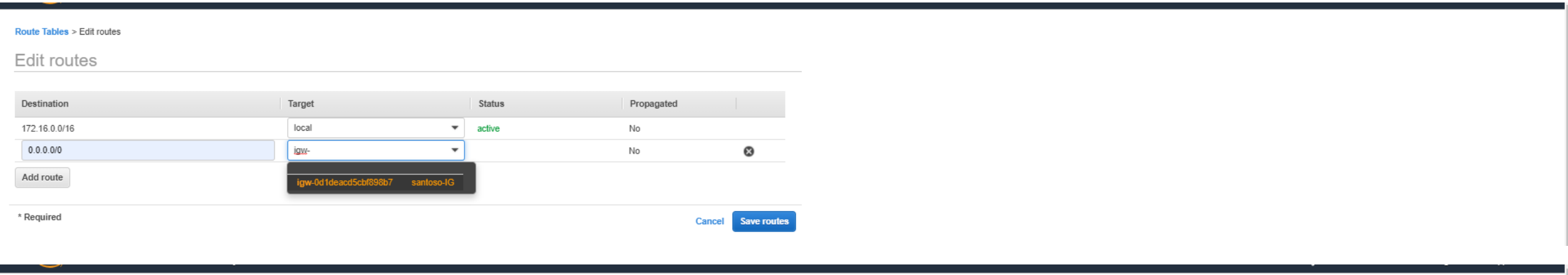
Associate to Subnet

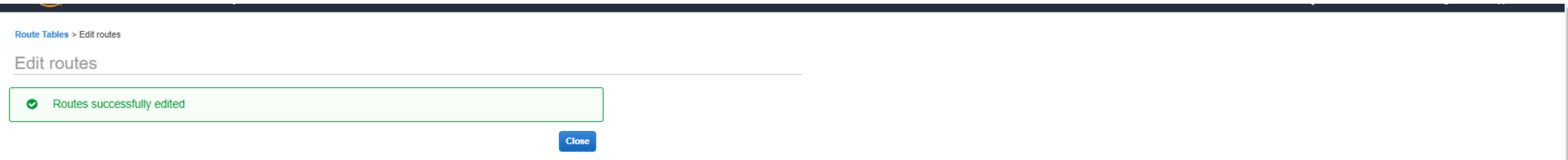




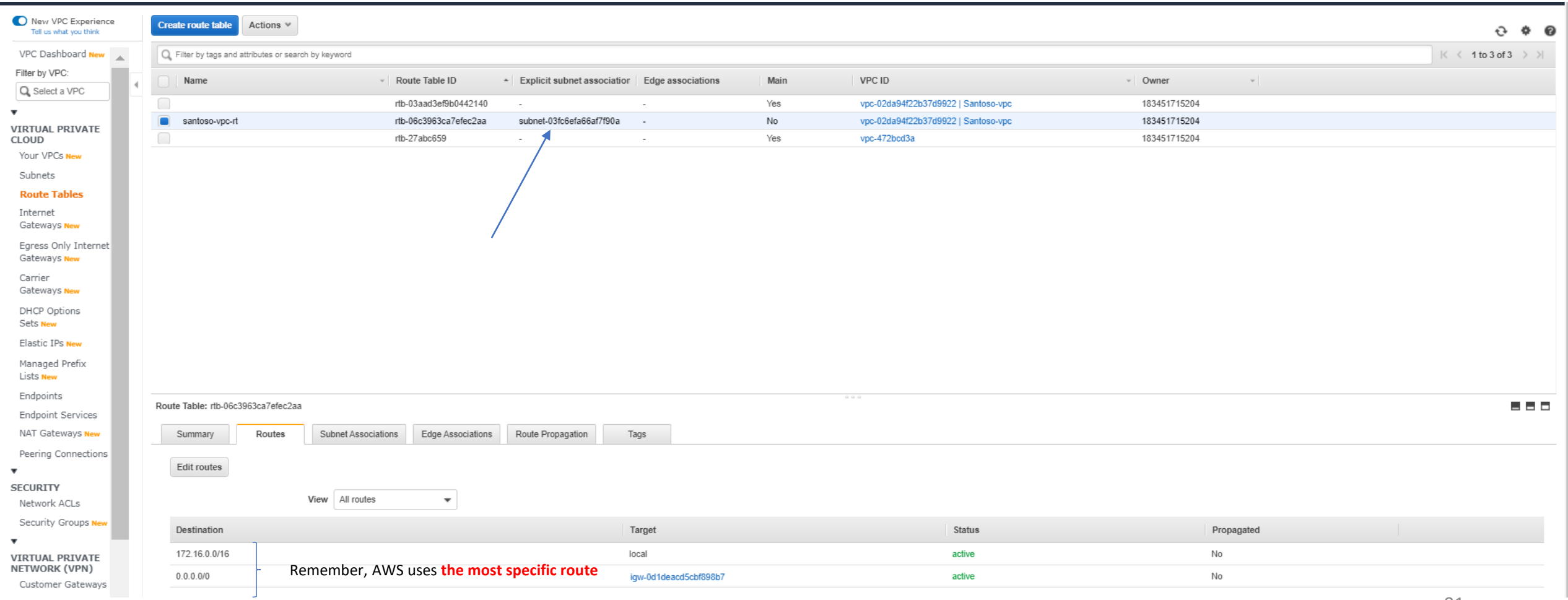
Edit Route to include IG



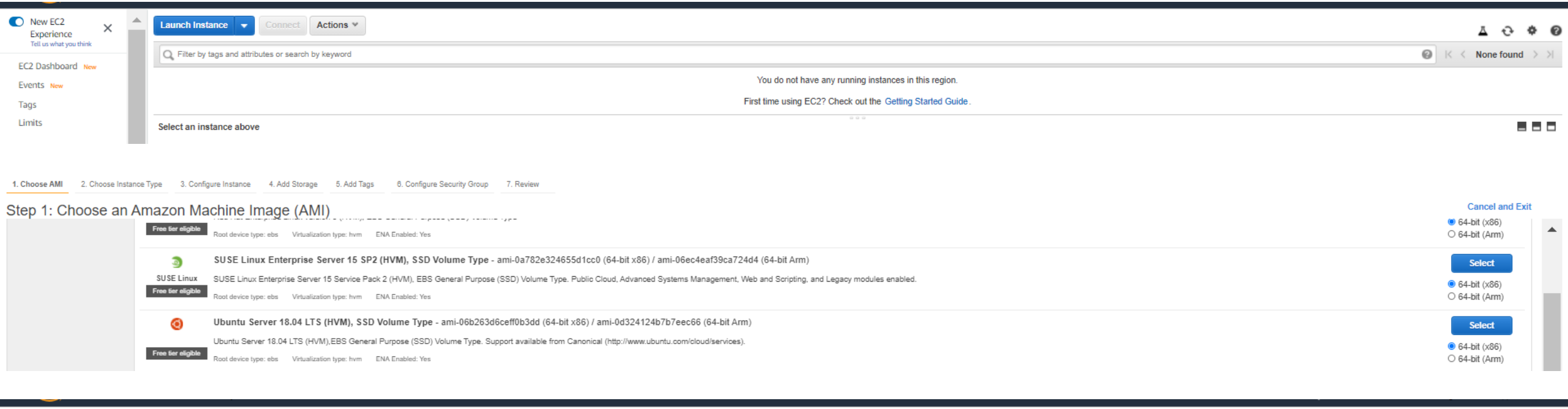

Route Tables > Edit routes
Edit routes
Destination
Target
Status
Propagated
172.16.0.0/16
local
active
No
0.0.0.0/0
igw-
No
Add route
igw-0d1deacd5cbf898b7 santoso-IG
* Required
Cancel
Save routes


Route Tables > Edit routes
Edit routes
Routes successfully edited
Close

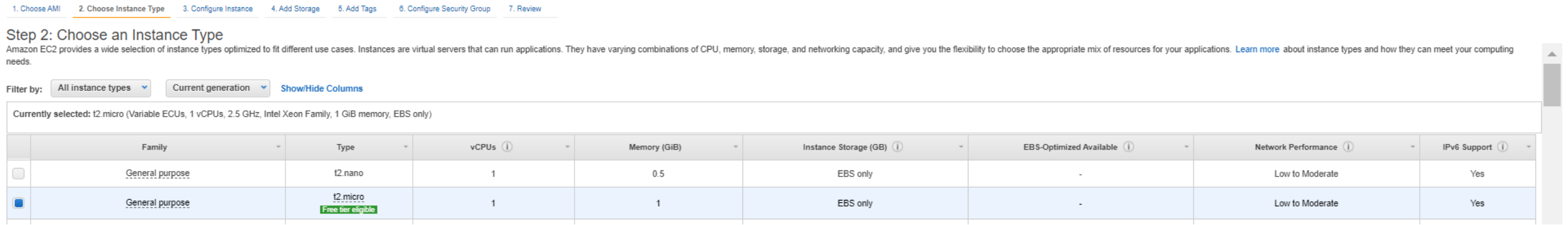
Check Subnet route table

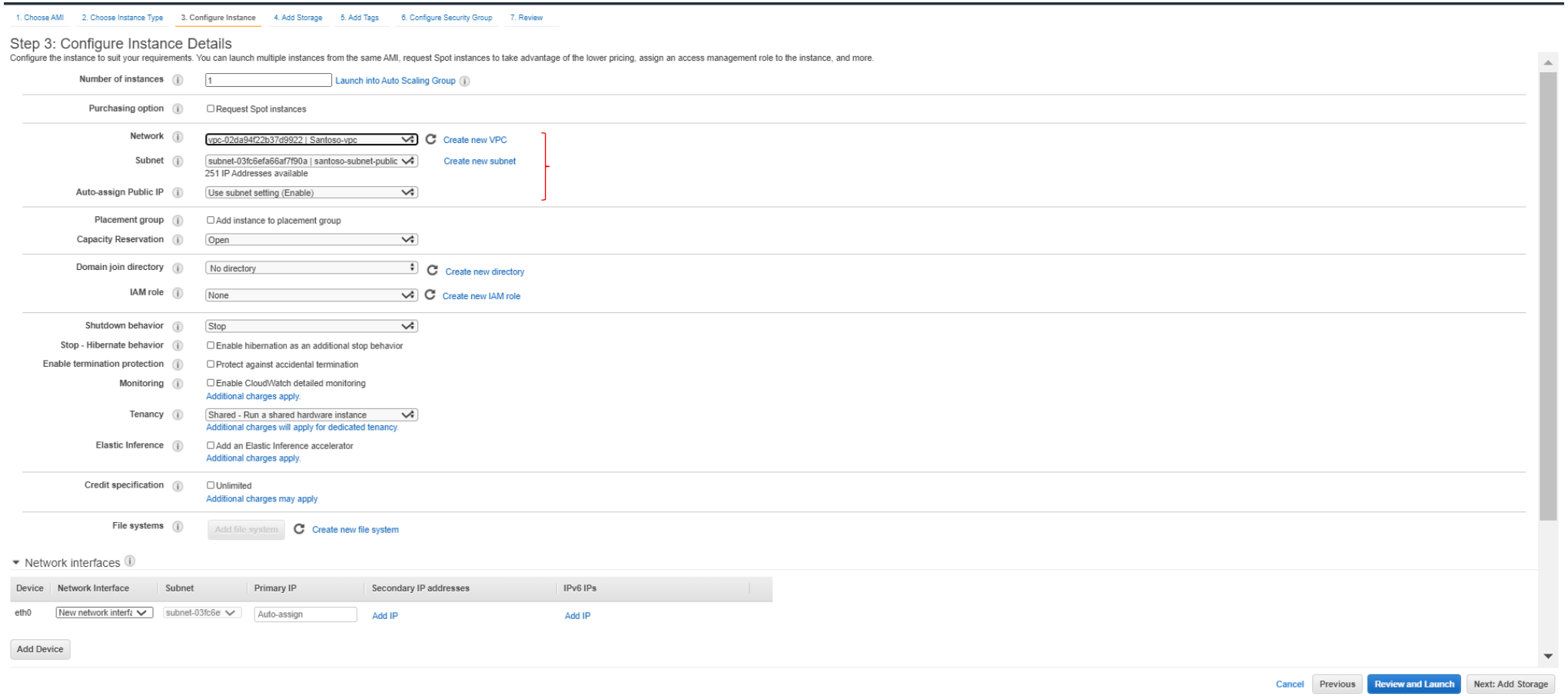


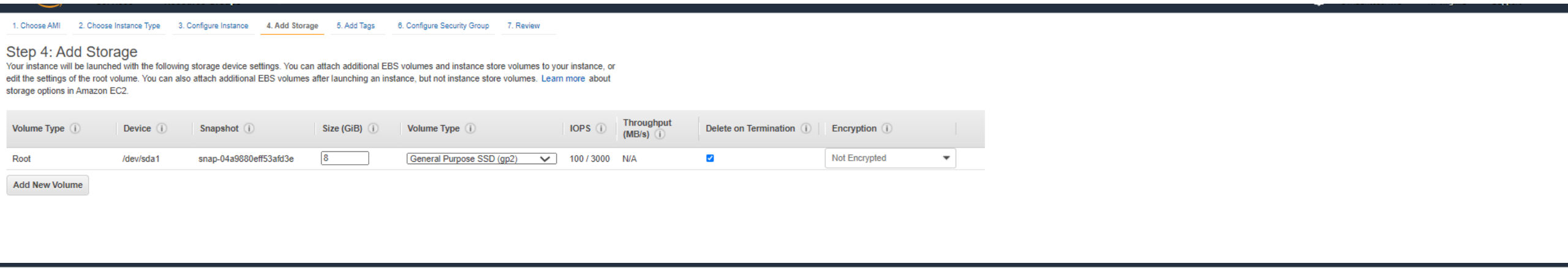
Spin up EC2

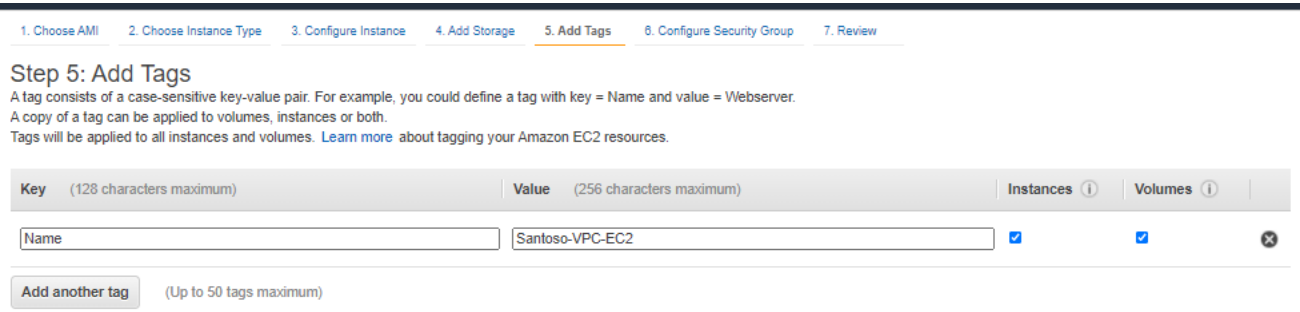

New EC2 Experience
Tell us what you think
EC2 Dashboard New
Events New
Tags
Limits
Launch Instance
Connect
Actions
Filter by tags and attributes or search by keyword
None found
You do not have any running instances in this region.
First time using EC2? Check out the Getting Started Guide.
Select an instance above

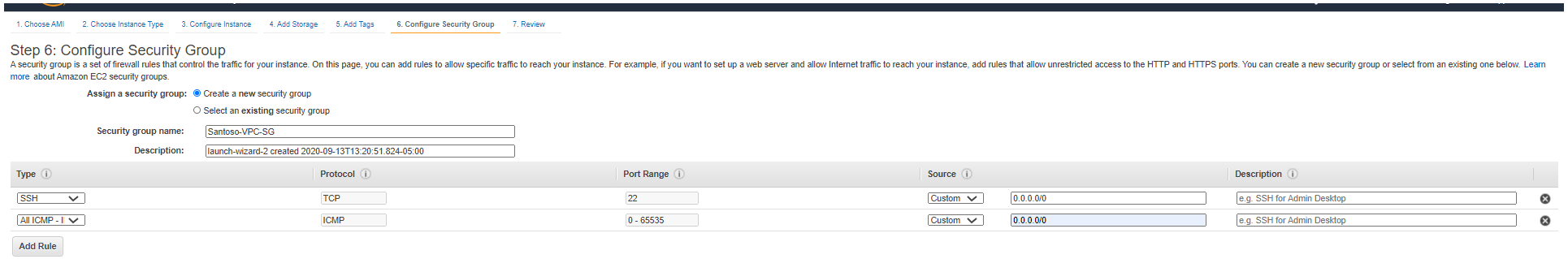
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 1: Choose an Amazon Machine Image (AMI)
Cancel and Exit
Free tier eligible
Root device type: ebs
Virtualization type: hvm
ENA Enabled: Yes
64-bit (x86)
64-bit (Arm)
SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type - ami-0a782e324655d1cc0 (64-bit x86) / ami-06ec4eaf39ca724d4 (64-bit Arm)
SUSE Linux
SUSE Linux Enterprise Server 15 Service Pack 2 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web and Scripting, and Legacy modules enabled.
Select
Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-06b263d6ceff0b3dd (64-bit x86) / ami-0d324124b7b7eec66 (64-bit Arm)
Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

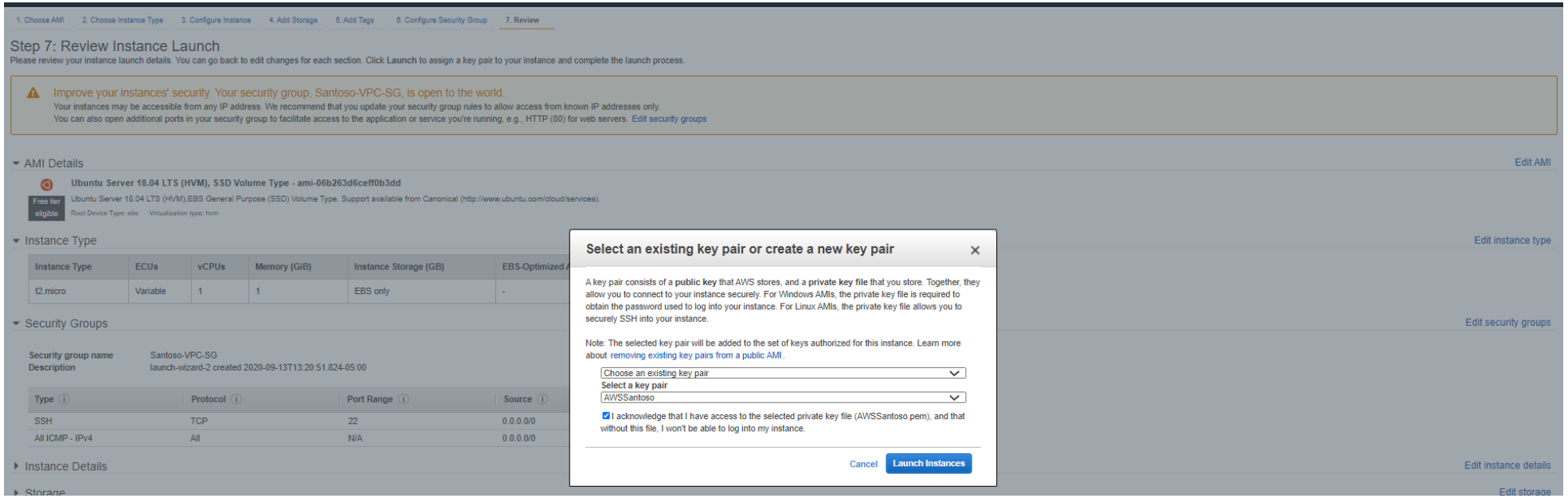

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 2: Choose an Instance Type
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.
Filter by:
All instance types
Current generation
Show/Hide Columns
Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)
Family
Type
vCPUs
Memory (GiB)
Instance Storage (GB)
EBS-Optimized Available
Network Performance
IPv6 Support
General purpose
t2.nano
1
0.5
EBS only
-
Low to Moderate
Yes
General purpose
t2.micro
Free tier eligible
1
1
EBS only
-
Low to Moderate
Yes

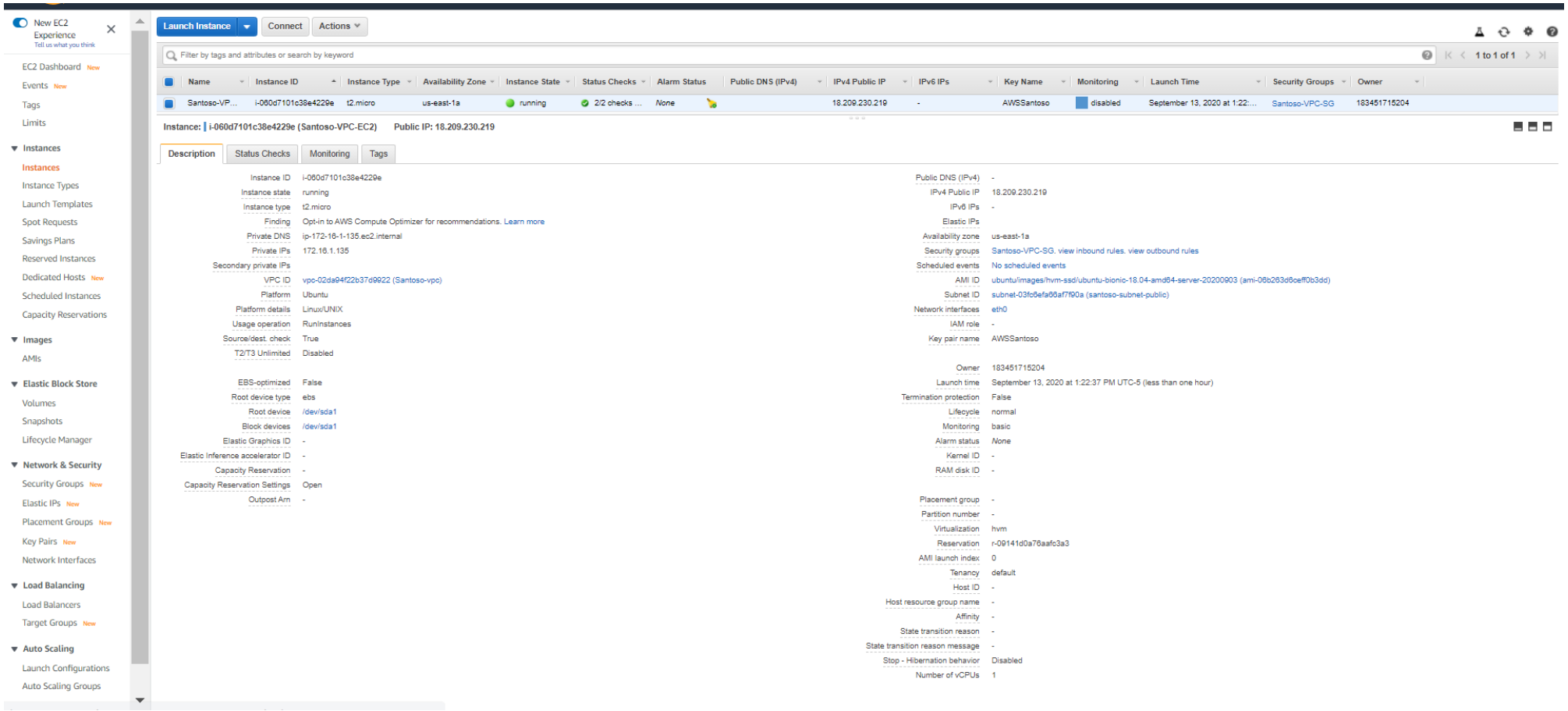

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 3: Configure Instance Details
Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.
Number of instances
1
Launch into Auto Scaling Group
Purchasing option
Request Spot instances
Network
vpc-02da94f22b37d9922 | Santoso-vpc
Create new VPC
Subnet
subnet-03fc6efa66af7f90a | santoso-subnet-public
Create new subnet
251 IP Addresses available
Auto-assign Public IP
Use subnet setting (Enable)
Placement group
Add instance to placement group
Capacity Reservation
Open
Domain join directory
No directory
Create new directory
IAM role
None
Create new IAM role
Shutdown behavior
Stop
Stop - Hibernate behavior
Enable hibernation as an additional stop behavior
Enable termination protection
Protect against accidental termination
Monitoring
Enable CloudWatch detailed monitoring
Additional charges apply.
Tenancy
Shared - Run a shared hardware instance
Additional charges will apply for dedicated tenancy.
Elastic Inference
Add an Elastic Inference accelerator
Additional charges apply.
Credit specification
Unlimited
Additional charges may apply
File systems
Add file system
Create new file system
Network interfaces
Device
Network Interface
Subnet
Primary IP
Secondary IP addresses
IPv6 IPs
eth0
New network interfa
subnet-03fc6e
Auto-assign
Add IP
Add IP
Add Device
Cancel
Previous
Review and Launch
Next: Add Storage


1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
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
Not Encrypted
Add New Volume

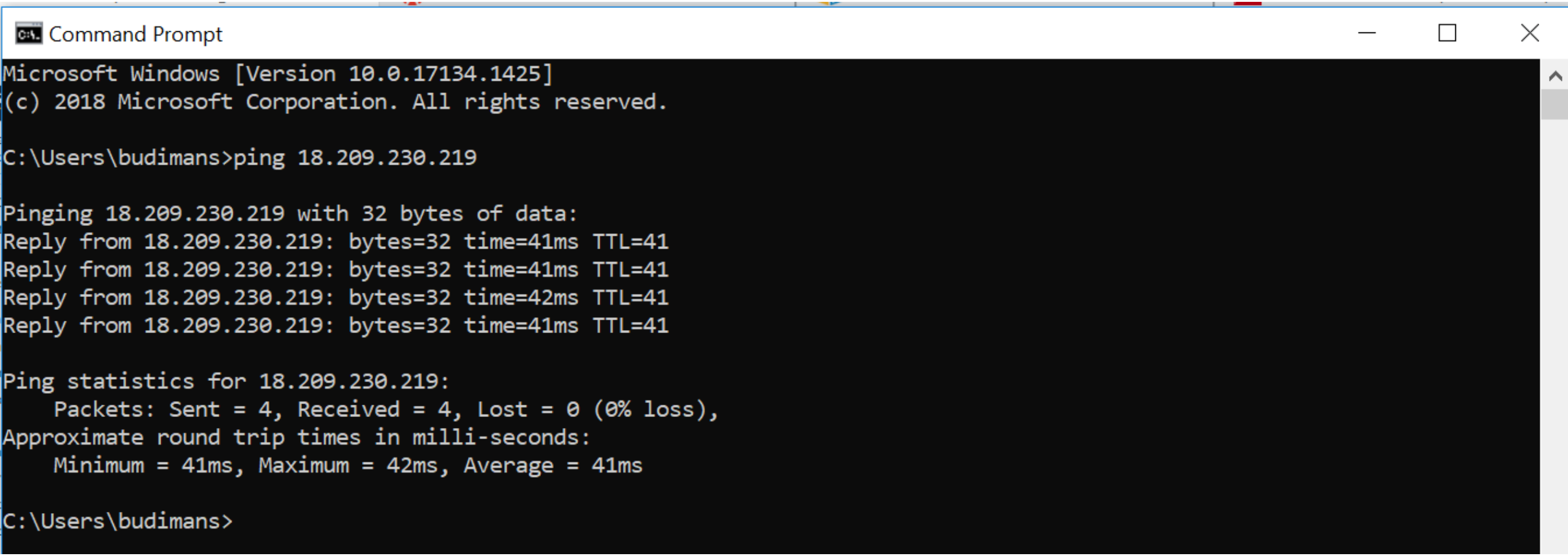

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 5: Add Tags
A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to 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
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: Santoso-VPC-SG
Description: launch-wizard-2 created 2020-09-13T13:20:51.824-05:00
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

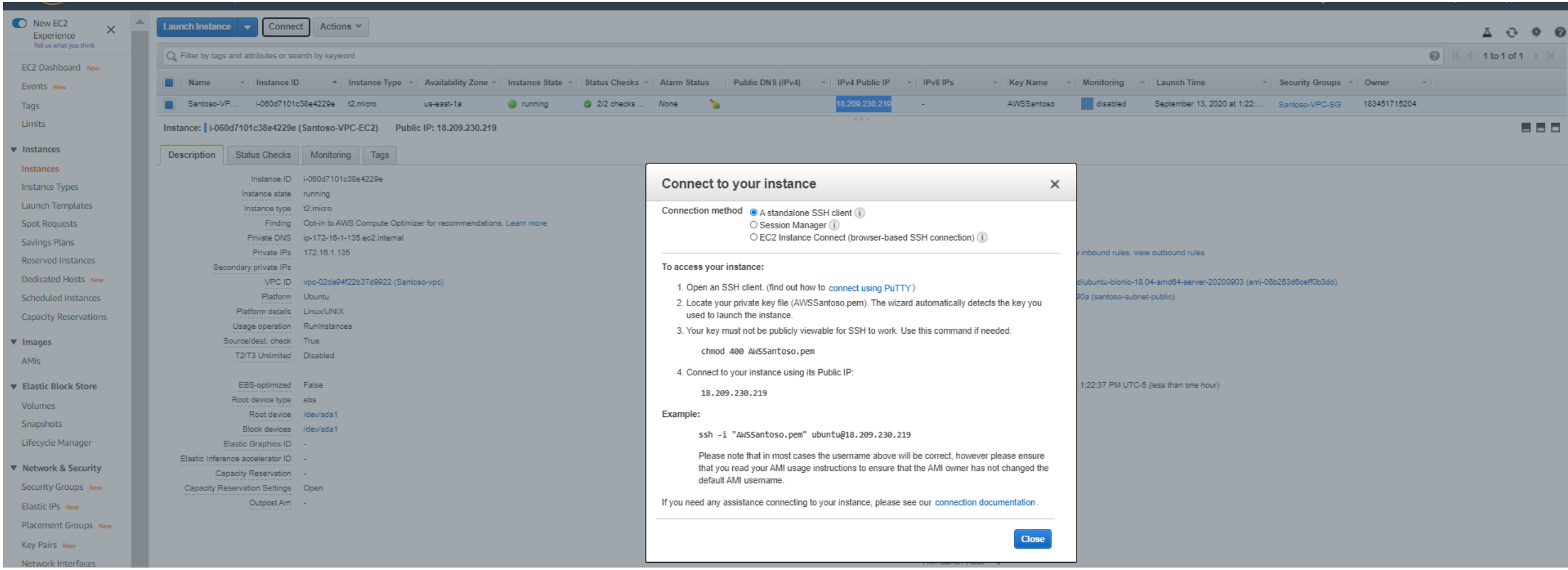

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 7: Review Instance Launch
Please review your instance launch details. You can go back to edit changes for each section. Click Launch to assign a key pair to your instance 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).
Root Device Type: ebs
Virtualization type: hvm
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
Instance Details
Edit instance details
Storage
Edit storage
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
Select a 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

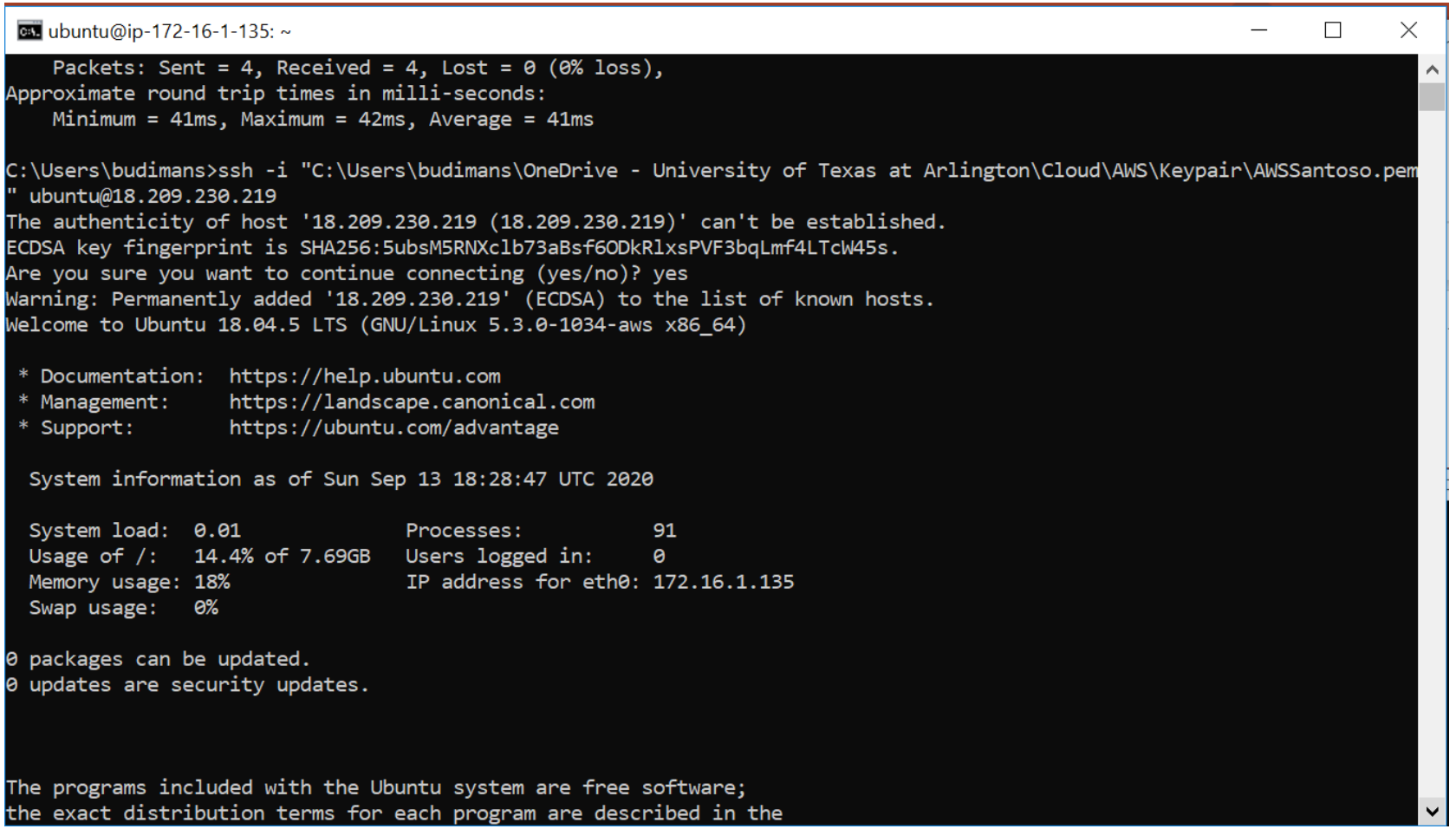

New EC2 Experience
Tell us what you think
EC2 Dashboard New
Events New
Tags
Limits
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts New
Scheduled Instances
Capacity Reservations
Images
AMIs
Elastic Block Store
Volumes
Snapshots
Lifecycle 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
1 to 1 of 1
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-VP...
i-060d7101c38e4229e
t2.micro
us-east-1a
running
2/2 checks ...
None
18.209.230.219
-
AWSSantoso
disabled
September 13, 2020 at 1:22:...
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
Instance state running
Instance type t2.micro
Finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
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
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 -
Public DNS (IPv4) -
IPv4 Public IP 18.209.230.219
IPv6 IPs -
Elastic IPs
Availability zone us-east-1a
Security groups Santoso-VPC-SG. view inbound rules. view outbound rules
Scheduled events No scheduled events
AMI ID ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20200903 (ami-06b263d6ceff0b3dd)
Subnet ID subnet-03fc6efa66af7f90a (santoso-subnet-public)
Network interfaces eth0
IAM role -
Key pair name AWSSantoso
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)

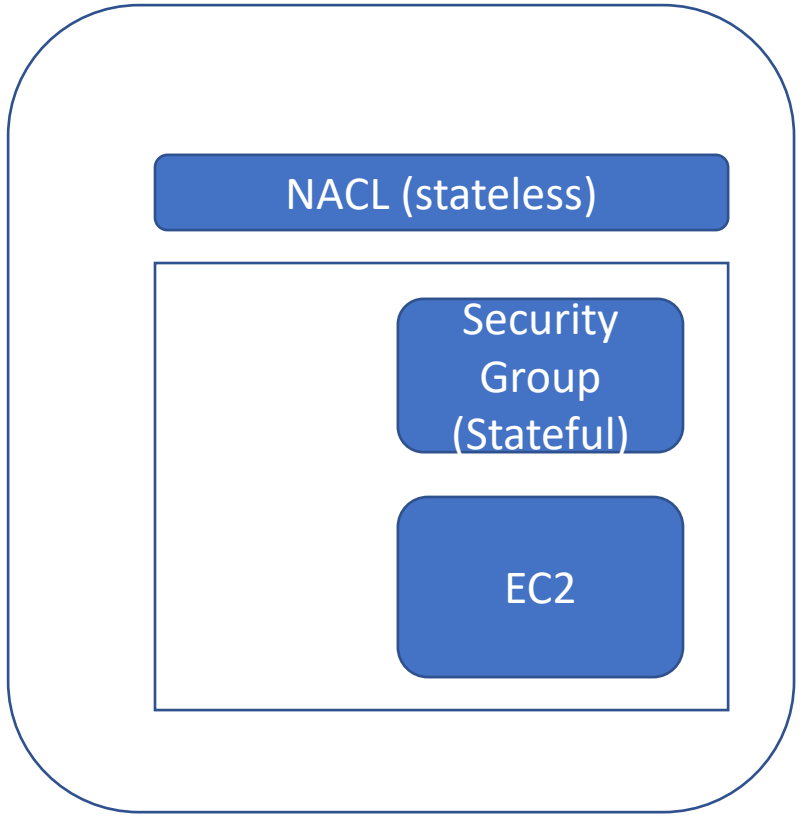


Test SSH

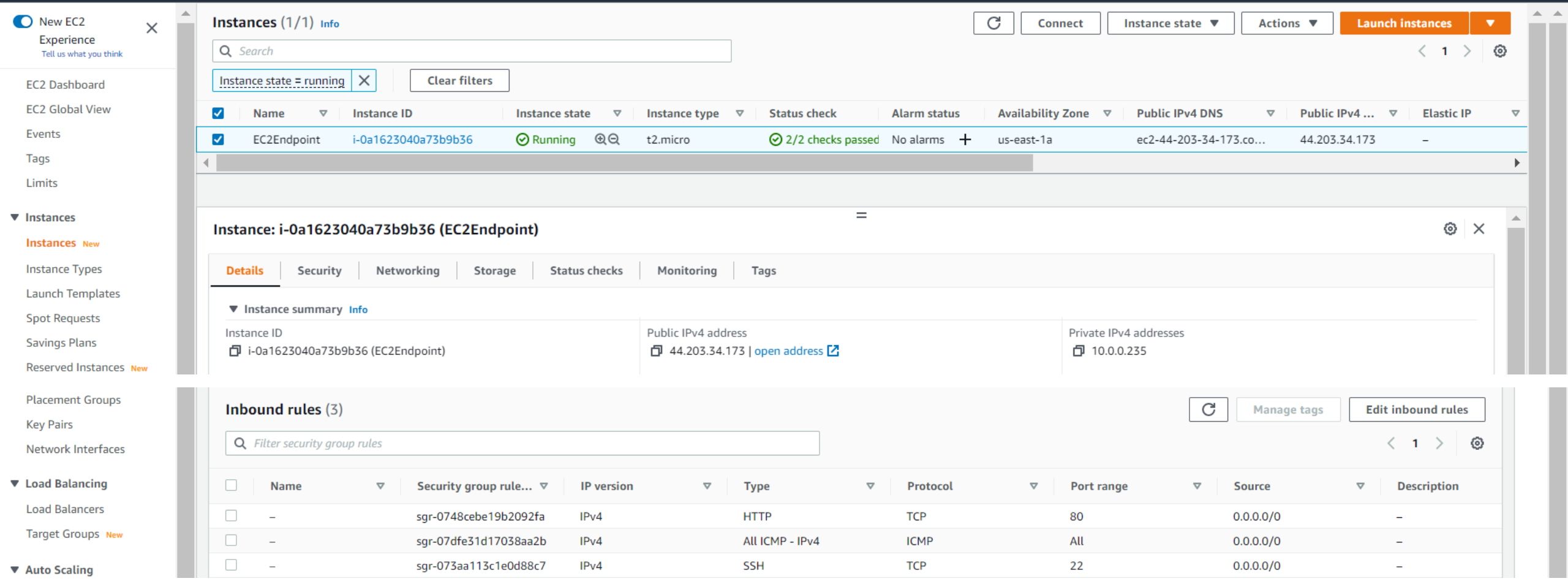

New EC2 Experience
Launch Instance
Connect
Actions
Filter by tags and attributes or search by keyword
1 to 1 of 1
Name
Instance ID
Instance Type
Availability Zone
Instance State
Status Checks
Alarm Status
Public DNS (IPv4)
IPv4 Public IP
IPv6 IPs
Key Name
Monitoring
Launch Time
Security Groups
Owner
Santoso-VP...
i-060d7101c38e4229e
t2.micro
us-east-1a
running
2/2 checks ...
None
18.209.230.219
AWSSantoso
disabled
September 13, 2020 at 1:22:...
Santoso-VPC-SG
183451715204
Instance: i-060d7101c38e4229e (Santoso-VPC-EC2)
Public IP: 18.209.230.219
Description
Status Checks
Monitoring
Tags
Connect to your instance
Connection method
A standalone SSH client
Session Manager
EC2 Instance Connect (browser-based SSH connection)
To access your instance:
1. Open an SSH client. (find out how to connect using PuTTY)
2. Locate your private key file (AWSSantoso.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:
chmod 400 AWSSantoso.pem
4. 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.
Close


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 SHA256:5ubsM5RNXclb73aBsf6ODkRlxsPVF3bqLmf4LTcW45s.
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

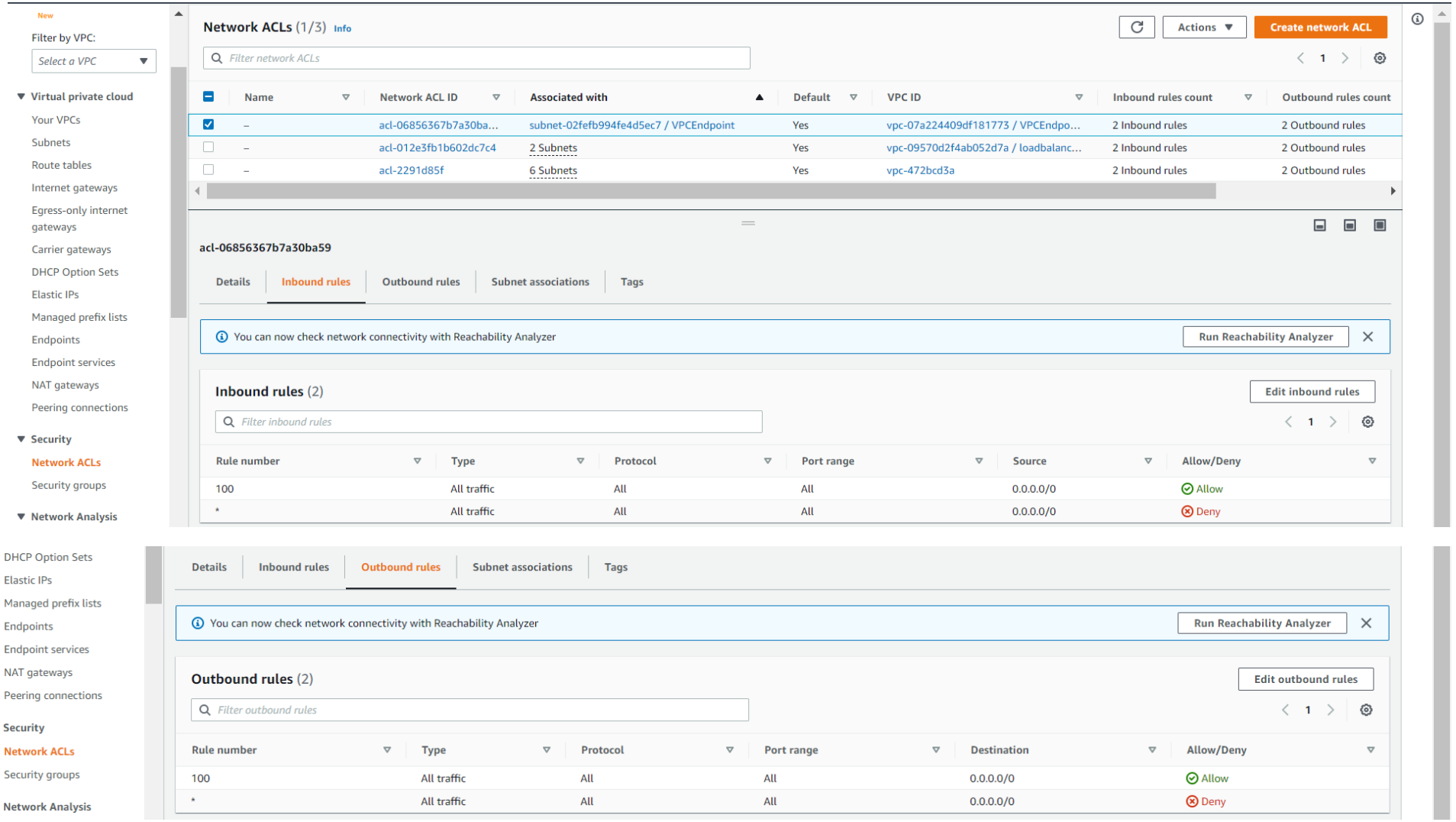


- Security Group inbound is set to allow ping (all icmp) from anywhere.
- The NACL default is allowing all inbound and outbound traffic.
- Add an NACL rule to deny outbound icmp.

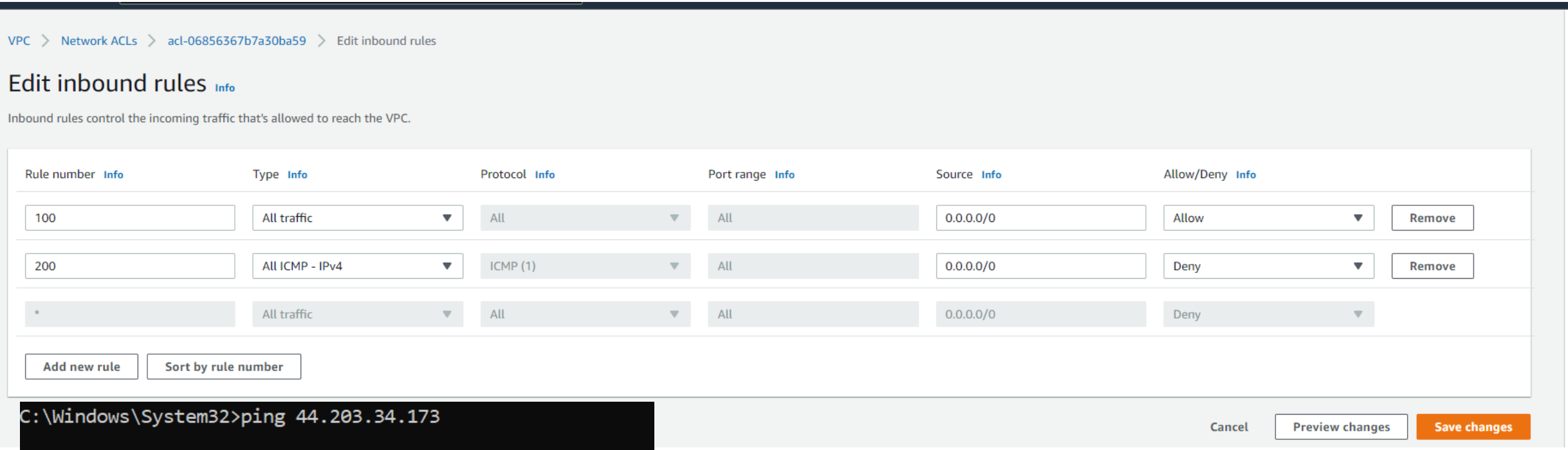



```
C:\Windows\System32>ping 44.203.34.173

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



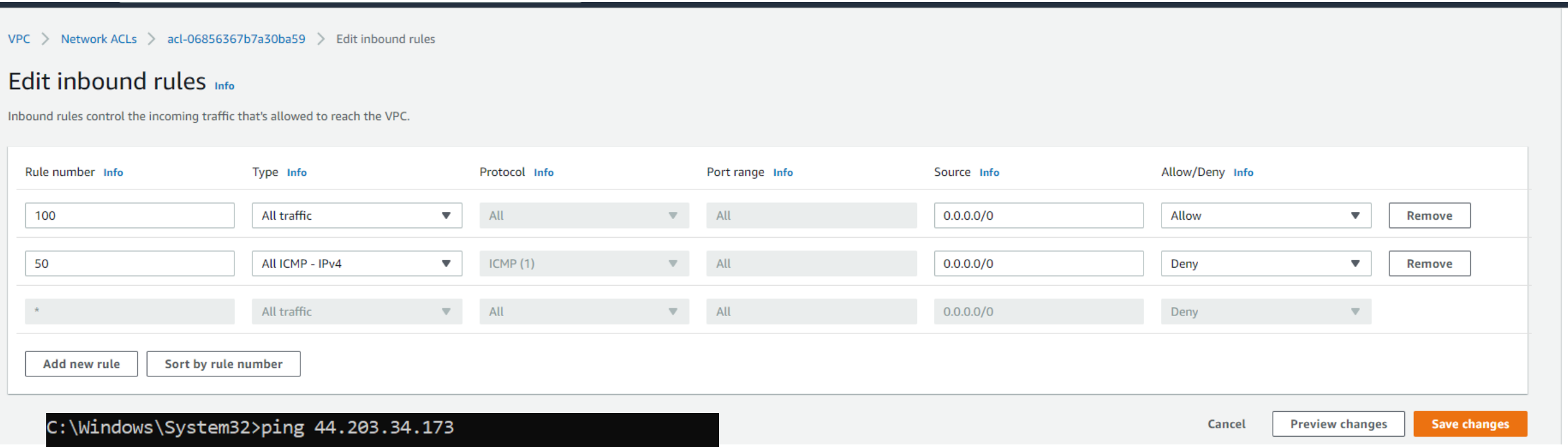
Add inbound rule number 200 deny all ICMP




```
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>
```

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

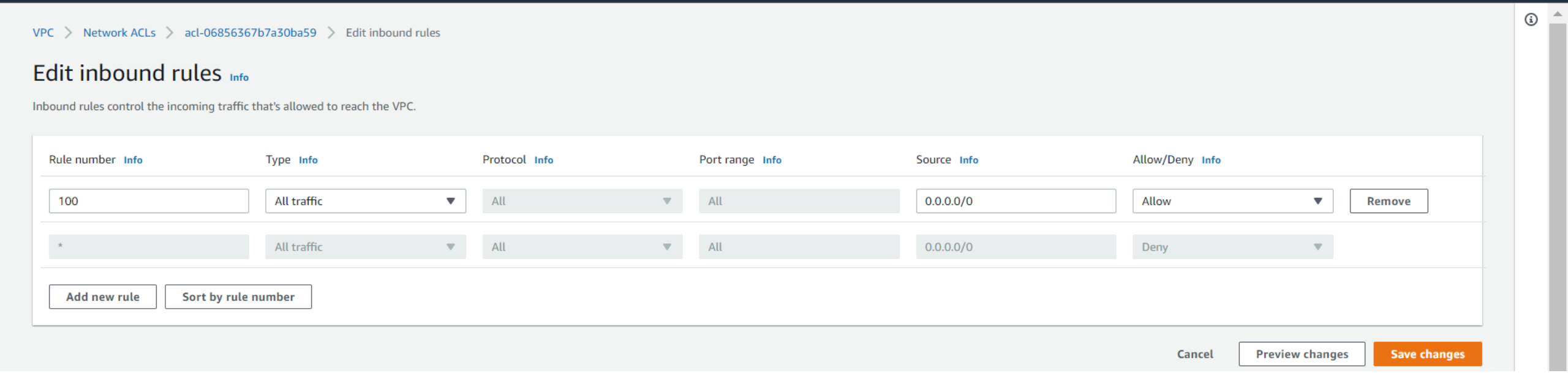
Change inbound rule number deny all ICMP to rule number 50




```
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>
```

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

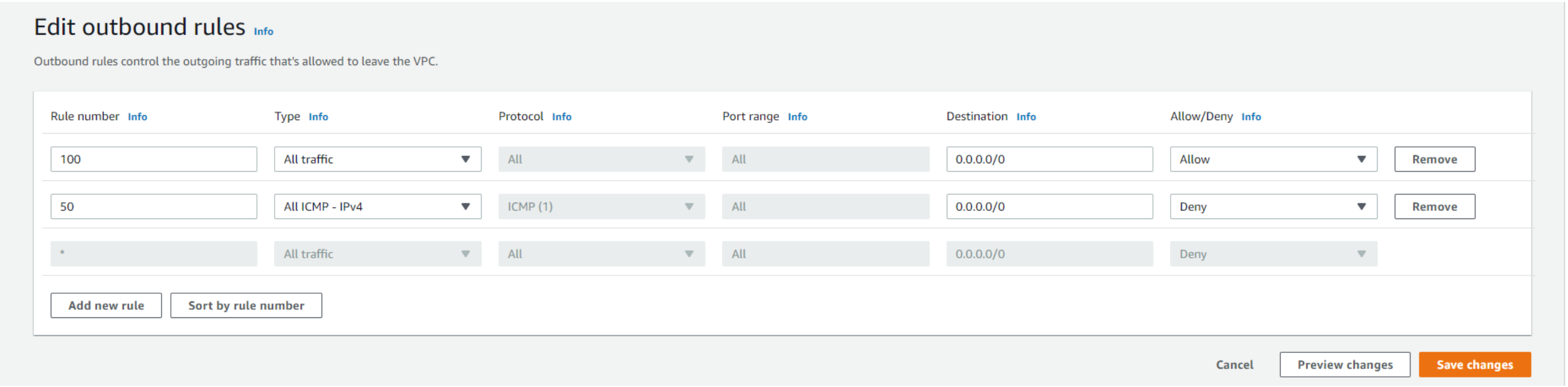
Change the inbound rule and remove the deny all icmp.




```
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.

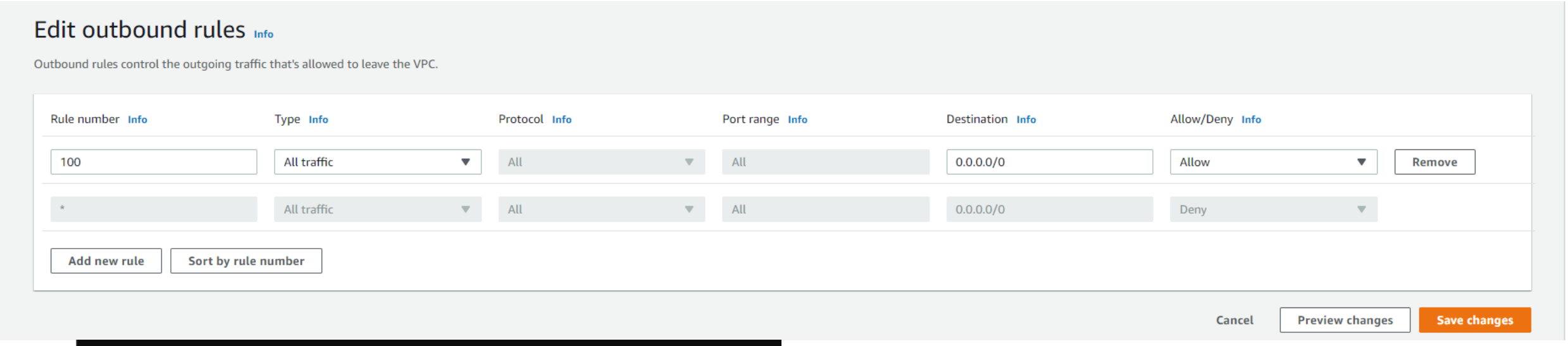
Add outbound rule to deny all ICMP




```
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.

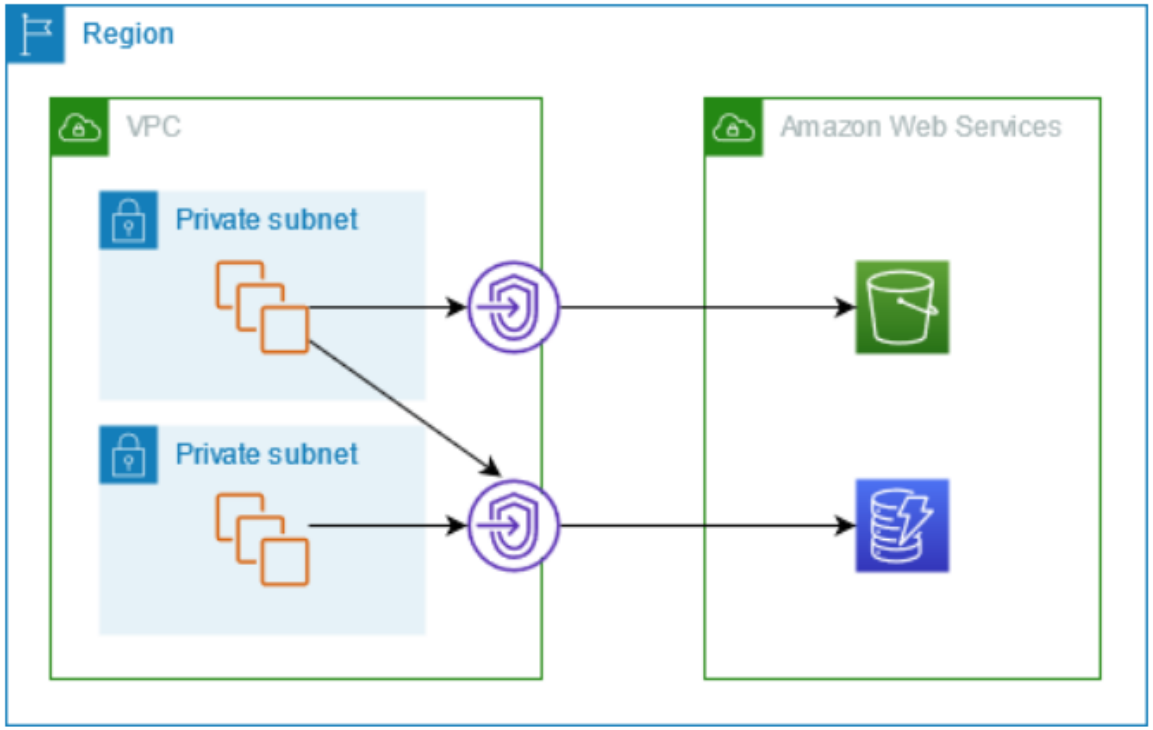
Restore the outbound rule




```
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.

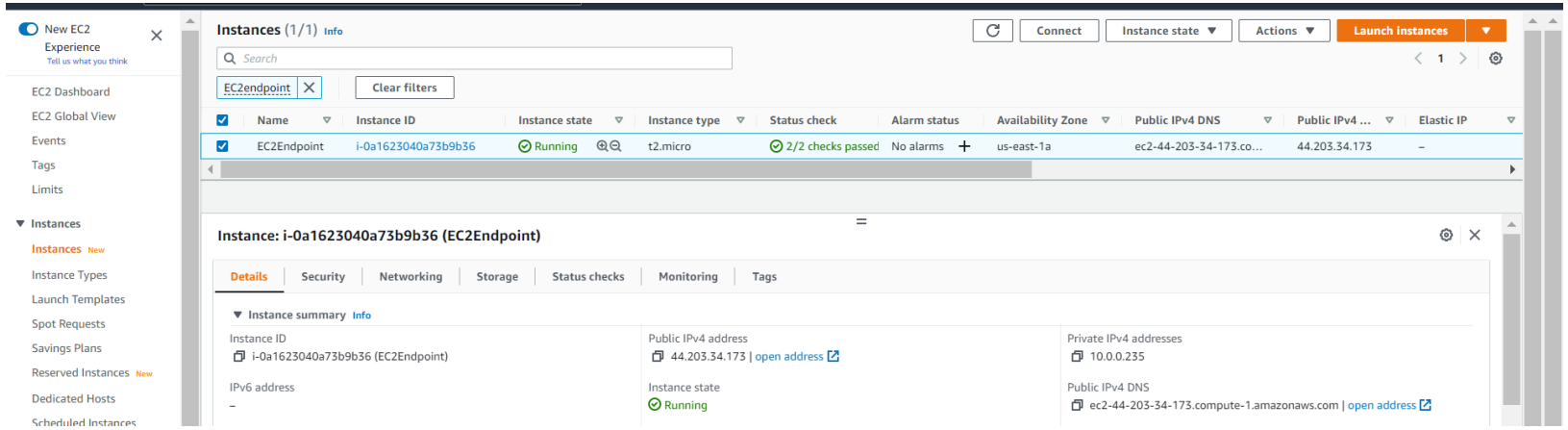
Demo 3: VPC Endpoint – Gateway Endpoint – S3

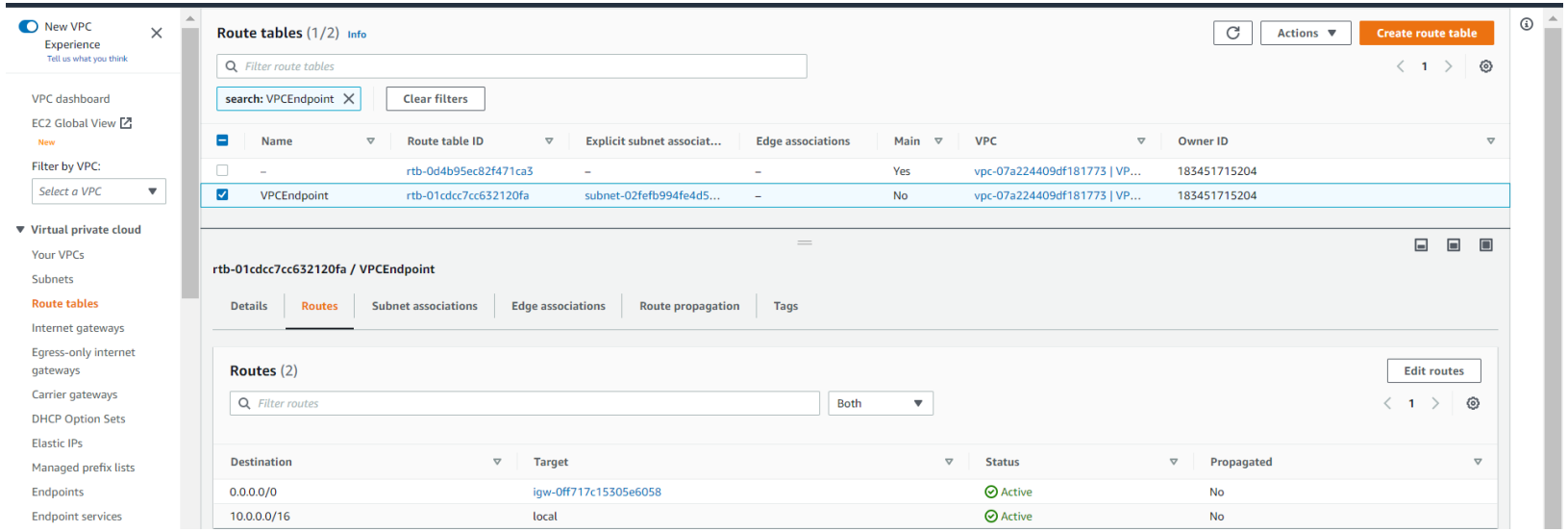


VPC endpoint is the entry point in your VPC that enables you to connect privately to a service, does not go through the internet.
Gateway Endpoint only for S3 and DynamoDB
https://docs.aws.amazon.com/vpc/latest/privatelink/gateway-endpoints.html

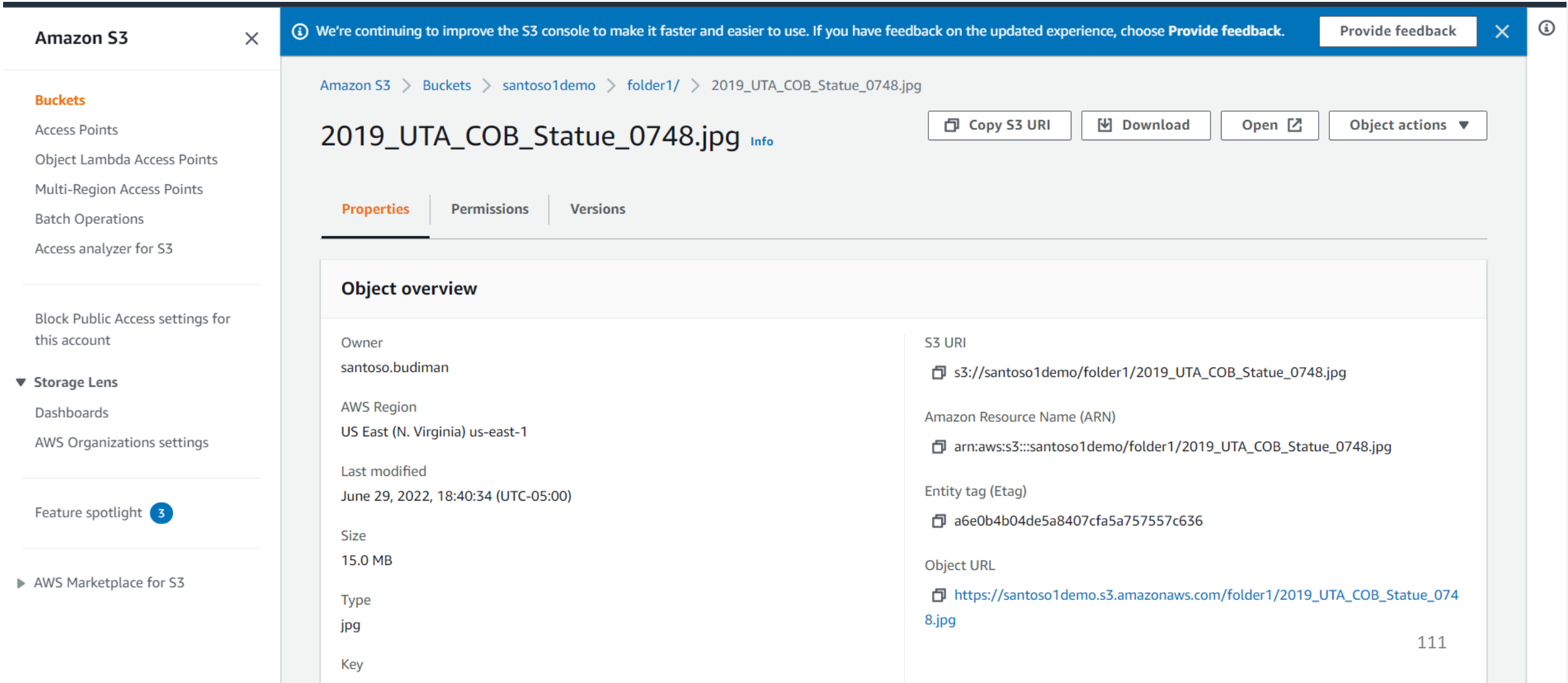
- In this demo, I have created an object (a picture) in an S3 bucket.
- Using the wget command we can download file from the web. We will use this to test.
  - wget is a networking command-line tool that allow people to download files and interact with REST APIs.
  - wget package is pre-installed in 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 weget command. If this works, the traffic goes through the private link.

EC2 original setup and Route Table





Object URL

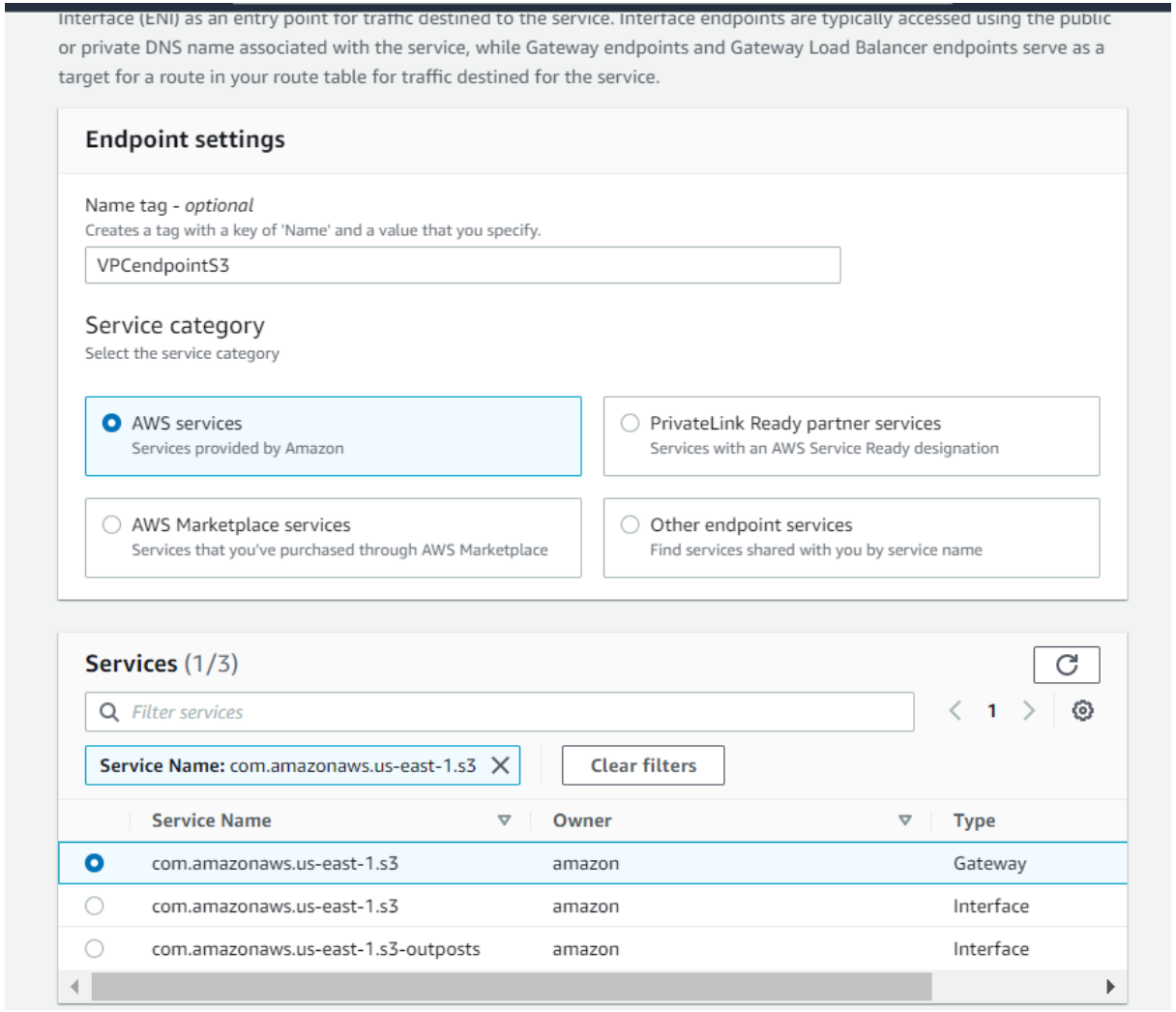

Amazon S3
We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.
Provide feedback
Buckets
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Amazon S3 > Buckets > santoso1demo > folder1/ > 2019_UTA_COB_Statue_0748.jpg
2019_UTA_COB_Statue_0748.jpg Info
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
Object overview
Owner
santoso.budiman
AWS Region
US East (N. Virginia) us-east-1
Last modified
June 29, 2022, 18:40:34 (UTC-05:00)
Size
15.0 MB
Type
jpg
Key
S3 URI
s3://santoso1demo/folder1/2019_UTA_COB_Statue_0748.jpg
Amazon Resource Name (ARN)
arn:aws:s3:::santoso1demo/folder1/2019_UTA_COB_Statue_0748.jpg
Entity tag (Etag)
a6e0b4b04de5a8407cfa5a757557c636
Object URL
https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg

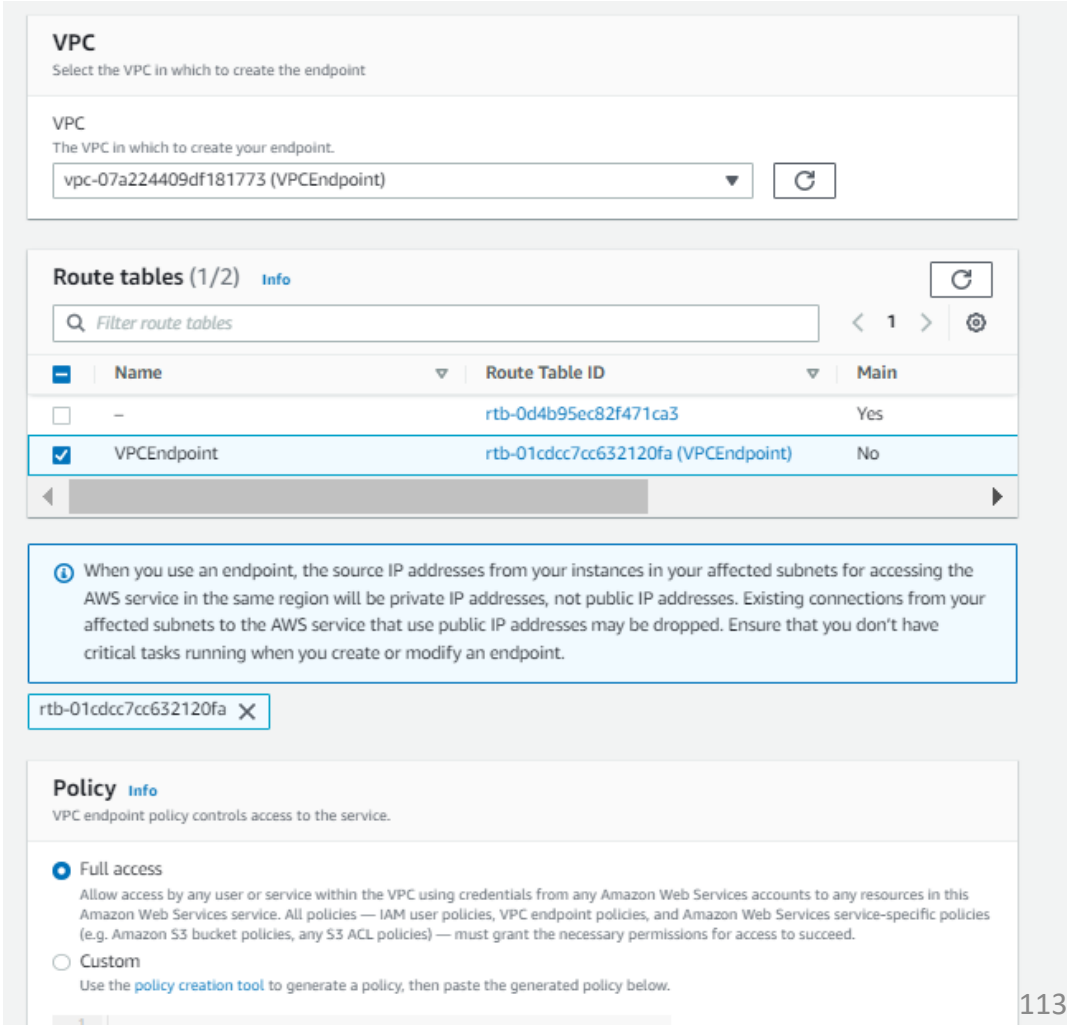
Download Object from S3

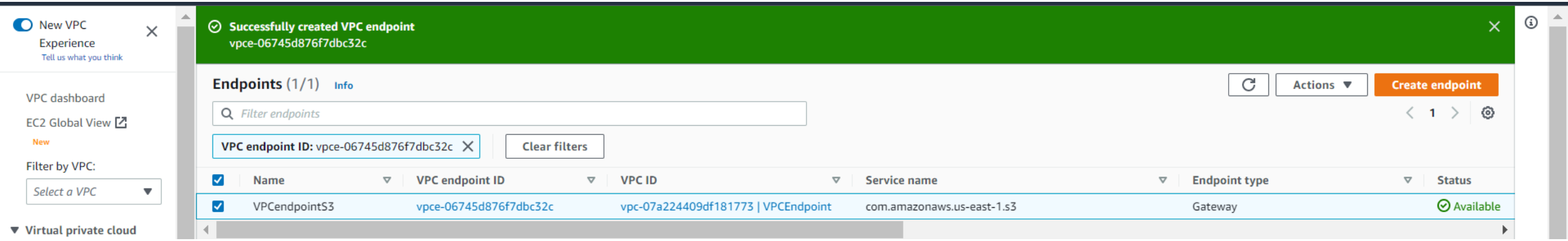
- Download is successful through the internet.

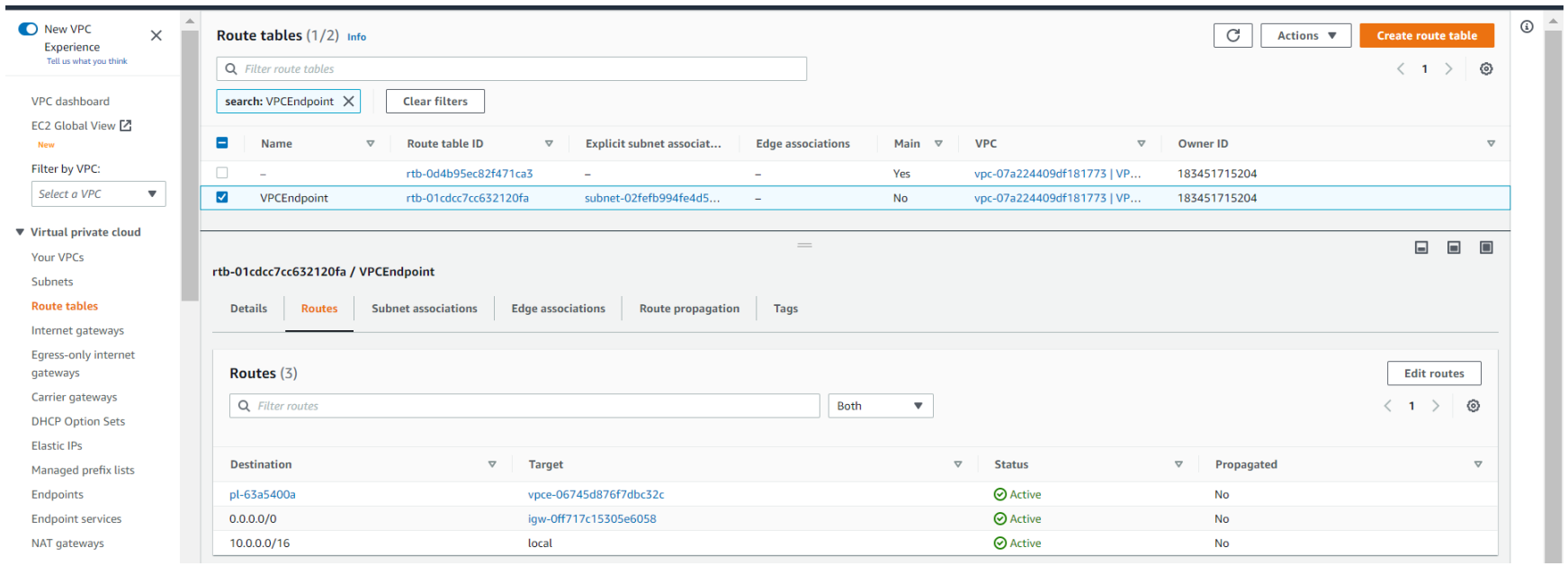
```
ubuntu@ip-10-0-0-235:~$ wget https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
--2022-08-03 01:02:48--  https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
Resolving santoso1demo.s3.amazonaws.com (santoso1demo.s3.amazonaws.com)... 52.217.197.89
Connecting to santoso1demo.s3.amazonaws.com (santoso1demo.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







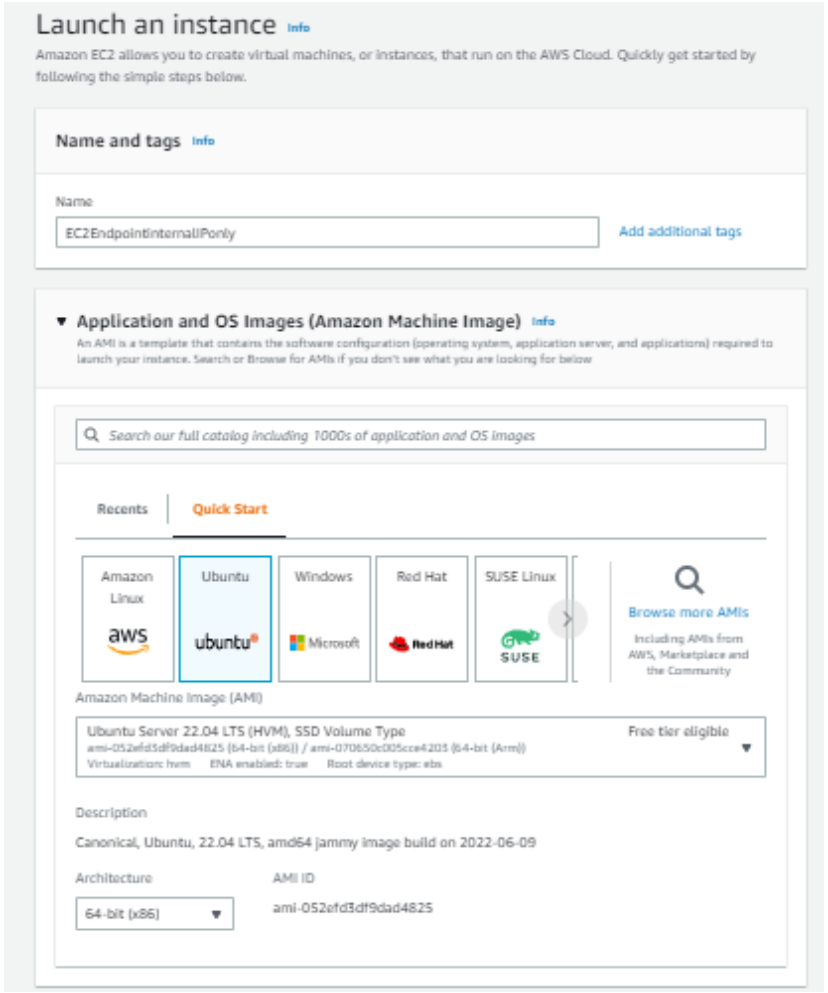


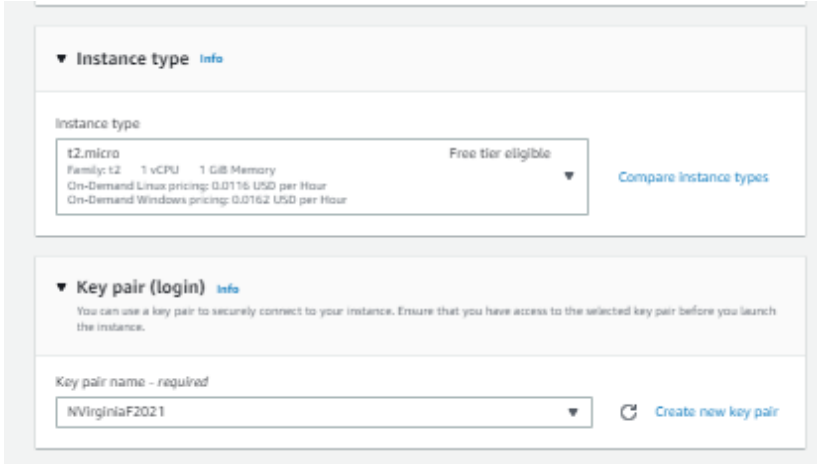
The Route Table is updated automatically.

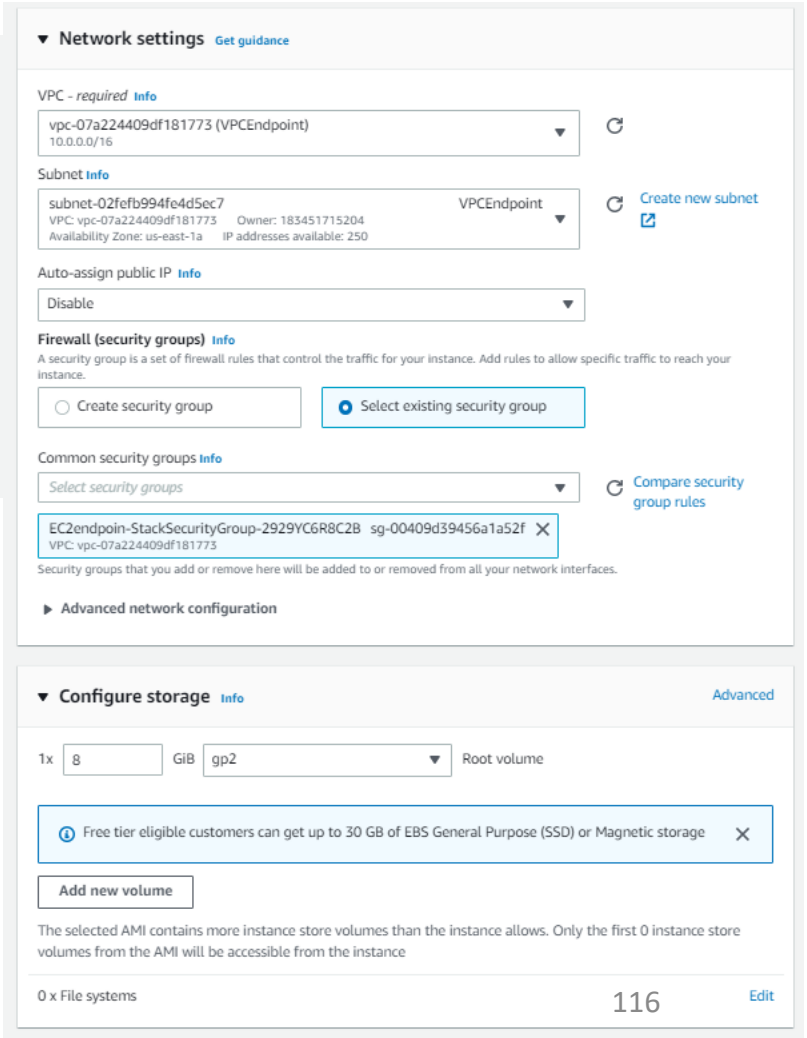
```
ubuntu@ip-10-0-0-235:~$ wget https://santosoldemo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
--2022-08-03 01:20:17--  https://santosoldemo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_0748.jpg
Resolving santosoldemo.s3.amazonaws.com (santosoldemo.s3.amazonaws.com)... 52.217.80.108
Connecting 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:~$
```

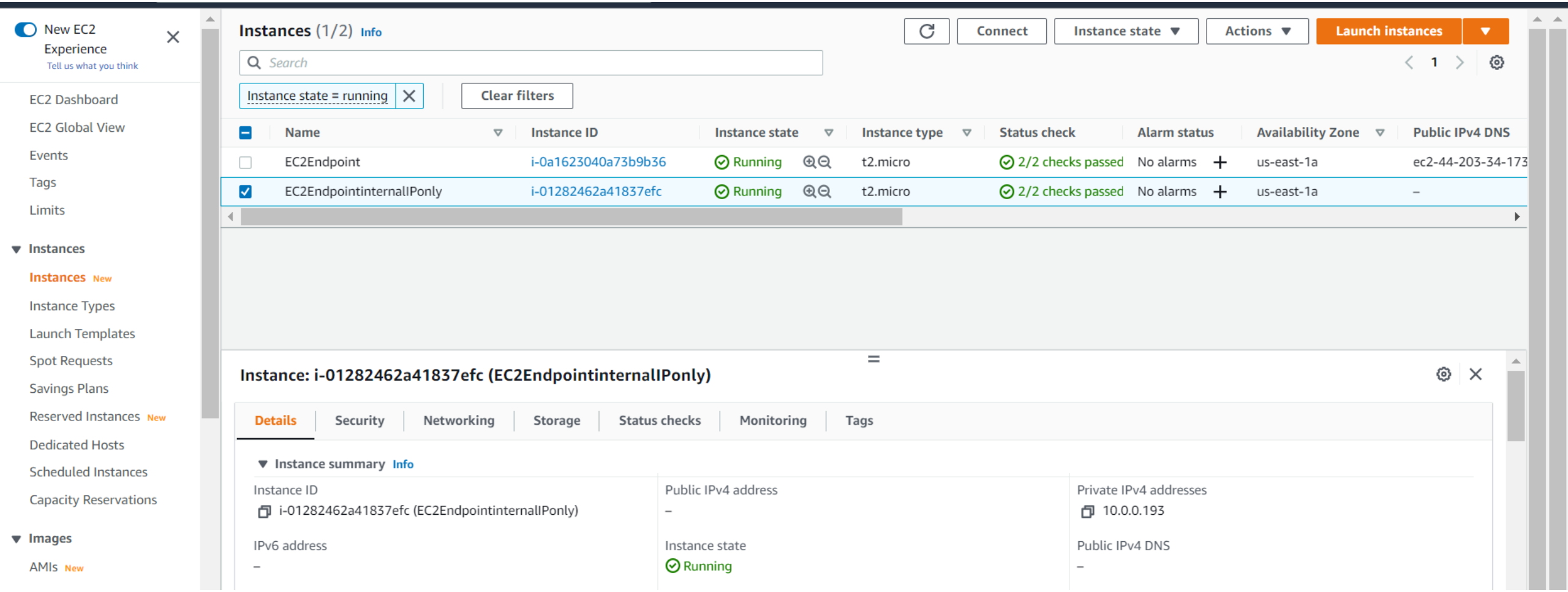
It is working without going through the internet

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








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
Clear filters
Connect
Instance state
Actions
Launch instances
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
EC2Endpoint
i-0a1623040a73b9b36
Running
t2.micro
2/2 checks passed
No alarms
us-east-1a
ec2-44-203-34-173
EC2EndpointinternalIPonly
i-01282462a41837efc
Running
t2.micro
2/2 checks passed
No alarms
us-east-1a
–
Instance: i-01282462a41837efc (EC2EndpointinternalIPonly)
Details
Security
Networking
Storage
Status checks
Monitoring
Tags
Instance summary Info
Instance ID
i-01282462a41837efc (EC2EndpointinternalIPonly)
Public IPv4 address
–
Private IPv4 addresses
10.0.0.193
IPv6 address
–
Instance state
Running
Public IPv4 DNS
–

SSH to EC2EndpointinternallPonly from EC2Endpoint (don’t forget the .pem file)

Then do the wget in EC2EndpointinternallPonly 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_Statue_0748.jpg
--2022-08-03 01:40:28--  https://santoso1demo.s3.amazonaws.com/folder1/2019_UTA_COB_Statue_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_Statue_0748.jpg’

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

2022-08-03 01:40:29 (101 MB/s) - ‘2019_UTA_COB_Statue_0748.jpg’ saved [15707507/15707507]

ubuntu@ip-10-0-0-193:~$ ls
2019_UTA_COB_Statue_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:~$
```

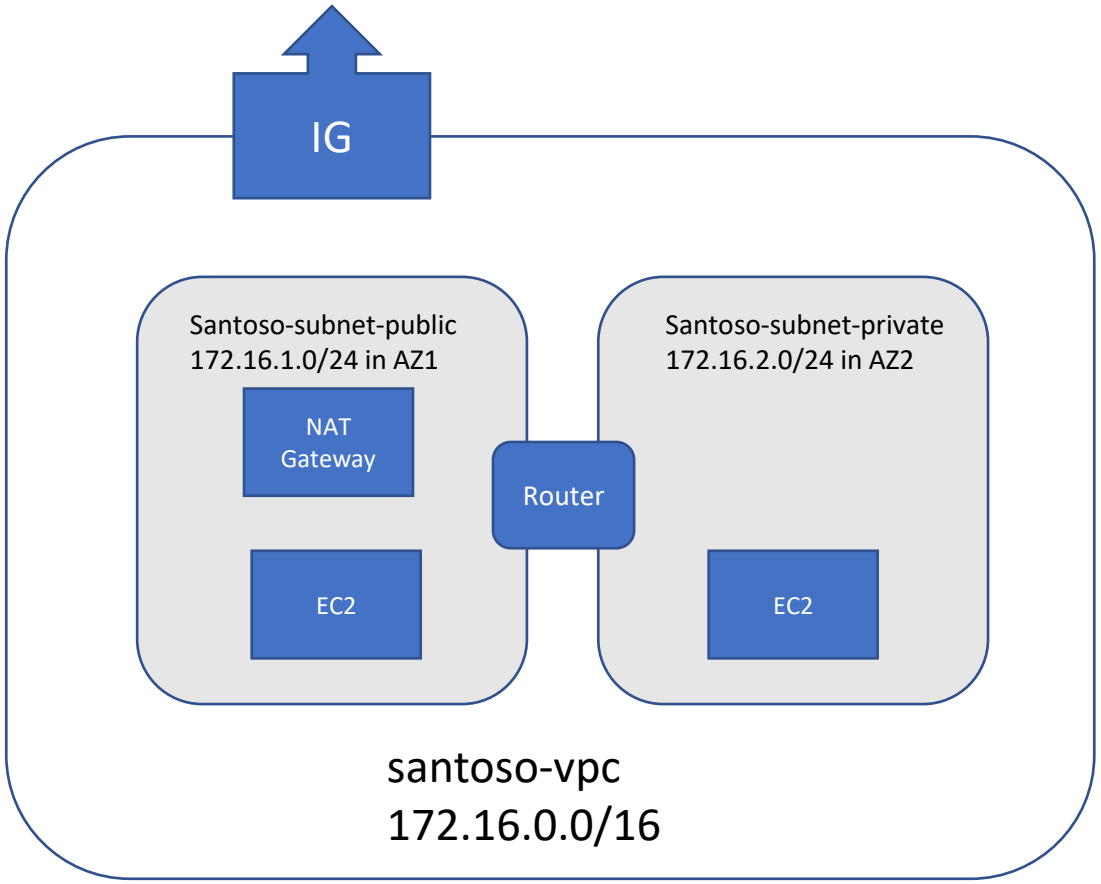
But the other EC2 with Public IP can go out of the internet.

```
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

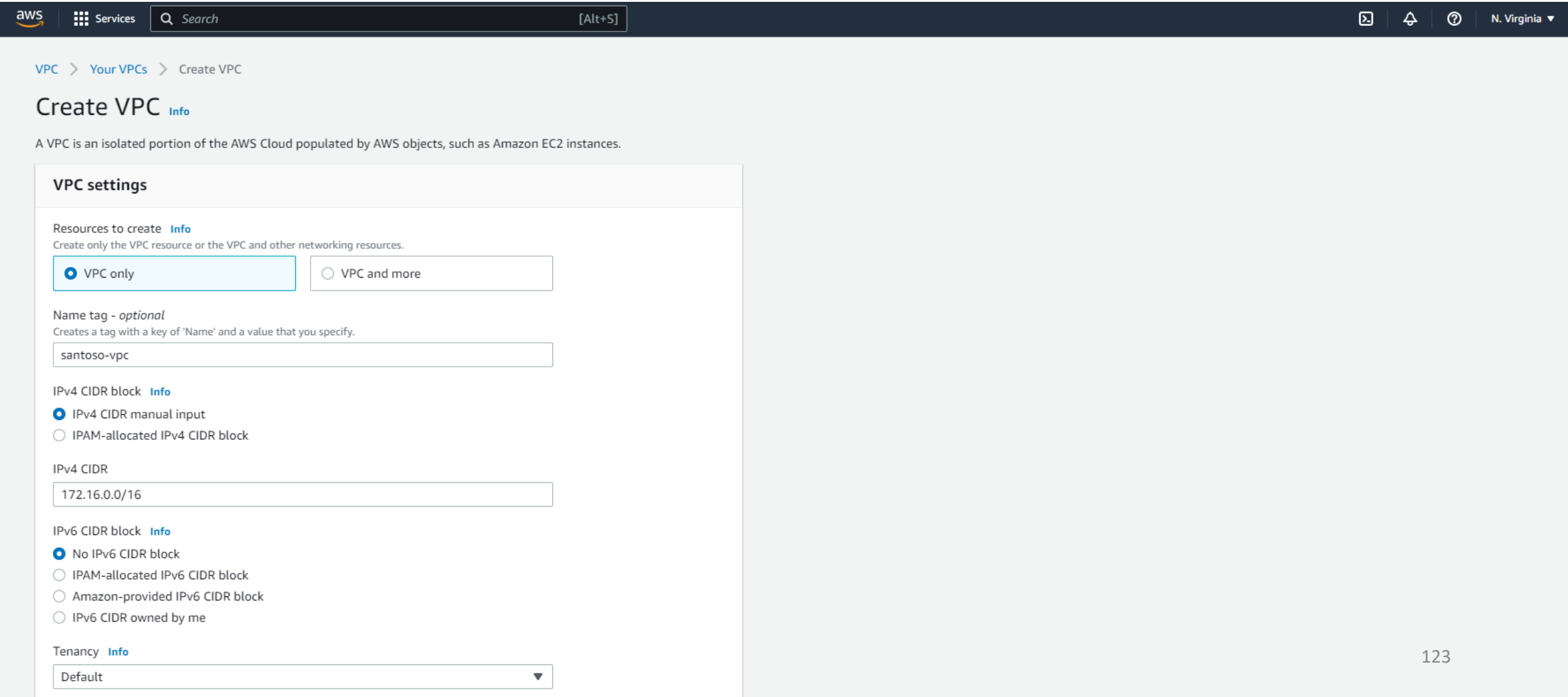
- VPC Name: santoso-vpc
- CIDR: 172.16.0.0/16
- Subnet 1: santoso-subnet-public
- Subnet 1 CIDR: 172.16.1.0/24
- Subnet 2: santoso-subnet-private
- Subnet 2 CIDR: 172.16.2.0/24



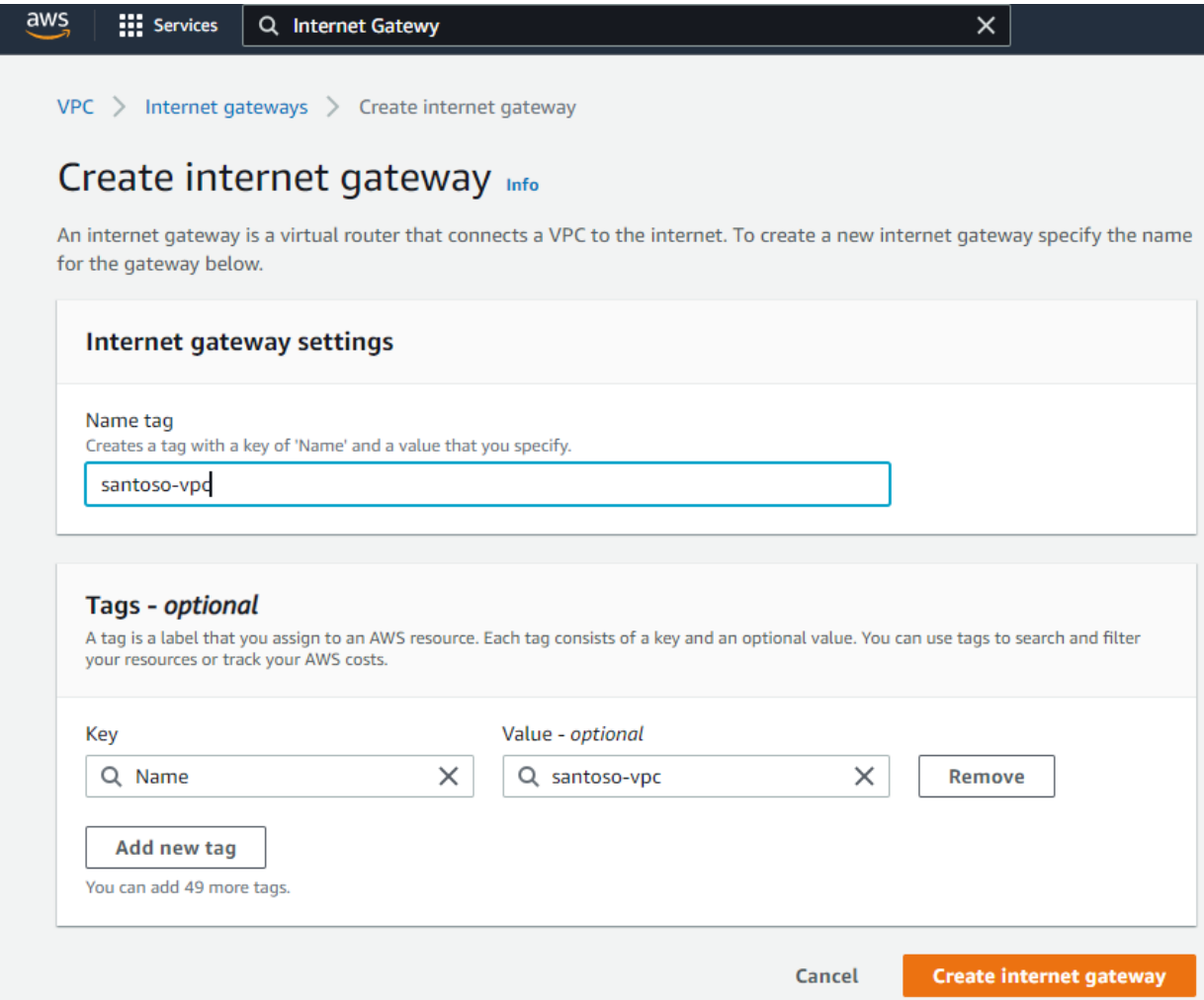
Steps

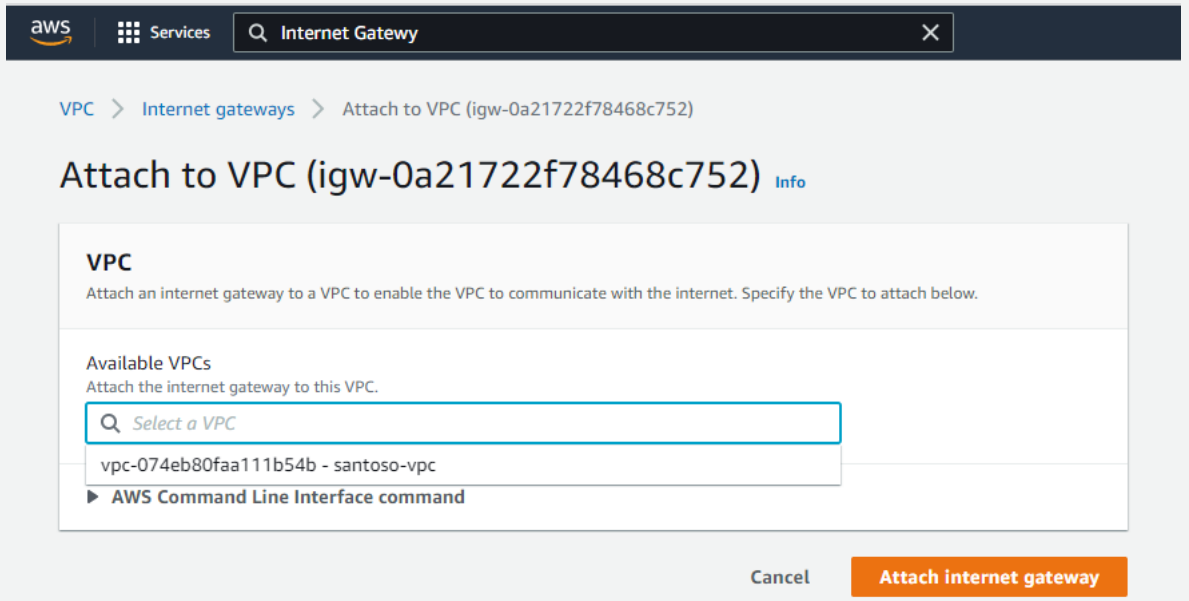
- Create VPC
- Create IG and attach to the VPC
- Create Public Subnet in AZ1
- Create Route Table for the public subnet
- Create EC2 in the public subnet – allow ssh and ping from anywhere
- Create Private Subnet in AZ2
- Create Route Table for the private subnet
- Create EC2 in the private subnet –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

Create VPC

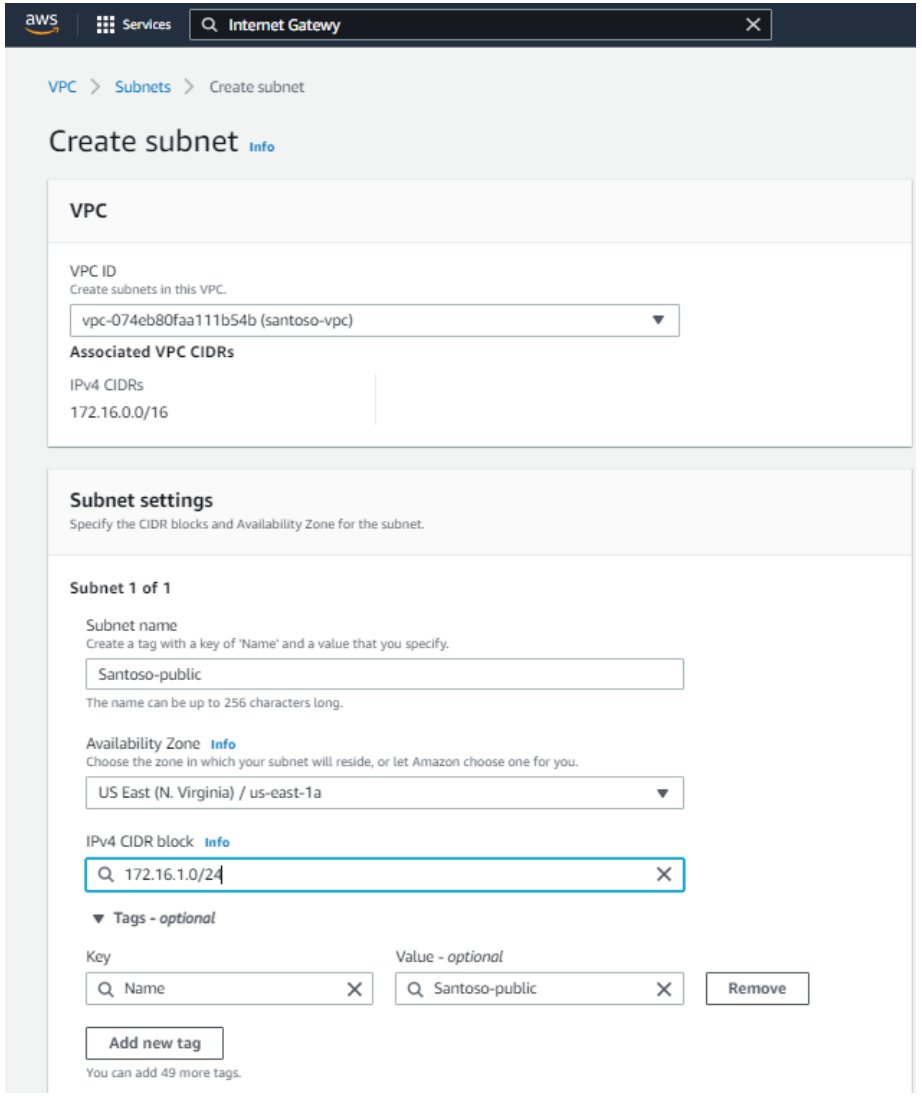

aws
Services
Search
[Alt+S]
N. Virginia
VPC > Your VPCs > Create VPC
Create VPC Info
A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.
VPC settings
Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
VPC 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

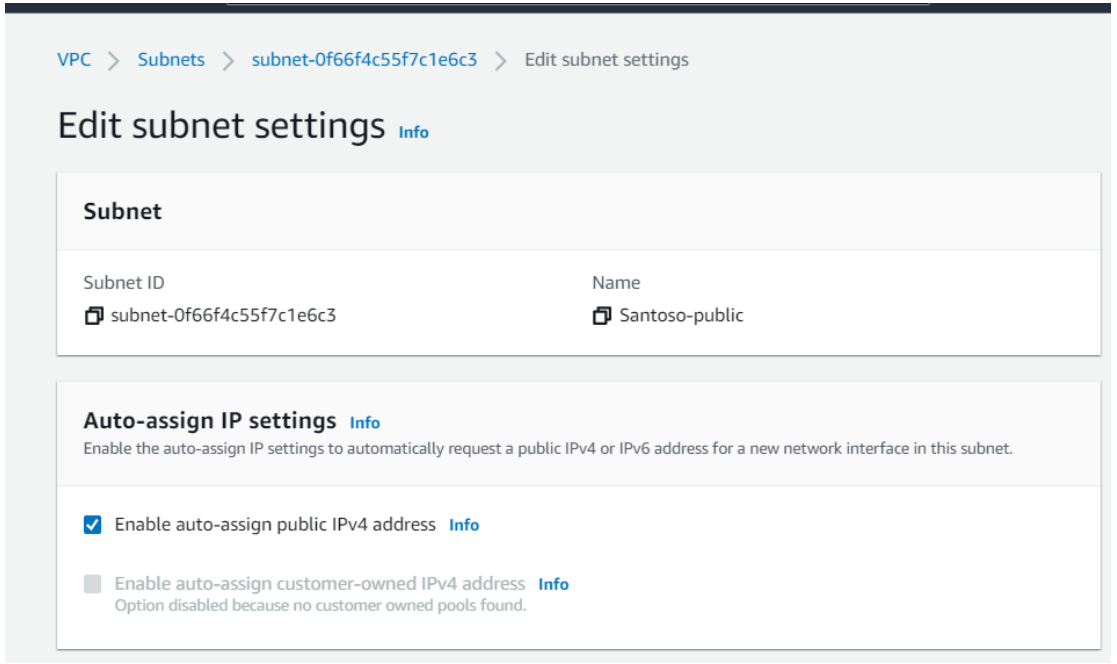
Create IG and Attach to the VPC



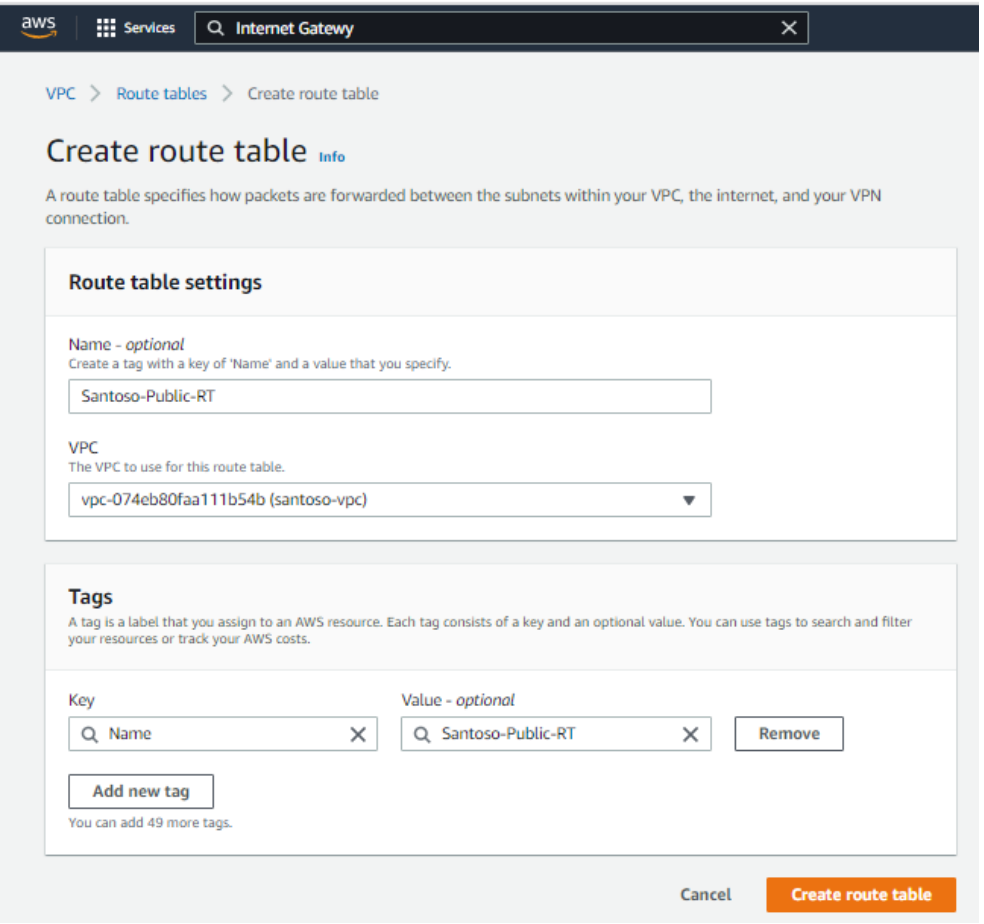


Create Public Subnet

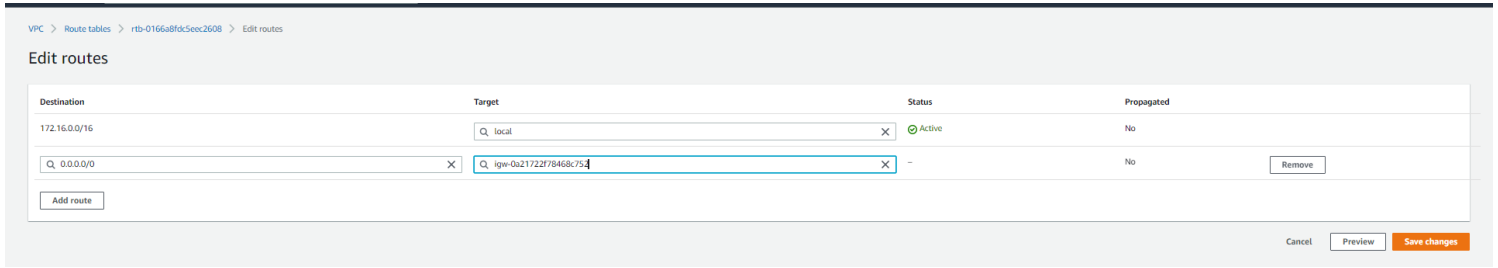




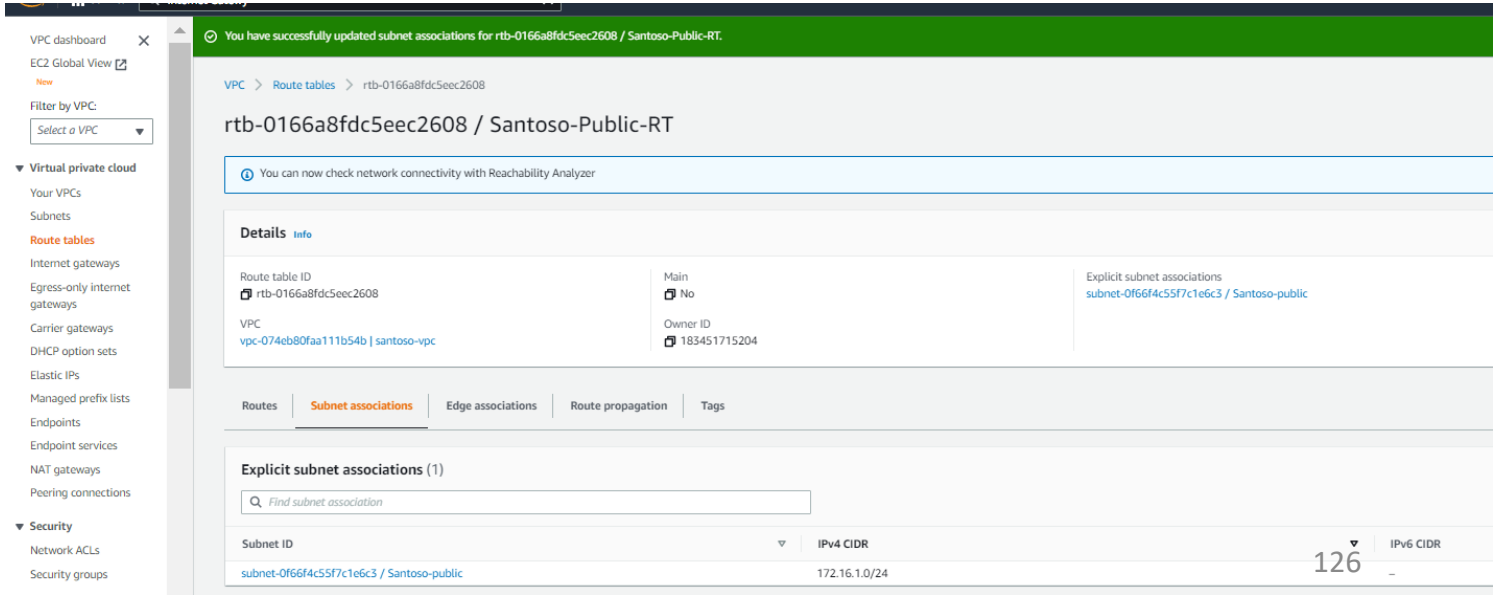
Create Route Table for the Public Subnet



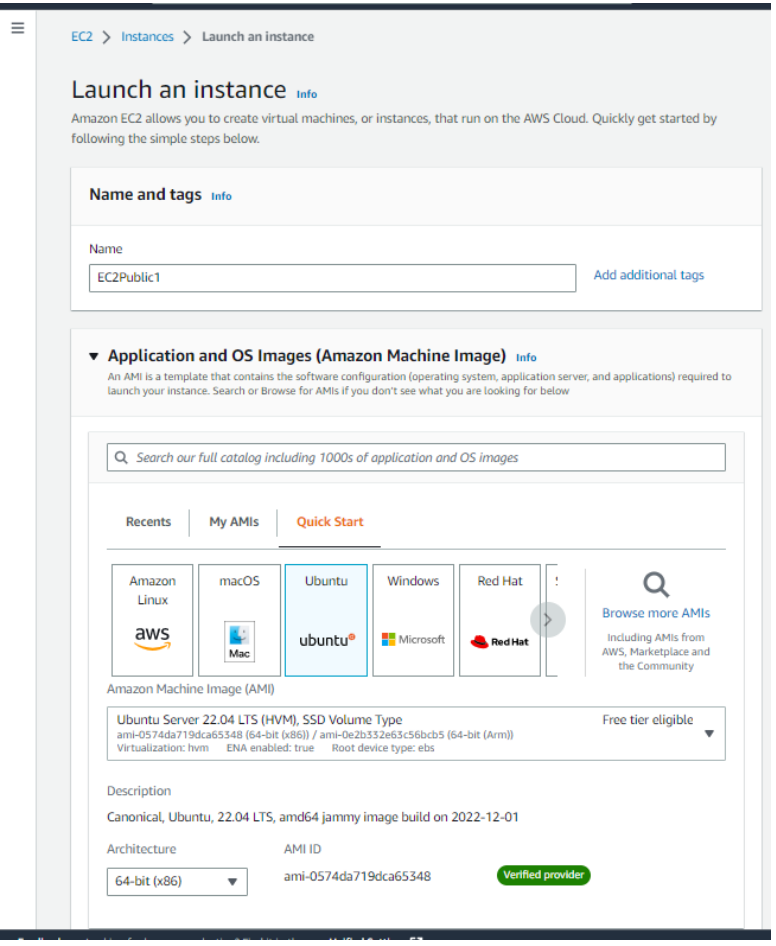
Add route to IG

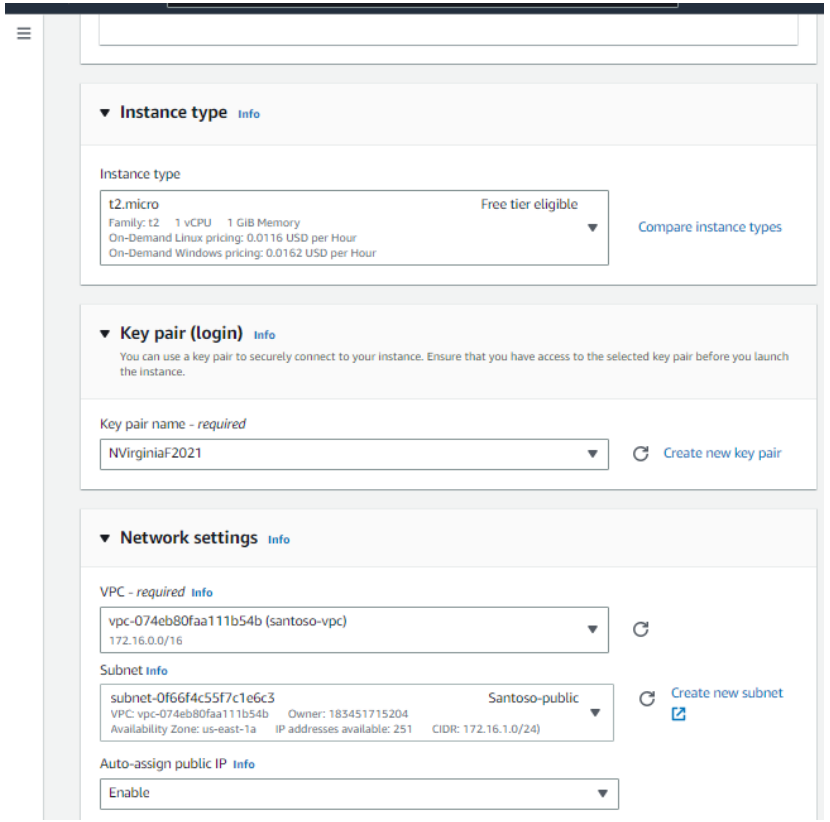


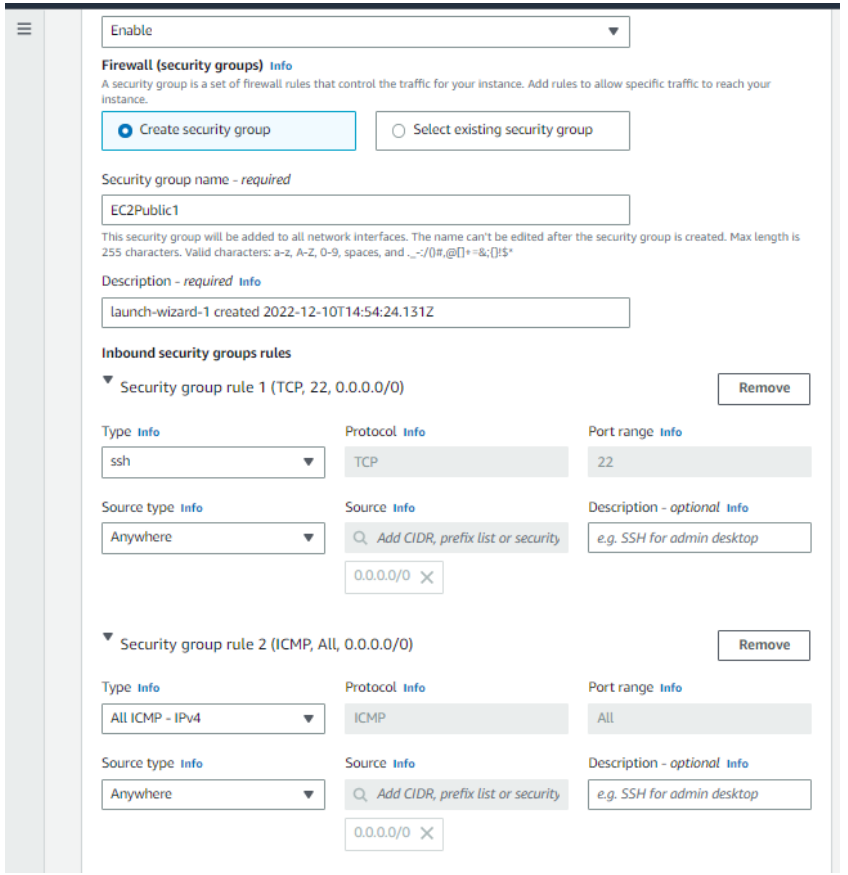
Associate to the public subnet



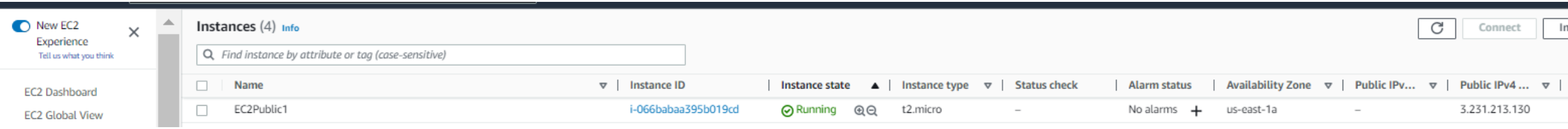
Create EC2 in the public subnet – allow ssh and ping from anywhere





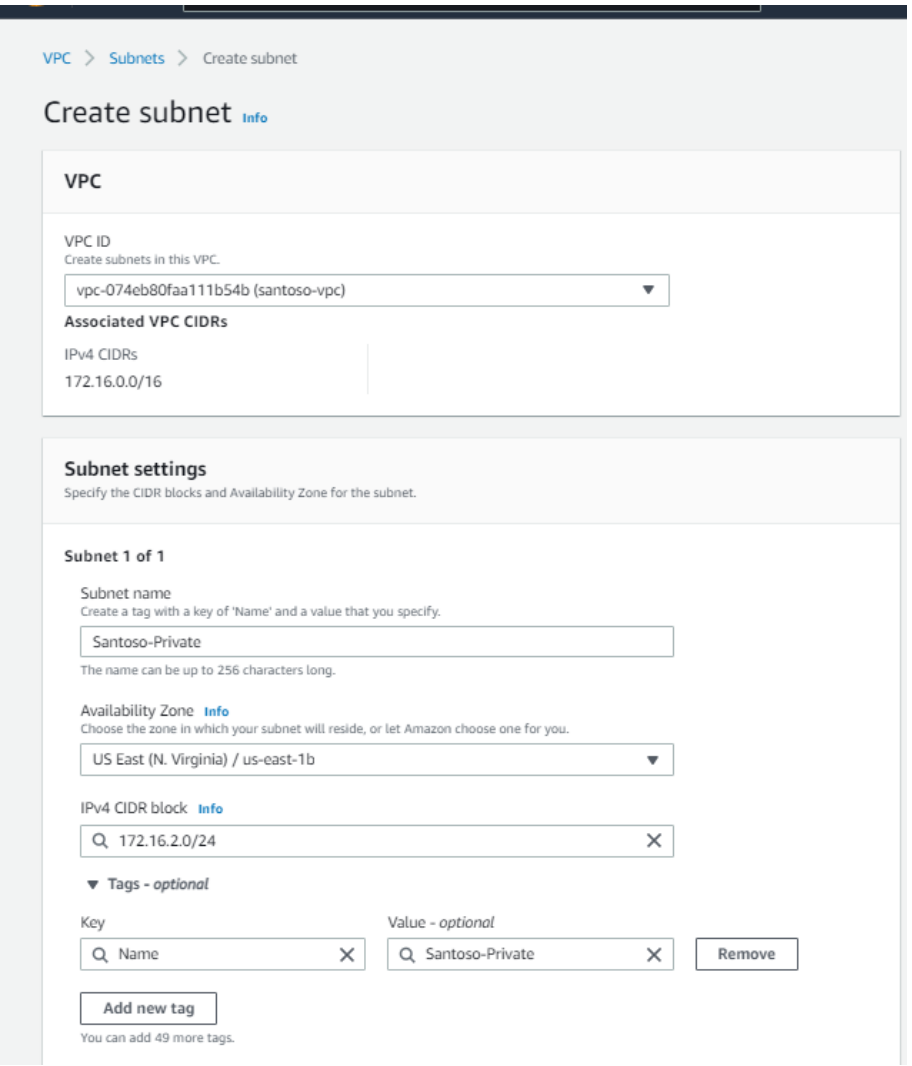


Test EC2Public1

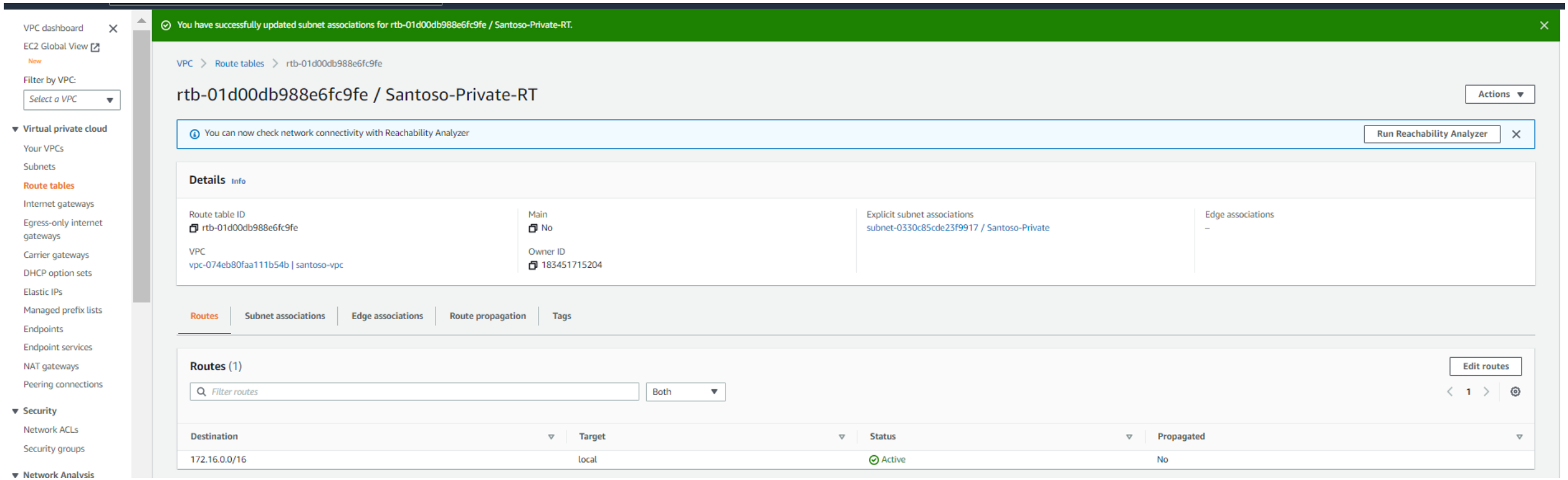



```
C:\Windows\System32>ping 3.231.213.130

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>
```

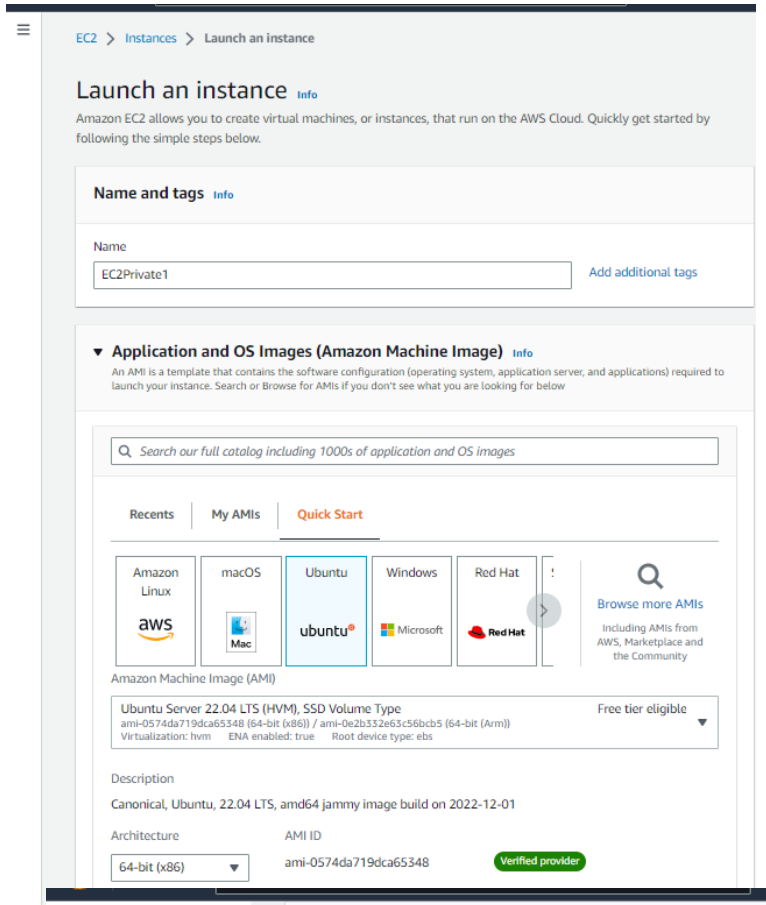
Create Private Subnet and the route table

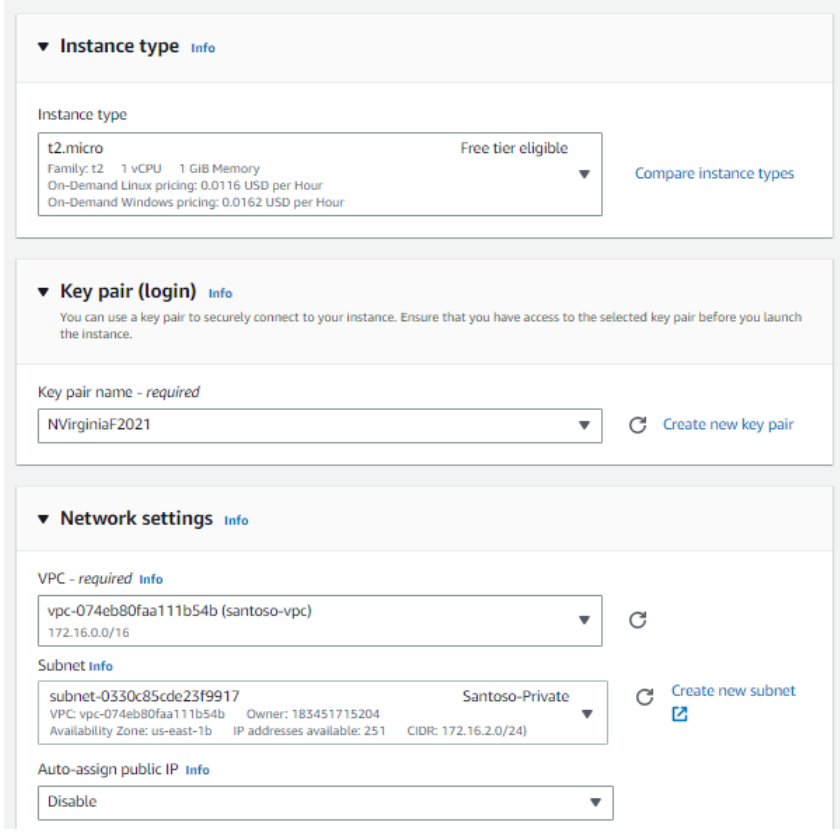


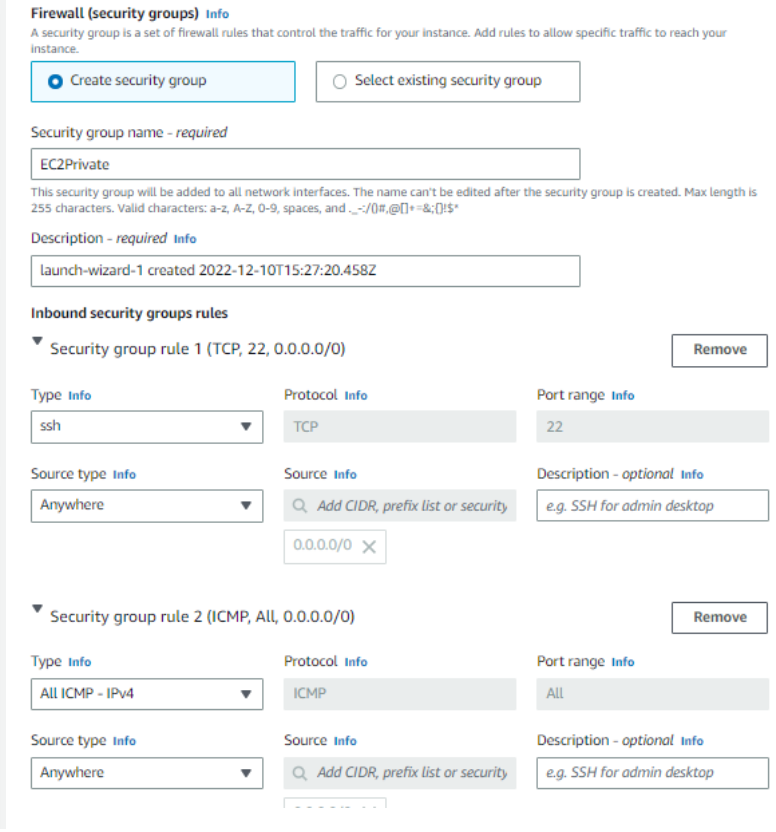
Create The route table

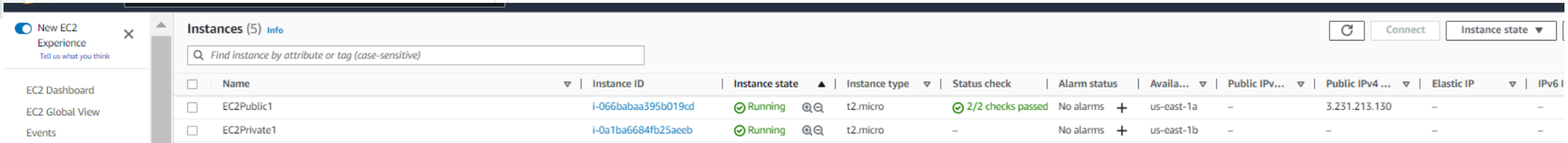


Create EC2Private









SSH to public EC2 and ping private EC2

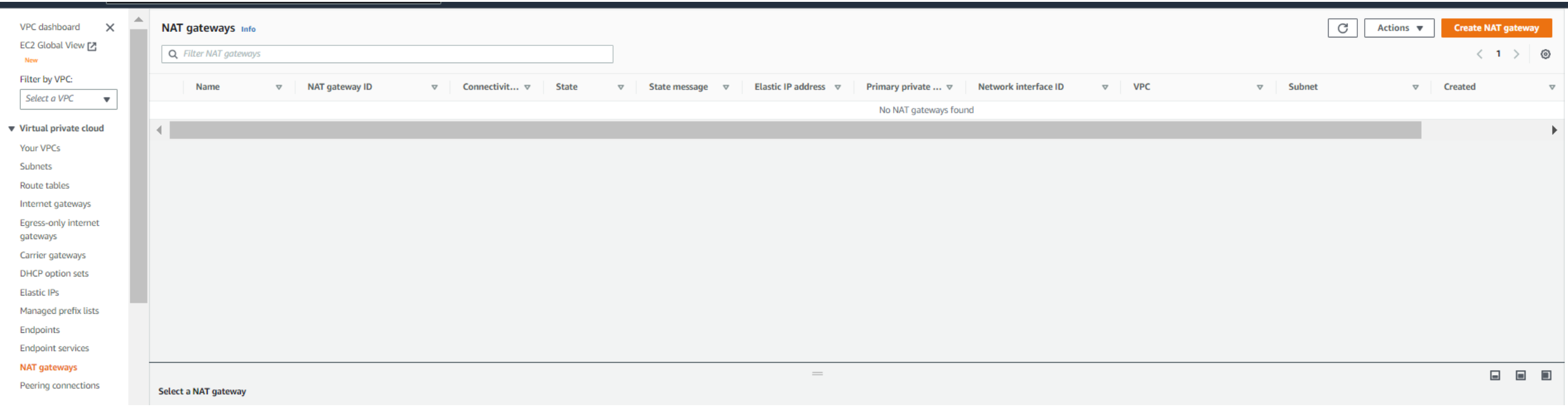
```
ubuntu@ip-172-16-1-68:~$ #this is public EC2
ubuntu@ip-172-16-1-68:~$ #ping to private EC2
ubuntu@ip-172-16-1-68:~$ ping -c3 172.16.2.61
PING 172.16.2.61 (172.16.2.61) 56(84) bytes of data.
64 bytes from 172.16.2.61: icmp_seq=1 ttl=64 time=1.18 ms
64 bytes from 172.16.2.61: icmp_seq=2 ttl=64 time=0.761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

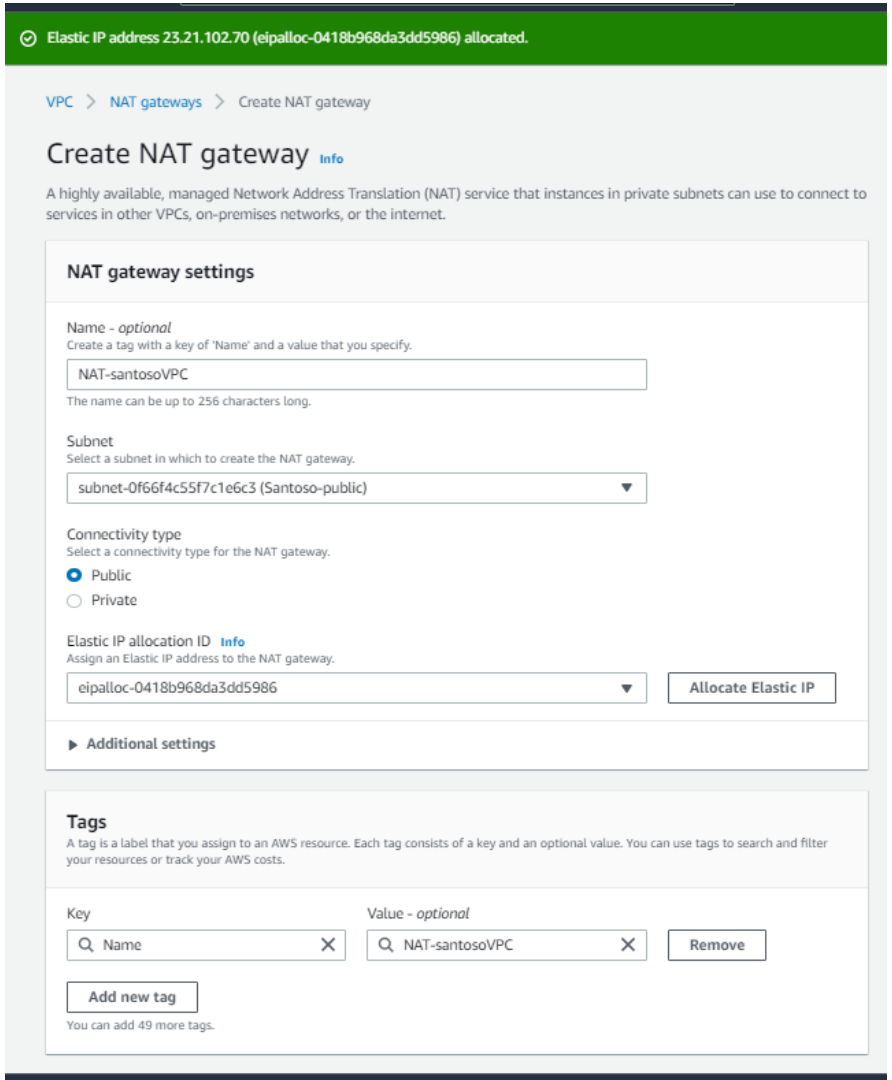
```
ubuntu@ip-172-16-1-68:~$ chmod 700 NVirginiaF2021.pem
ubuntu@ip-172-16-1-68:~$ ssh -i "NVirginiaF2021.pem" ubuntu@172.16.2.61
```

```
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:~$
```

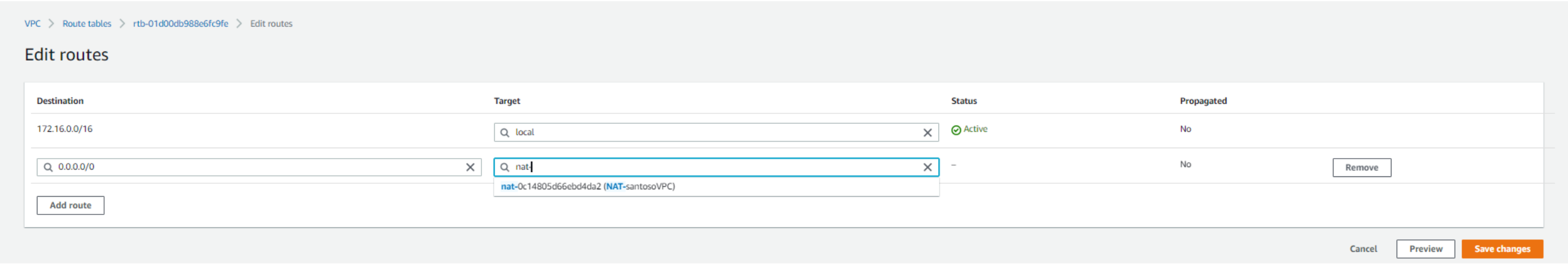
Create NAT Gateway in the public subnet



Must be in Public subnet and must allocate Elastic IP



Update private subnet Routing Table



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:~$
```

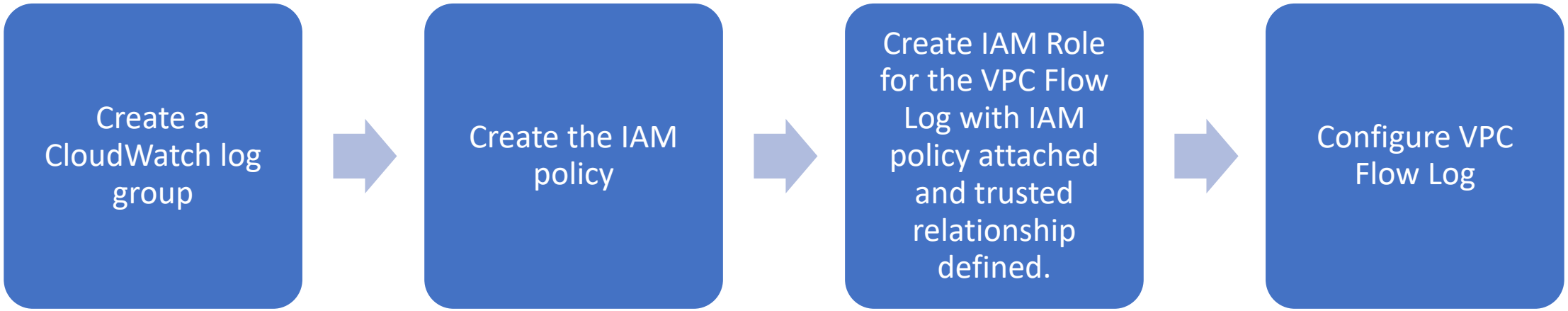
```
ubuntu@ip-172-16-2-61:~$ #ping publicEC2 using its public IP
ubuntu@ip-172-16-2-61:~$ ping -c3 3.231.213.130
PING 3.231.213.130 (3.231.213.130) 56(84) bytes of data.
64 bytes from 3.231.213.130: icmp_seq=1 ttl=62 time=1.35 ms
64 bytes from 3.231.213.130: icmp_seq=2 ttl=62 time=0.994 ms
64 bytes from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



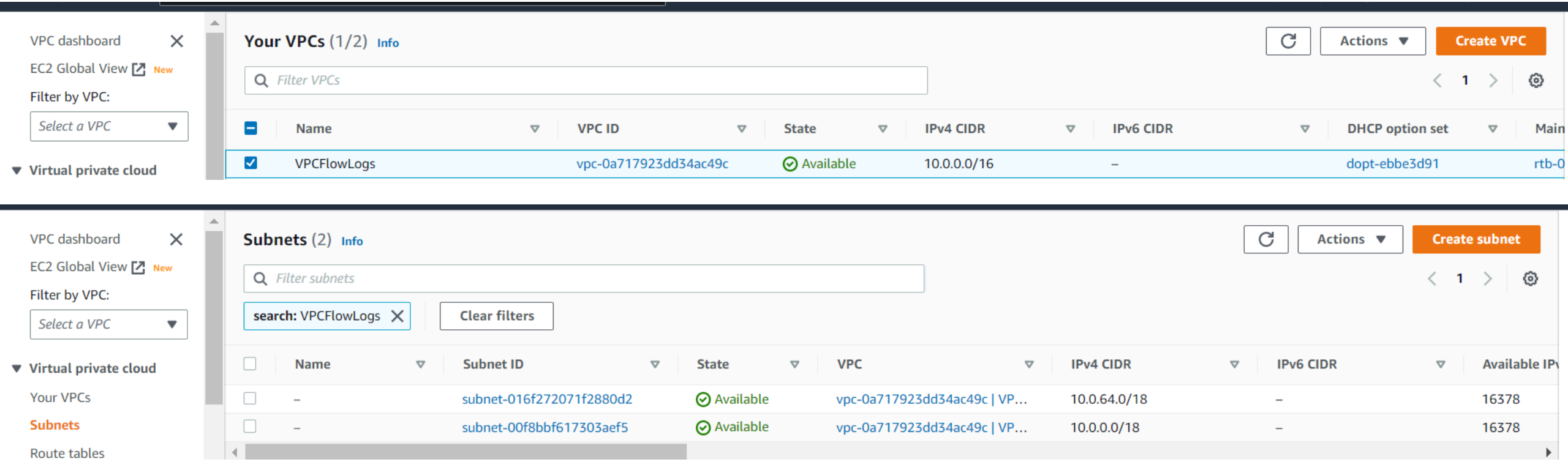
https://docs.aws.amazon.com/vpc/latest/tgw/flow-logs-cwl.html

Demo shortcut

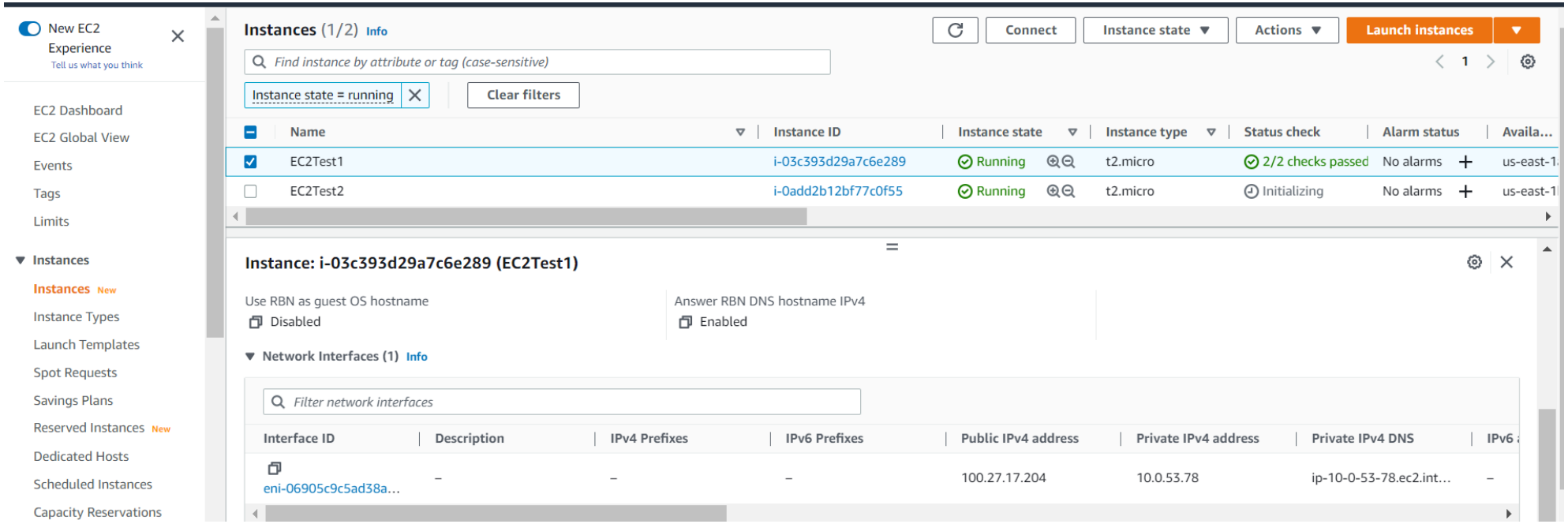
- We have not talked about IAM and the process of creating IAM role and policy is a bit involved.
- Since the purpose of the demo is to show a way to see IP traffic going to and from network interfaces in your VPC and to speed up the demo, I have created and will use the below in the demo.
  - IAM Policy: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

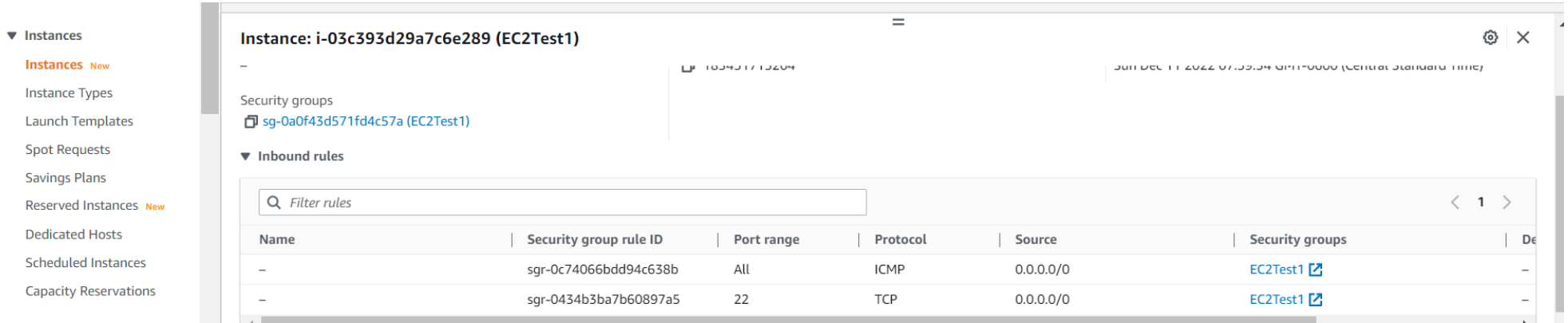
Create a VPC + 2 subnets + 2 EC2s

Create a VPC with 2 subnets

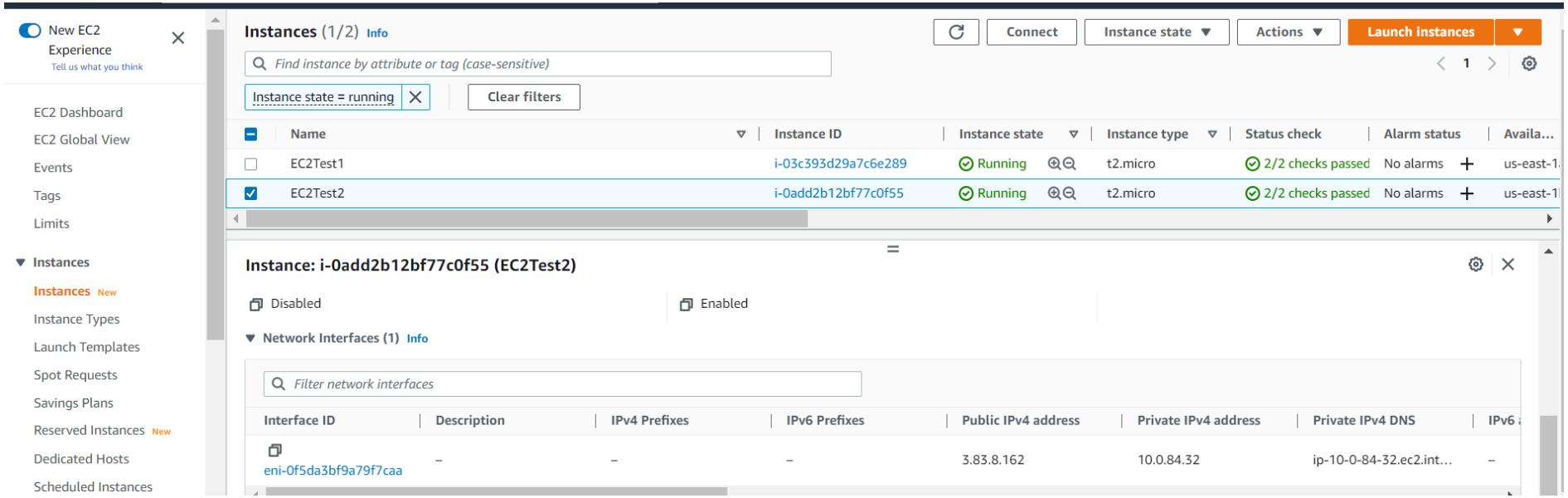


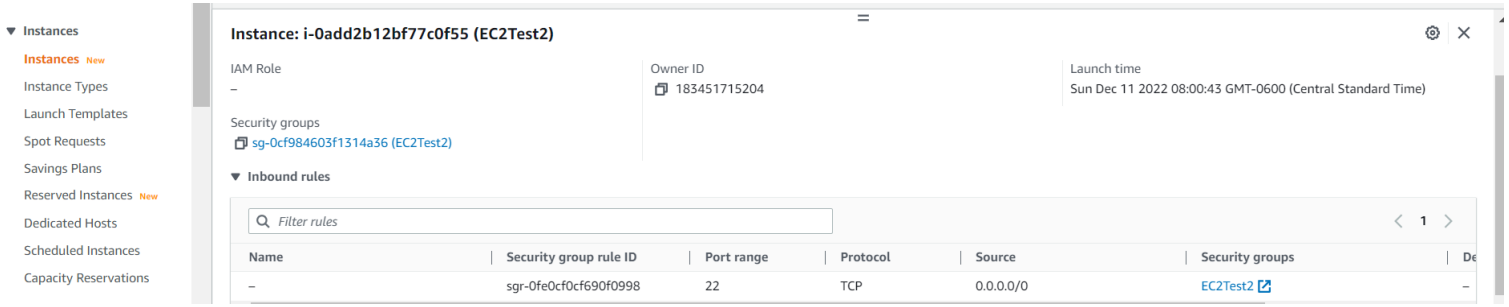
Spin up an EC2 in each subnet: 1st EC2





Spin up an EC2 in each subnet: 2nd EC2

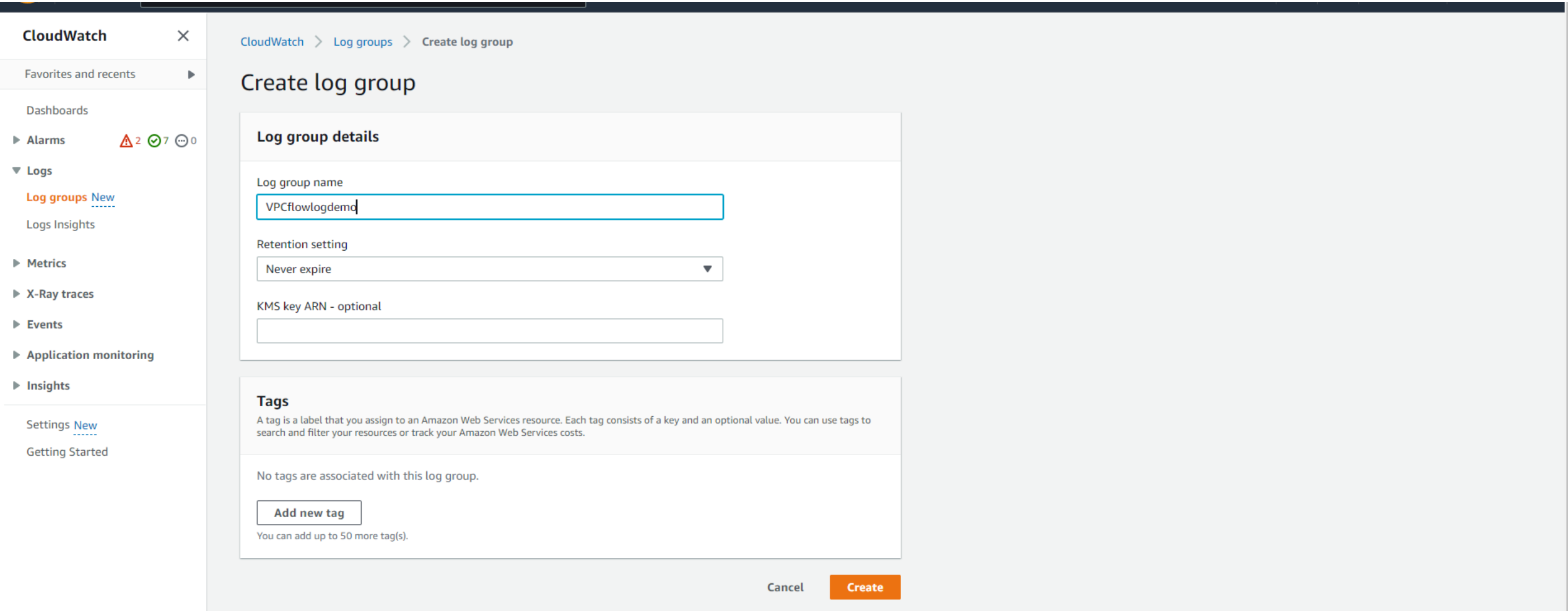




Create VPC Flow Log

Create CloudWatch group

Create CloudWatch Log



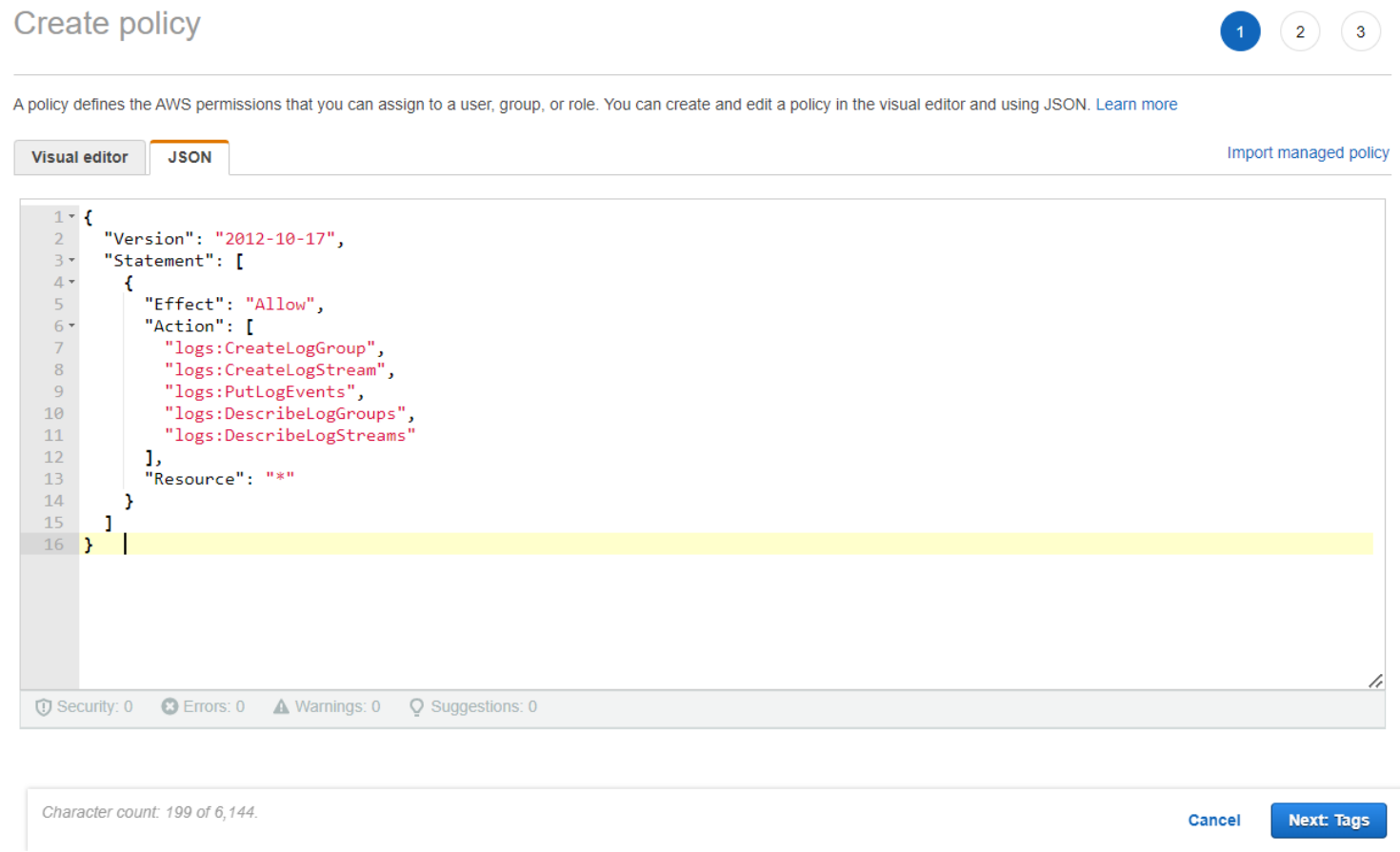
Create an IAM Role for the VPC Flow log

To create an IAM role for flow logs

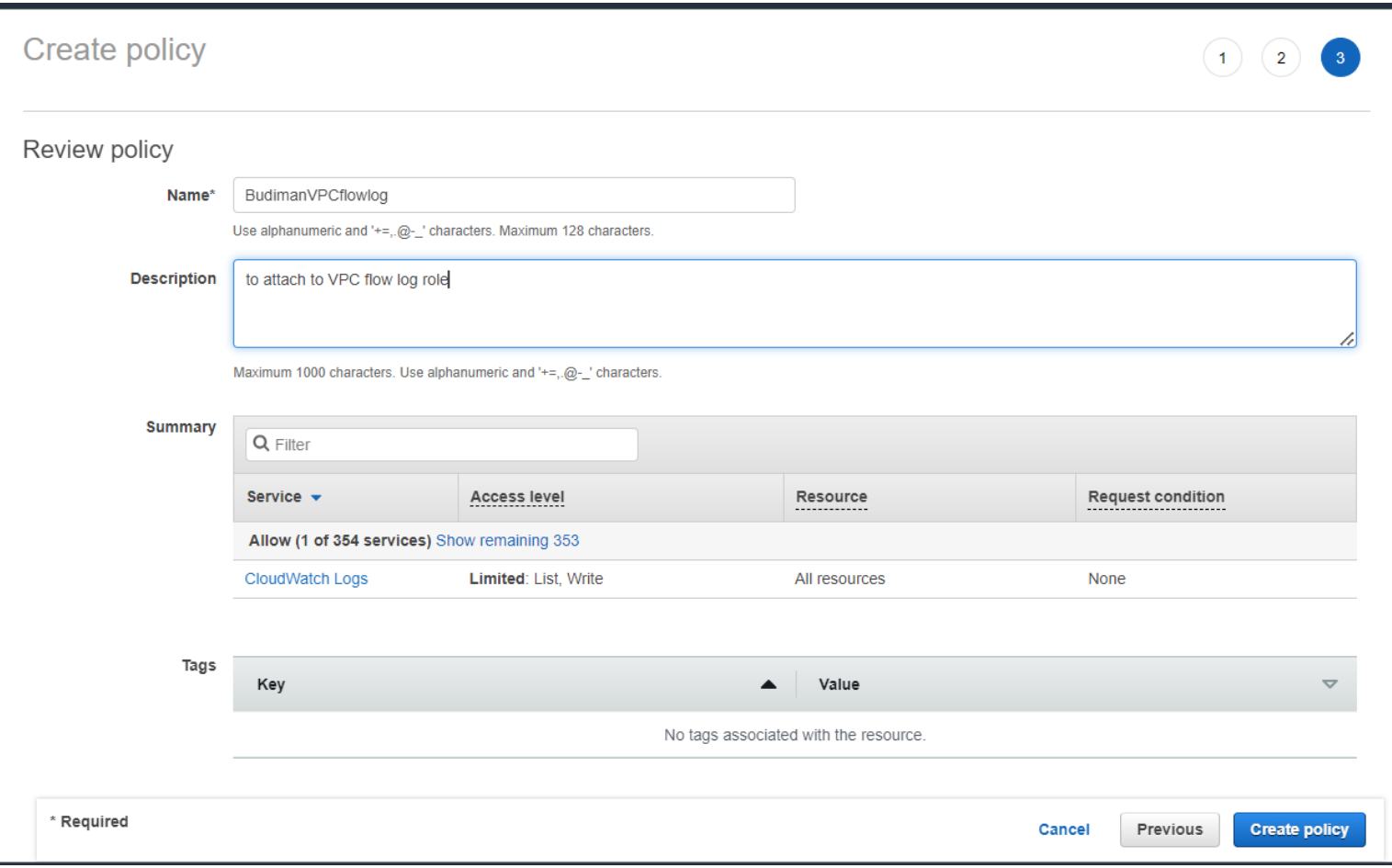
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Roles**, **Create role**.
3. For **Select type of trusted entity**, choose **AWS service**. For **Use case**, choose **EC2**. Choose **Next: Permissions**.
4. On the **Attach permissions policies** page, 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

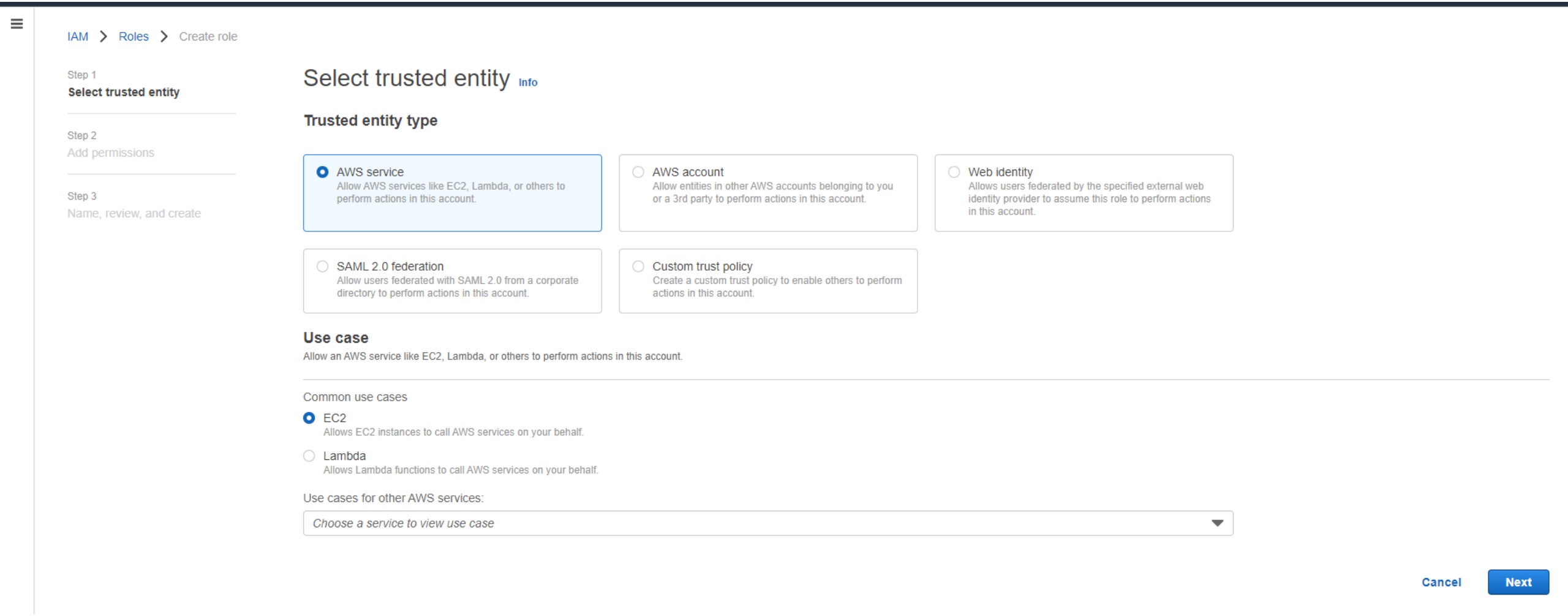
Create IAM Policy



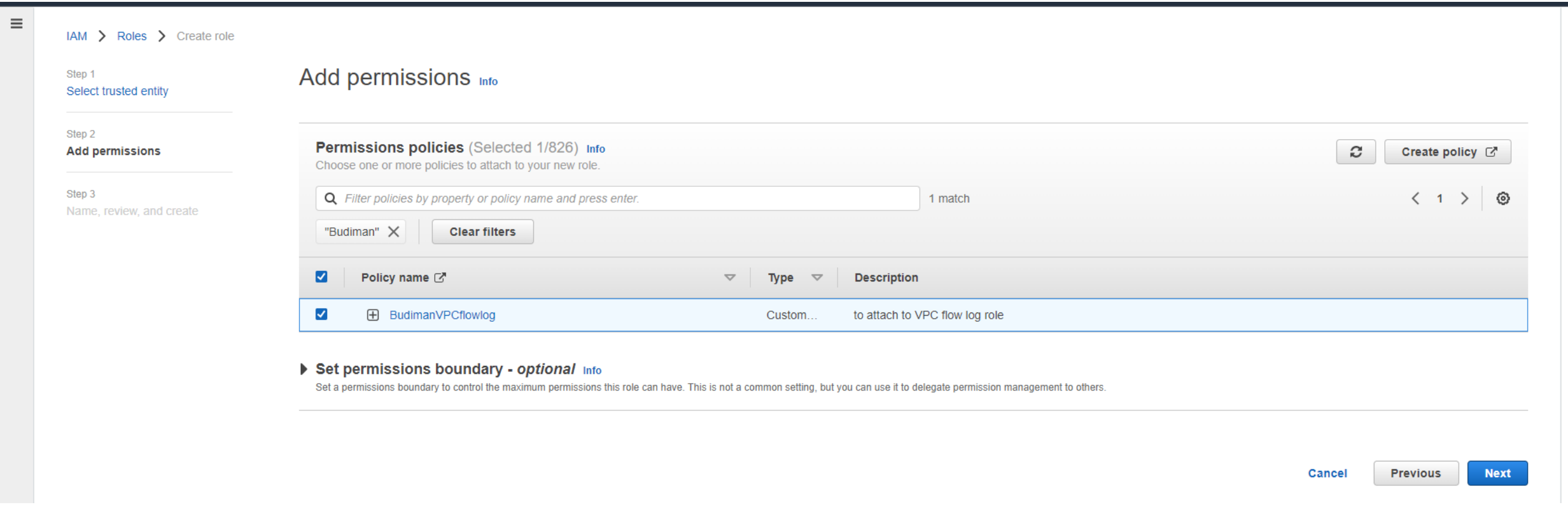
Create Policy

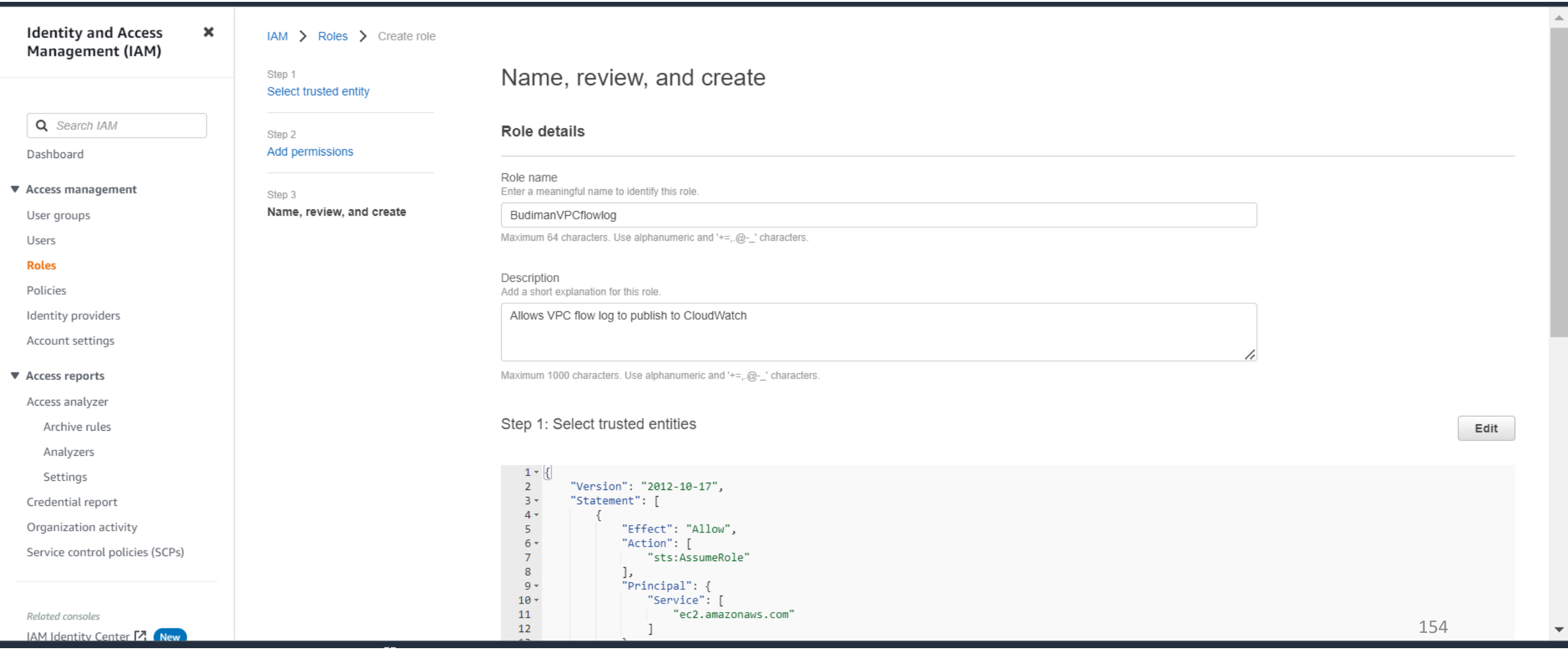


Create Role

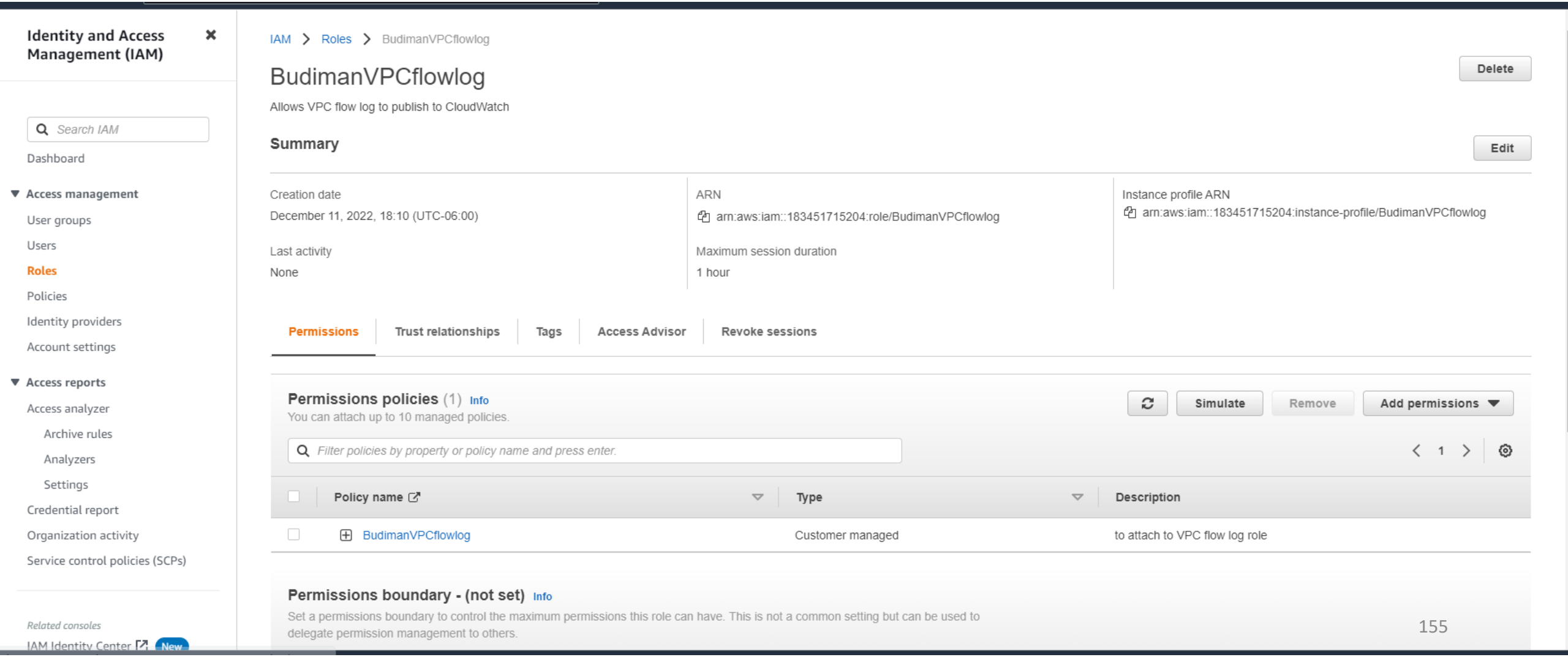

IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity Info
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts 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

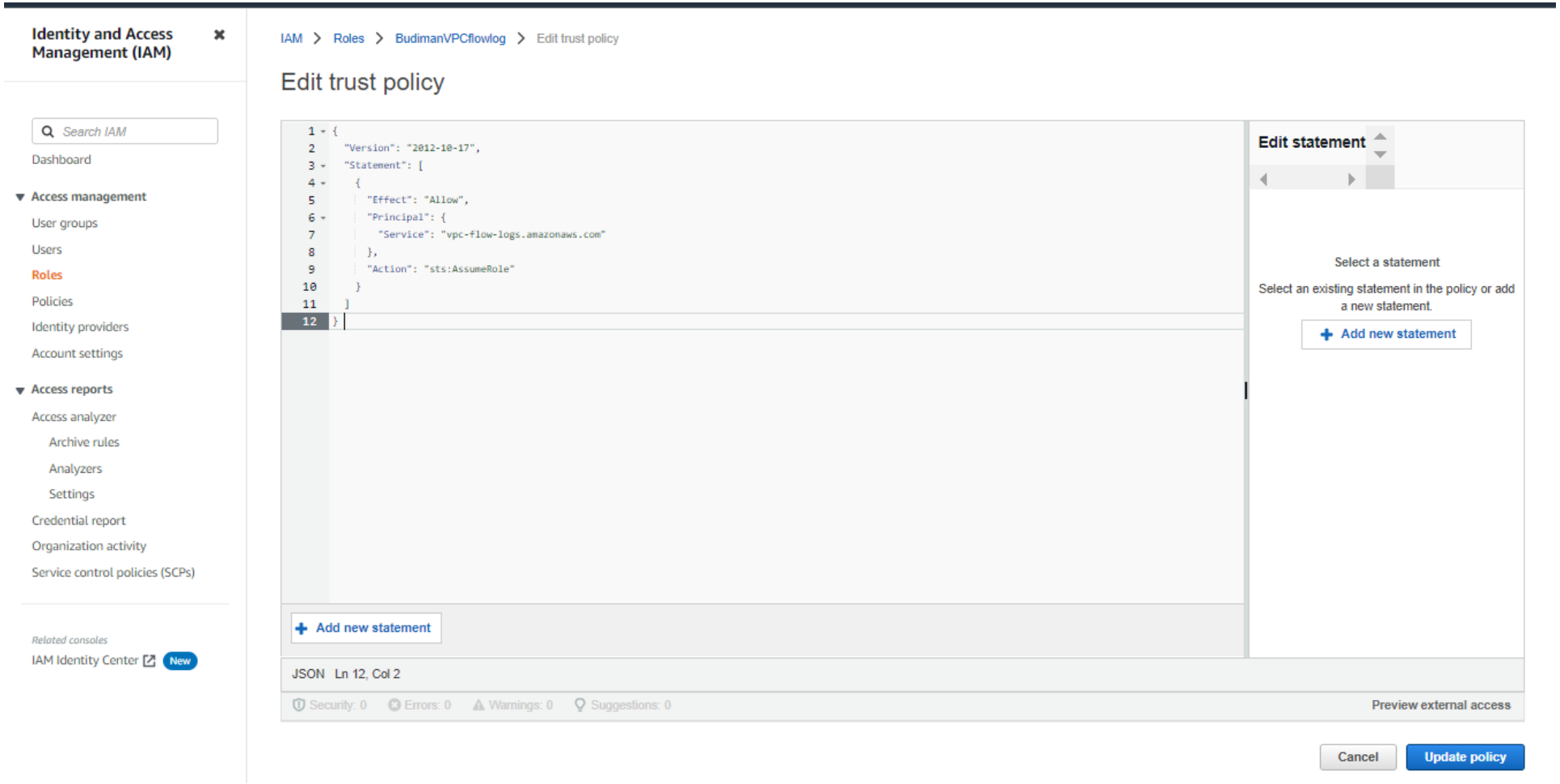

IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (Selected 1/826) Info
Choose one or more policies to attach to your new role.
Create policy
Filter policies by property or policy name and press enter.
1 match
1
"Budiman"
Clear filters
Policy name
Type
Description
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

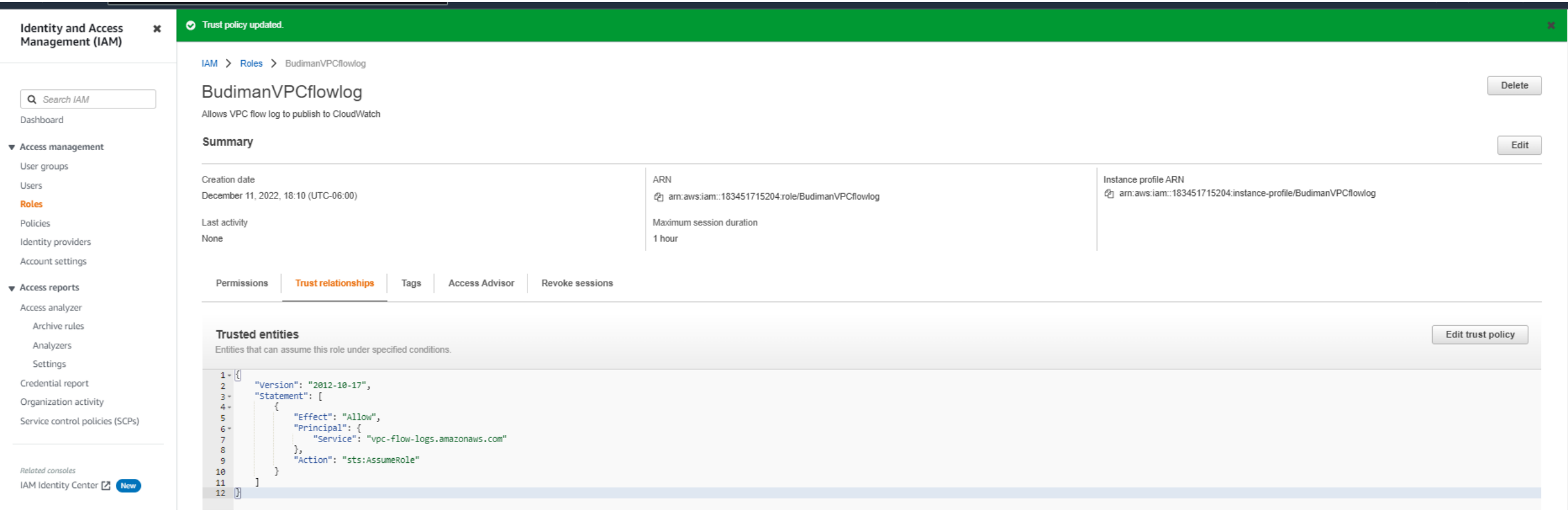

Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
Related consoles
IAM 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.
BudimanVPCflowlog
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows VPC flow log to publish to CloudWatch
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Step 1: Select trusted entities
Edit
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
"ec2.amazonaws.com"
]

IAM Role is created, now edit the Trusted Relationship

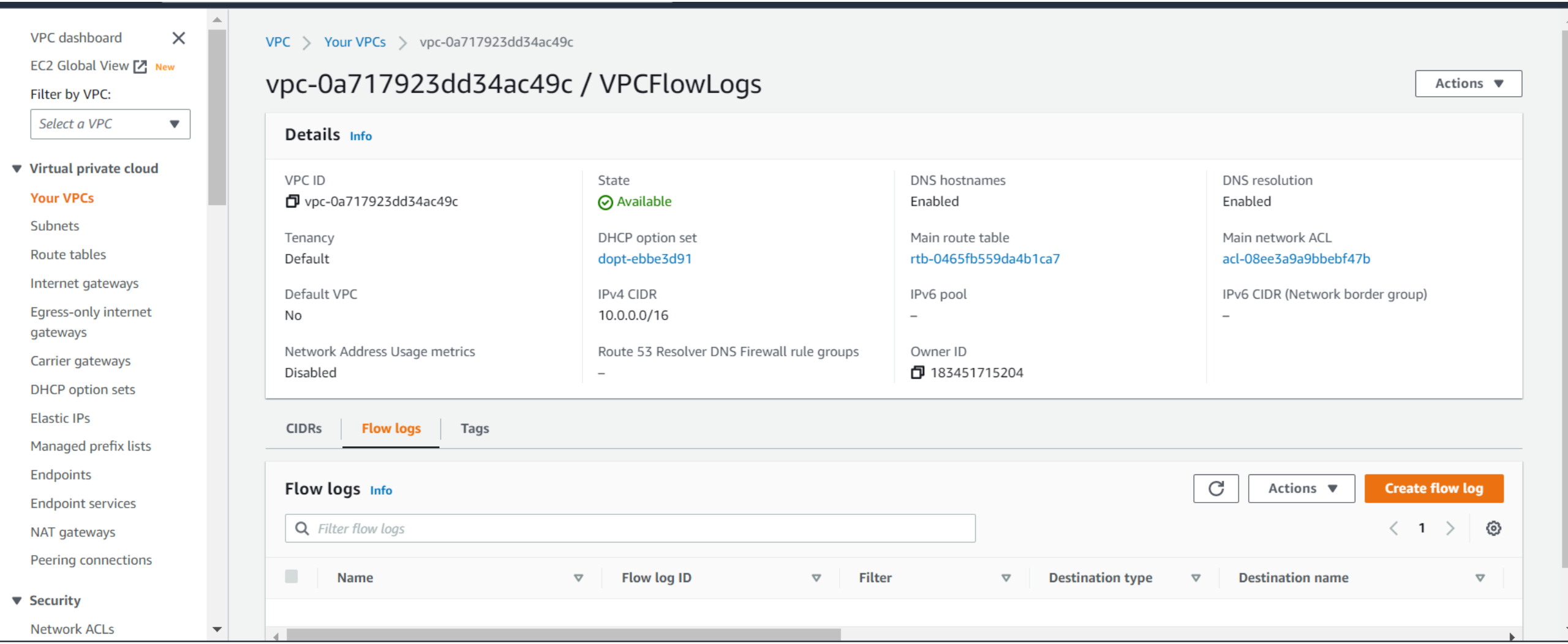


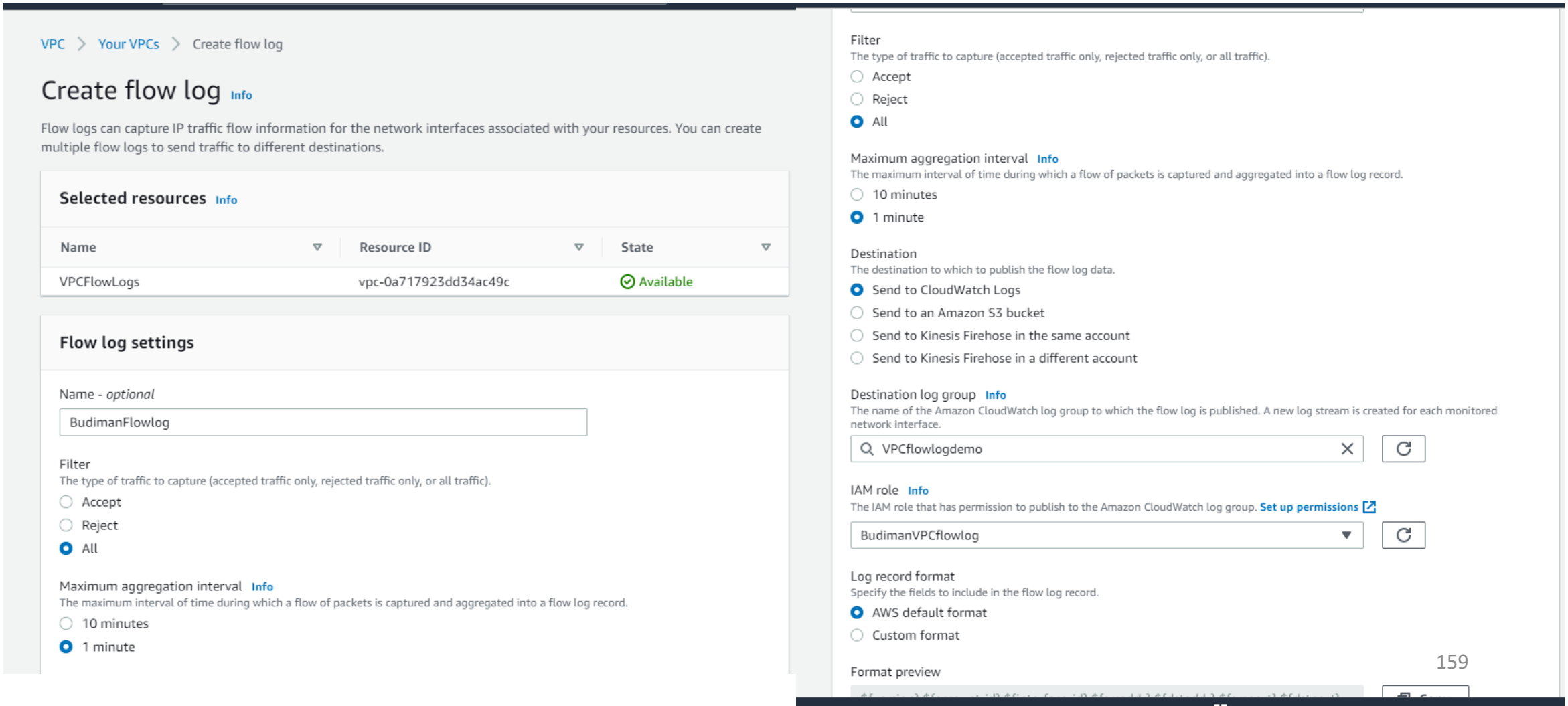
Edit Trust Policy

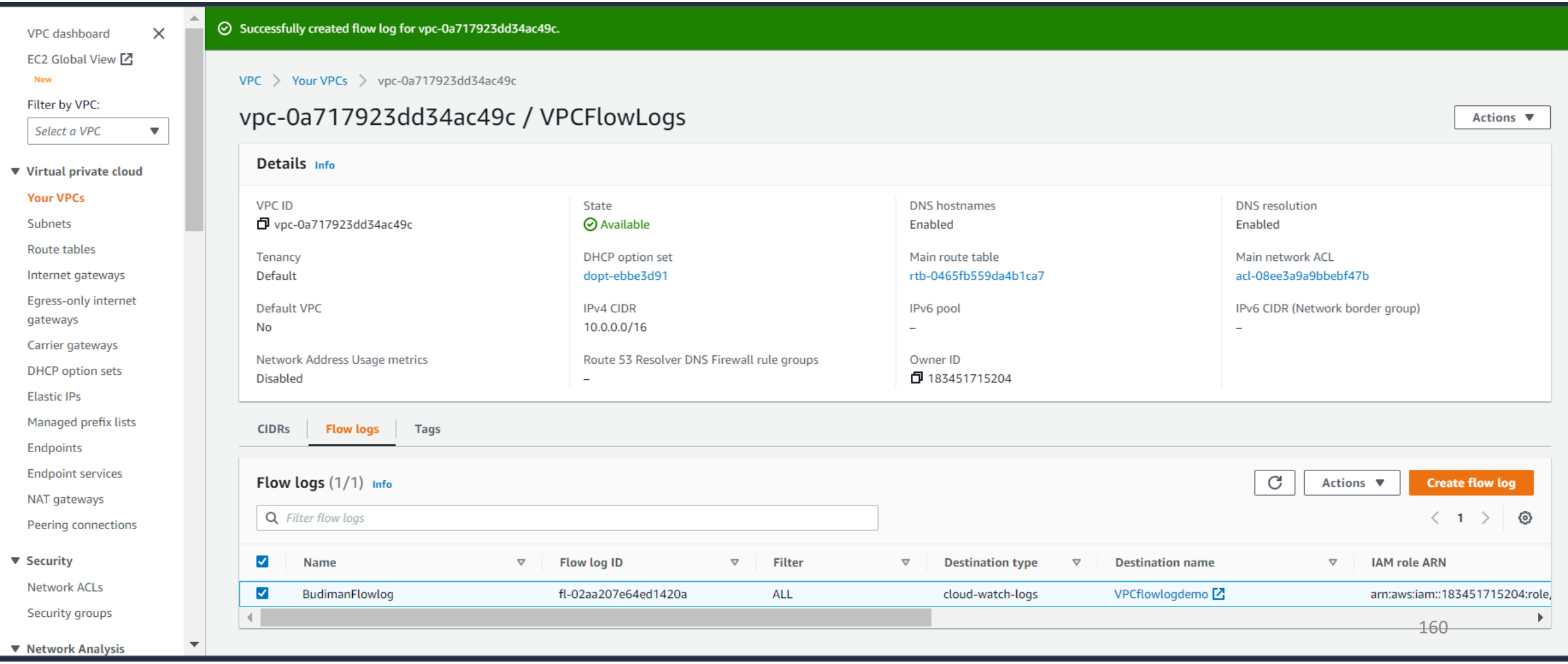



Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
Related consoles
IAM Identity Center
New
Trust policy updated.
IAM > Roles > BudimanVPCflowlog
BudimanVPCflowlog
Delete
Allows VPC flow log to publish to CloudWatch
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
Entities that can assume this role under specified conditions.
Edit trust policy
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Principal": {
"Service": "vpc-flow-logs.amazonaws.com"
},
"Action": "sts:AssumeRole"
}
]
}

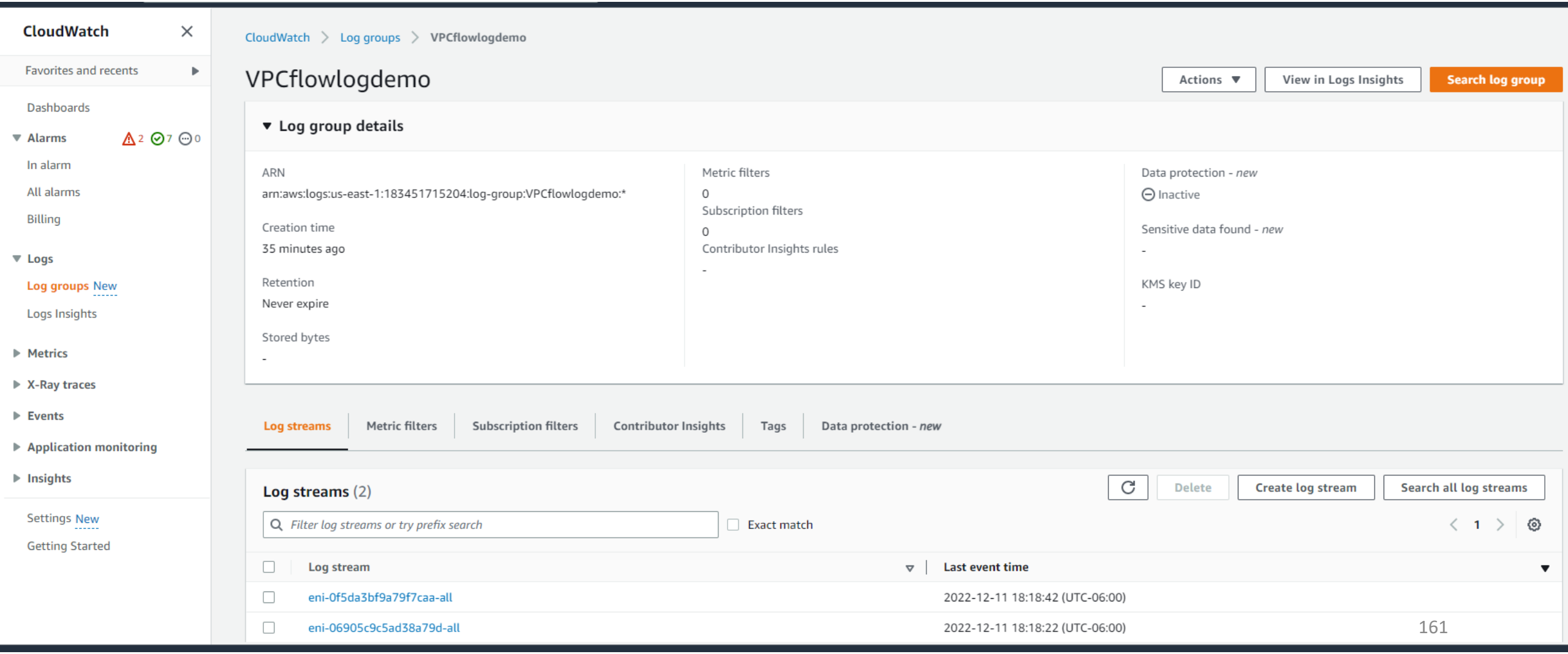
Create VPC flow log



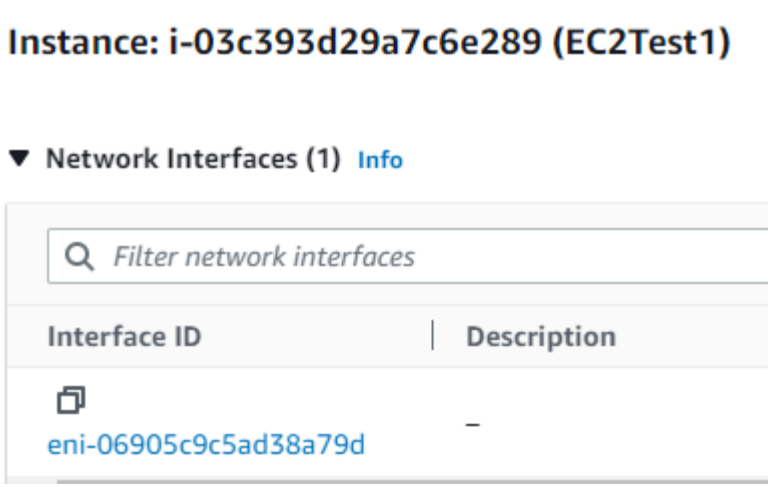

VPC > Your VPCs > Create flow log
Create flow log Info
Flow logs can capture IP traffic flow information for the network interfaces associated with your resources. You can create multiple flow logs to send traffic to different destinations.
Selected resources Info
Name
Resource ID
State
VPCFlowLogs
vpc-0a717923dd34ac49c
Available
Flow log settings
Name - optional
BudimanFlowlog
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
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.
VPCflowlogdemo
IAM role Info
The IAM role that has permission to publish to the Amazon CloudWatch log group. Set up permissions
BudimanVPCflowlog
Log record format
Specify the fields to include in the flow log record.
AWS default format
Custom format
Format preview

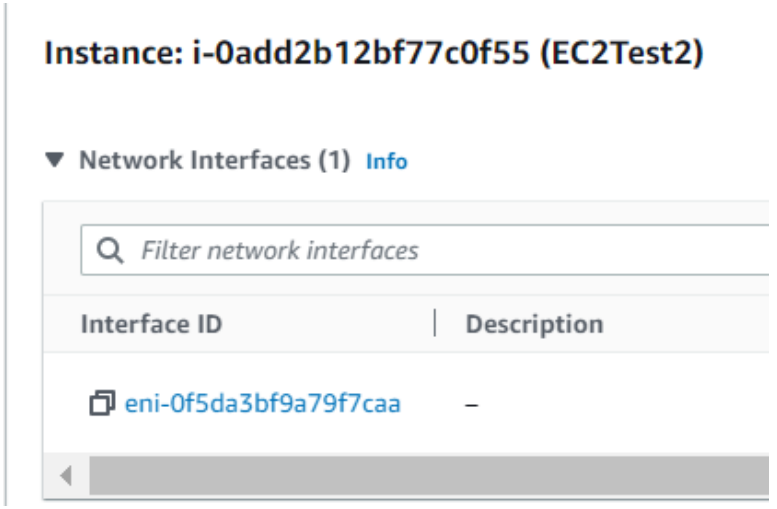

VPC dashboard
EC2 Global View
New
Filter by VPC:
Select a VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections
Security
Network ACLs
Security groups
Network Analysis
Successfully created flow log for vpc-0a717923dd34ac49c.
VPC > Your VPCs > vpc-0a717923dd34ac49c
vpc-0a717923dd34ac49c / VPCFlowLogs
Actions
Details Info
VPC ID
vpc-0a717923dd34ac49c
State
Available
DNS hostnames
Enabled
DNS resolution
Enabled
Tenancy
Default
DHCP option set
dopt-ebbe3d91
Main route table
rtb-0465fb559da4b1ca7
Main network ACL
acl-08ee3a9a9bbebf47b
Default VPC
No
IPv4 CIDR
10.0.0.0/16
IPv6 pool
–
IPv6 CIDR (Network border group)
–
Network Address Usage metrics
Disabled
Route 53 Resolver DNS Firewall rule groups
–
Owner ID
183451715204
CIDRs
Flow logs
Tags
Flow logs (1/1) Info
Actions
Create flow log
Filter flow logs
1
Name
Flow log ID
Filter
Destination type
Destination name
IAM role ARN
BudimanFlowlog
fl-02aa207e64ed1420a
ALL
cloud-watch-logs
VPCflowlogdemo
arn:aws:iam::183451715204:role,

CloudWatch


CloudWatch
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
arn:aws:logs:us-east-1:183451715204:log-group:VPCflowlogdemo:*
Creation time
35 minutes ago
Retention
Never expire
Stored bytes
-
Metric filters
0
Subscription filters
0
Contributor Insights rules
-
Data protection - new
Inactive
Sensitive data found - new
-
KMS key ID
-
Log streams
Metric filters
Subscription filters
Contributor Insights
Tags
Data protection - new
Log streams (2)
Delete
Create log stream
Search all log streams
Filter log streams or try prefix search
Exact match
1
Log stream
Last event time
eni-0f5da3bf9a79f7caa-all
2022-12-11 18:18:42 (UTC-06:00)
eni-06905c9c5ad38a79d-all
2022-12-11 18:18:22 (UTC-06:00)

CloudWatch creates a log stream per ENI

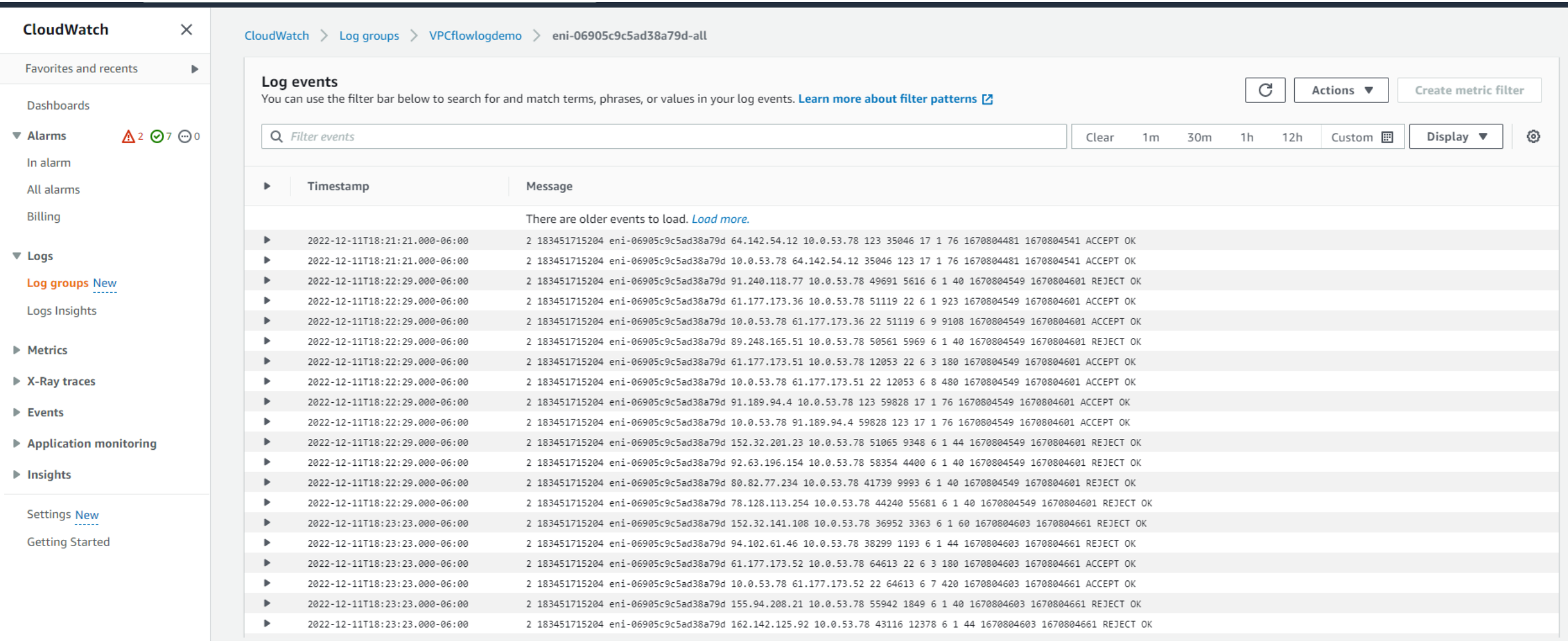




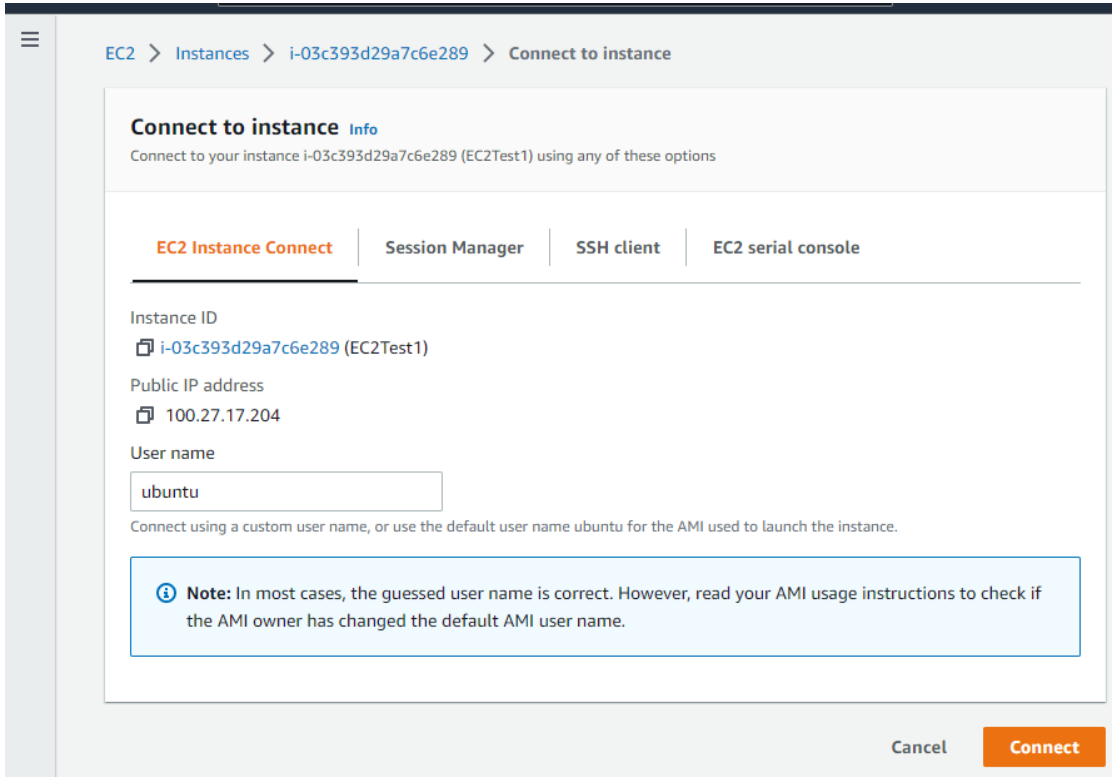
Flow log syntax

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

CloudWatch


CloudWatch
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
Actions
Create metric filter
Filter events
Clear 1m 30m 1h 12h Custom
Display
Timestamp
Message
There are older events to load. Load more.
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



Getting Started

2022-12-11T18:46:34.000-06:00 2 183451715204 eni-06905c9c5ad38a79d 152.89.196.220 10.0.53.78 18772 22 6 4 240 1670805994 1670806042 ACCEPT OK

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