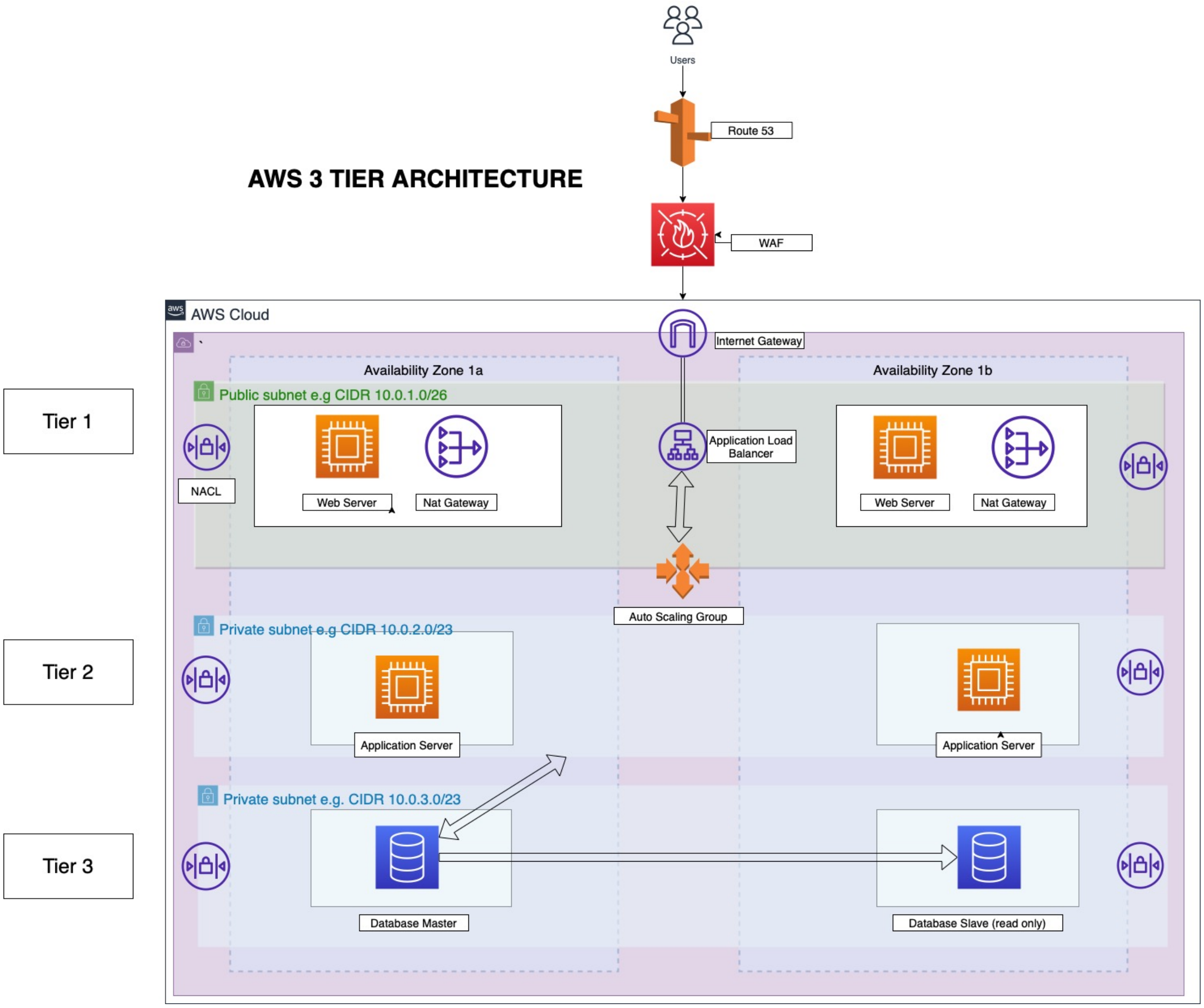
CloudForce Ke Team challenge 2 - David Tumaini

Problem Statment:

Deploy the necessary resources and components as identified in the in the 3-tier your architecture design shown below.



We are solving for:

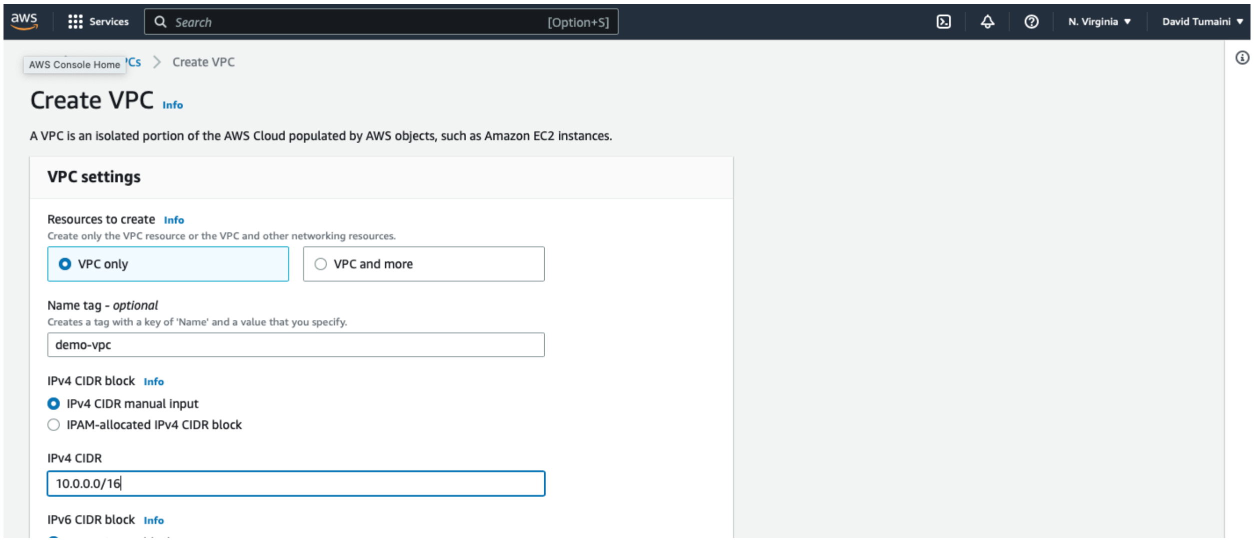
-Modularity - In a 3-tier architecture, each tier can be managed independently. Teams can focus on different tiers and changes made quickly. It also helps us recover quickly from an unexpected event by focussing solely on the faulty part.
-Scalability - Each tier can scale horizontally in response to demand by adding more resources and load balancing the existing resources.
-High availability - We can host the application in different locations.
-Fault tolerant - Our infrastructure can comfortaly adapt to any expected or unexpected change both to traffic and fault.
-Security - Users can only reach the front end webservers after clearing the firewall through the application load balancer. The back end servers and the database tier will be in a private subnet as we do not want to expose them over the internet.

Definitions:

Some background before we proceed, a few definitions to get us started..
a) Presentation Layer / Web Tier — Allows users to interact with your website or app. Its main purpose is to display information to and collect information from the user.
b) Application layer / App Tier — This is where sorting and processing of data. It is the brains of the application. It houses the business logic used to process user inputs. It can also add, modify or delete data in the database tier.
c) Data layer / Database tier — Secure storage of data in a secure manner. Its where information processed by the application tier is stored and managed.
d) Route53 - ...is a highly available and scalable Domain Name System (DNS) web service. Route 53connects user requests to internet applications running on AWS or on-premises.
e) WAF - AWS Web Application Firewall is a web application firewall that helps protect apps and APIs against bots and exploits that consume resources, skew metrics, or cause downtime.
F) Internet Gateway - ... allows communication between instances in a VPC and the internet. It imposes no availability risks or bandwidth constraints on network traffic. It provides a target for route tables to connect to the internet and performs network address translation (NAT) for instances that have been assigned IPv4 Public IP addresses
g) Nat Geteway - Resources in a private subnet do not have internet connectivity. This is intentional because it protects the resources from being accessed from the internet. However, sometimes resources in a private subnet need to communicate with the internet to download software updates and access internet services. Thus, you will want to give resources outbound connectivity to the internet while keeping them protected from inbound access.
This can be accomplished with a NAT Gateway that is launched in the public subnet:
h) NACL - A network access control list (ACL) allows or denies specific inbound or outbound traffic at the subnet level. You can use the default network ACL for your VPC, or you can create a custom network ACL for your VPC with rules that are similar to the rules for your security groups in order to add an additional layer of security to your VPC.
i) **Route Tables -** is used to direct traffic in/out of a subnet. It contains a number of CIDRs (IP address ranges) and where to direct the appropriate traffic. For example: Traffic for the Internet (0.0.0.0/0) is usually: Sent to an Internet Gateway if the Route Table is associated with a public subnet.

Steps

1. Setup the Virtual Private Cloud (VPC). Log on to your AWS services, go to VPC and click on 'Create VPC' button. Give your VPC a name and a CIDR block of 10.0.0.0/16. CIDR is a method of assigning IP addresses



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

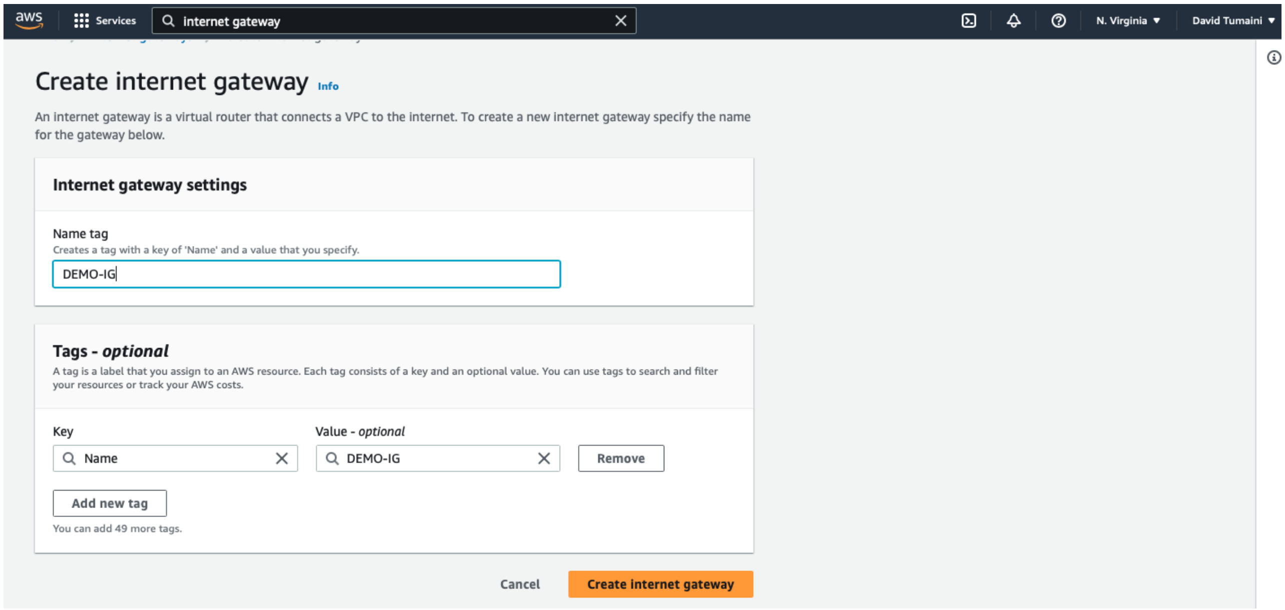
Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.
demo-vpc

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

IPv4 CIDR
10.0.0.0/16

IPv6 CIDR block Info

2. Setup the Internet Geteway - navigate to Internet Gateways page and click on create internet gateway button.



Create internet gateway Info

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

Internet gateway settings

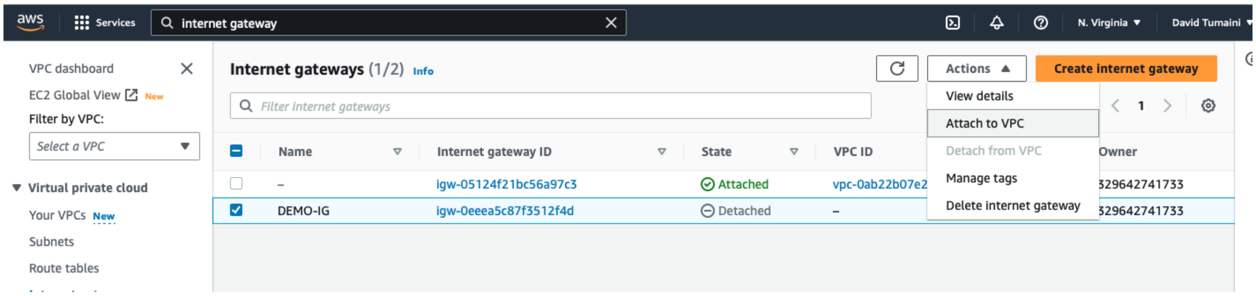
Name tag
Creates a tag with a key of 'Name' and a value that you specify.
DEMO-IG

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

Key: Name
Value - *optional*: DEMO-IG

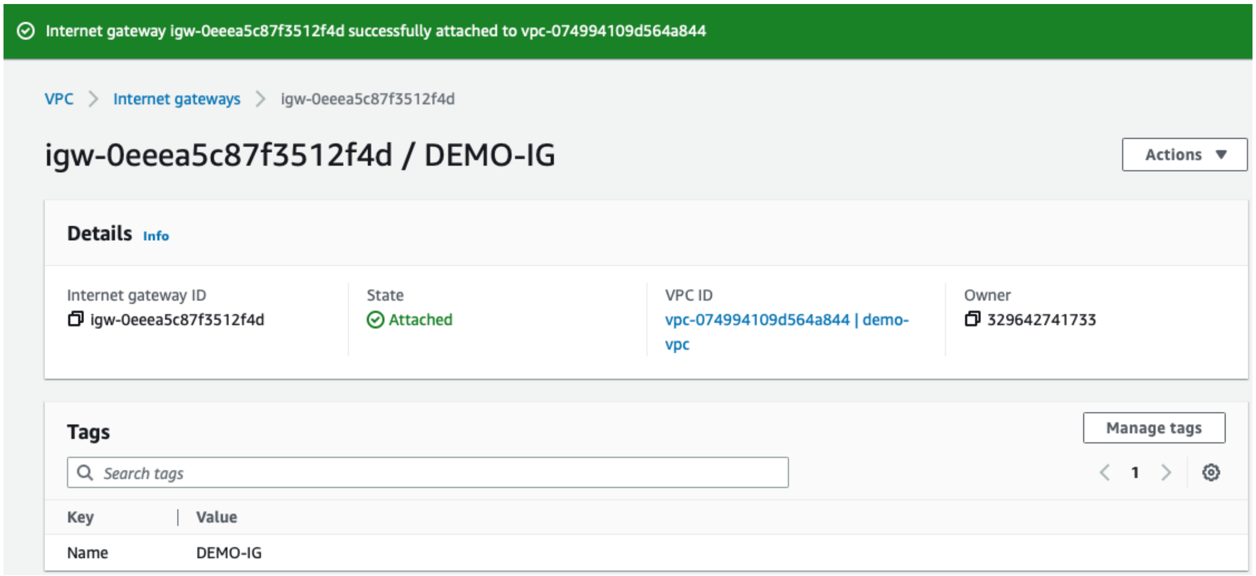
You can add 49 more tags.

We need to attached a VPC to our Internet Gateway: Select the Internet Gateway, click on actions button and then select 'Attach to VPC'. Select the VPC to attach and then click 'Attach'.



Internet gateways (1/2) Info

Name	Internet gateway ID	State	VPC ID	Owner
–	igw-05124f21bc56a97c3	Attached	vpc-0ab22b07e2	329642741733
DEMO-IG	igw-0eeea5c87f3512f4d	Detached	–	329642741733



Internet gateway igw-0eeea5c87f3512f4d successfully attached to vpc-074994109d564a844

igw-0eeea5c87f3512f4d / DEMO-IG

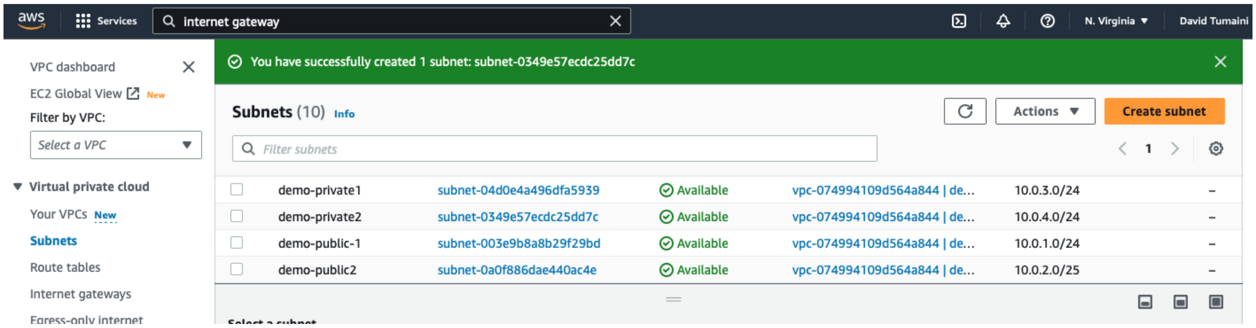
Details Info

Internet gateway ID	State	VPC ID	Owner
igw-0eeea5c87f3512f4d	Attached	vpc-074994109d564a844 | demo-vpc	329642741733

Tags

Key	Value
Name	DEMO-IG

3. Create four (4) Subnets: A subnet is a way for us to group our resources within the VPC with their IP range. It can be either public (can directly access the internet) or private. Lets create the following subnets.
demo-public1 - CIDR (10.0.1.0/24) - Availability zone (us-east-1a)
demo-public2 - CIDR (10.0.2.0/24) - Availability zone (us-east-1b)
demo-private1 - CIDR (10.0.3.0/24) - Availability zone (us-east-1a)
demo-private2 - CIDR (10.0.4.0/24) - Availability zone (us-east-1b)

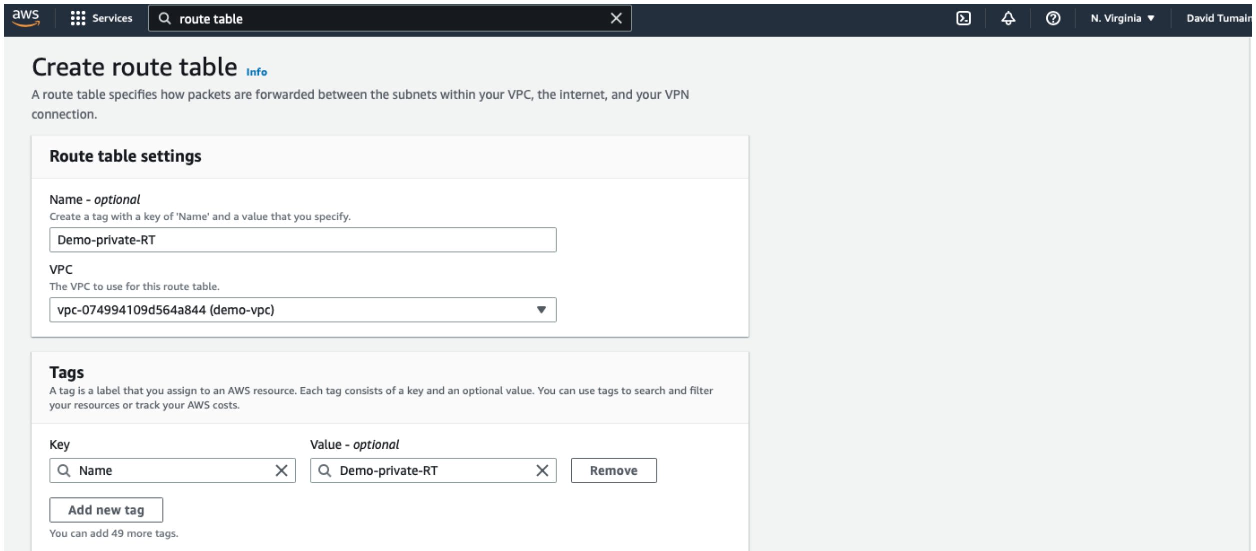


You have successfully created 1 subnet: subnet-0349e57ecdc25dd7c

Subnets (10) Info

Name	Subnet ID	State	VPC	IPv4 CIDR	
demo-private1	subnet-04d0e4a496dfa5939	Available	vpc-074994109d564a844 | de...	10.0.3.0/24	–
demo-private2	subnet-0349e57ecdc25dd7c	Available	vpc-074994109d564a844 | de...	10.0.4.0/24	–
demo-public-1	subnet-003e9b8a8b29f29bd	Available	vpc-074994109d564a844 | de...	10.0.1.0/24	–
demo-public2	subnet-0a0f886dae440ac4e	Available	vpc-074994109d564a844 | de...	10.0.2.0/25	–

4. Create two route tables. We will create a public route table and a private route table. The public route table will define subnets with direct access to internet, and the resources to the private subnets will only have access to the internet through NAT gateway.



Create route table Info

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

Route table settings

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

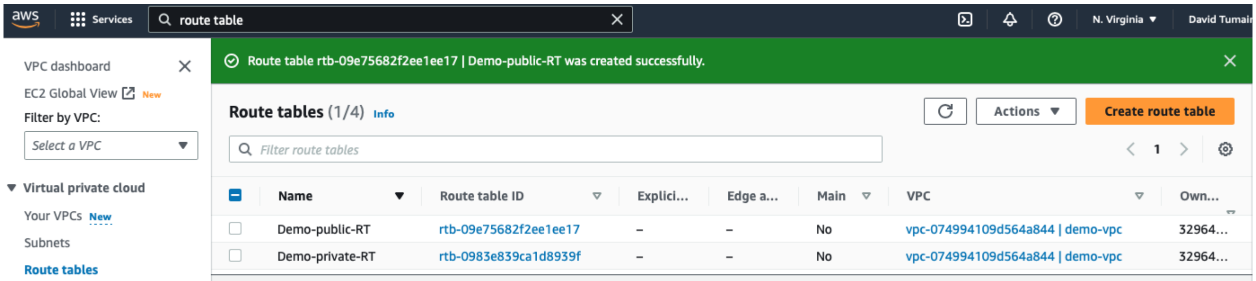
VPC
The VPC to use for this route table.
vpc-074994109d564a844 (demo-vpc)

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	Demo-private-RT	Remove

Add new tag
You can add 49 more tags.

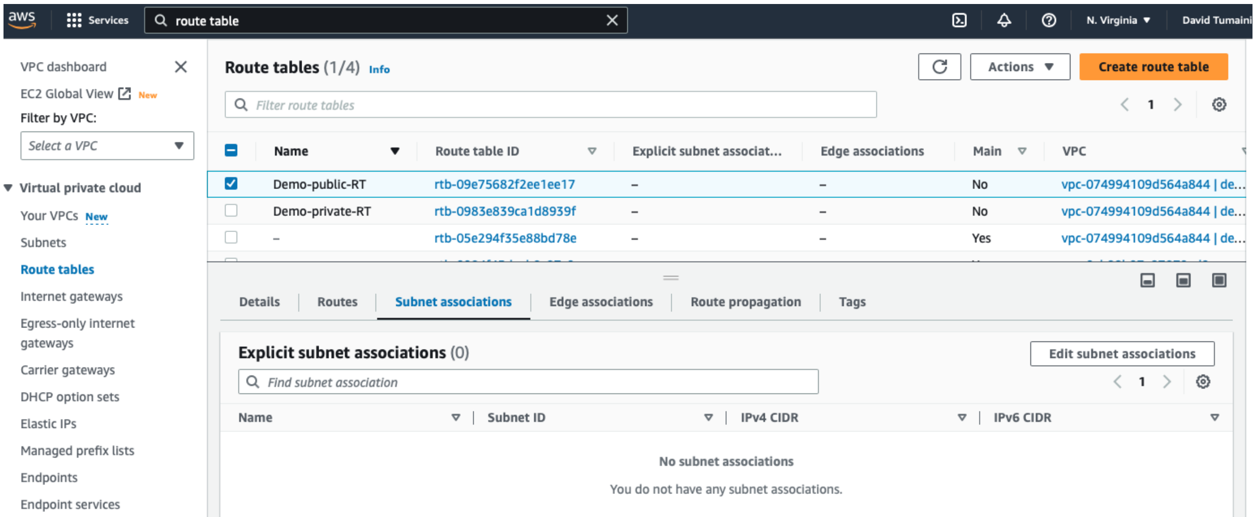


Route table rtb-09e75682f2ee1ee17 | Demo-public-RT was created successfully.

Route tables (1/4) Info

☐	Name	Route table ID	Explici...	Edge a...	Main	VPC	Own...
☐	Demo-public-RT	rtb-09e75682f2ee1ee17	–	–	No	vpc-074994109d564a844 | demo-vpc	32964...
☐	Demo-private-RT	rtb-0983e839ca1d8939f	–	–	No	vpc-074994109d564a844 | demo-vpc	32964...

The pubic and private subnets need to be associated with the respective route tables. To do that, we select the route table and choose the ‘Subnet Association’ tab.



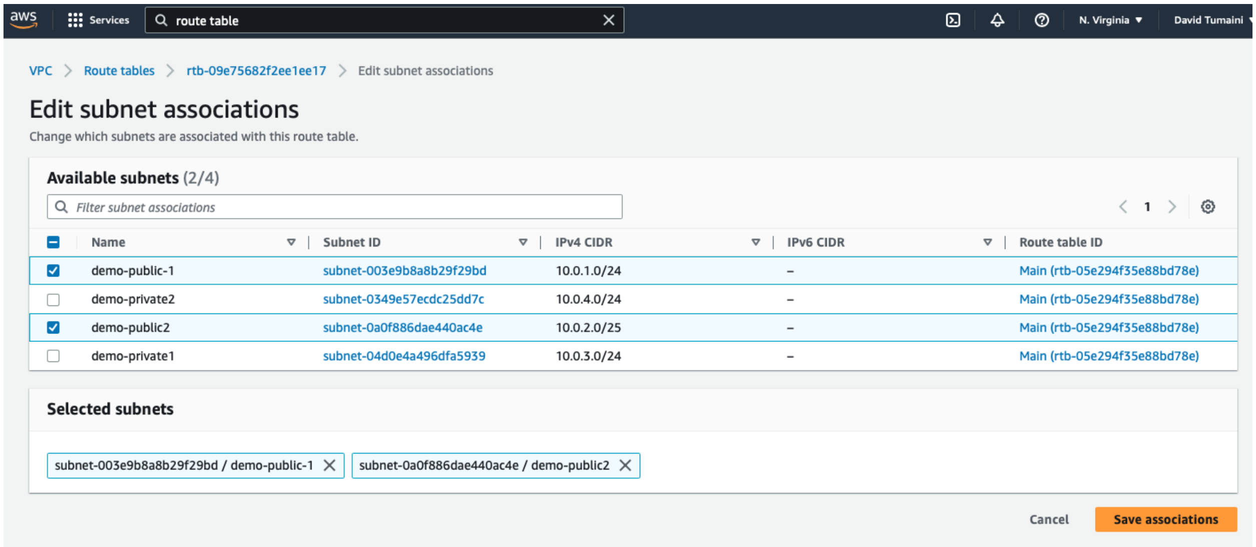
Route tables (1/4) Info

☐	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC
☑	Demo-public-RT	rtb-09e75682f2ee1ee17	–	–	No	vpc-074994109d564a844 | de...
☐	Demo-private-RT	rtb-0983e839ca1d8939f	–	–	No	vpc-074994109d564a844 | de...
☐	–	rtb-05e294f35e88bd78e	–	–	Yes	vpc-074994109d564a844 | de...

Explicit subnet associations (0)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR

No subnet associations
You do not have any subnet associations.



aws
Services
route table
N. Virginia ▼
David Tumaini
VPC > Route tables > rtb-09e75682f2ee1ee17 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

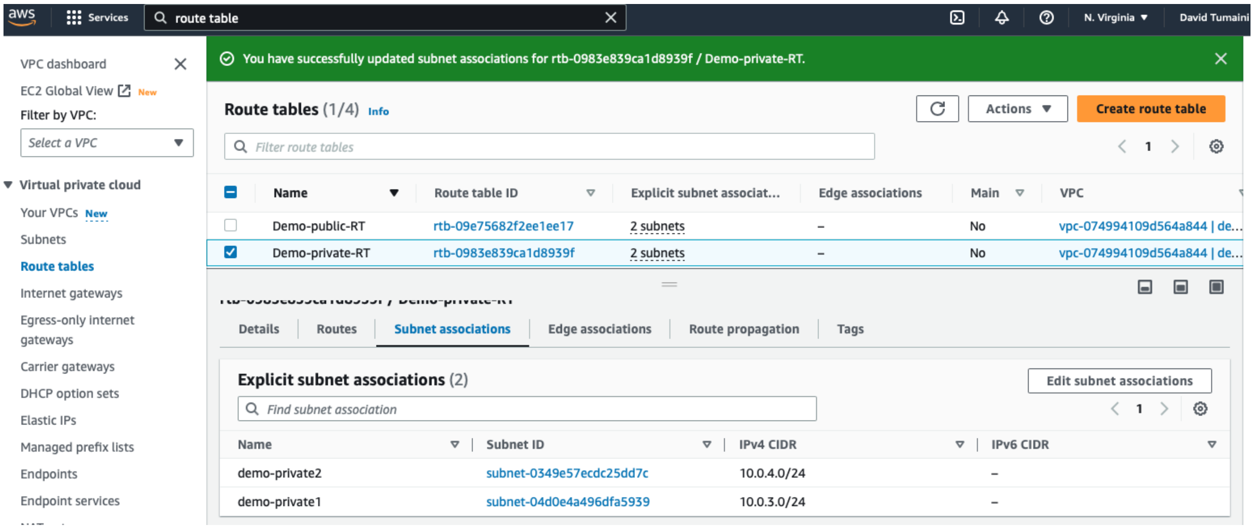
Available subnets (2/4)

Filter subnet associations

	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
☑	demo-public-1	subnet-003e9b8a8b29f29bd	10.0.1.0/24	–	Main (rtb-05e294f35e88bd78e)
□	demo-private2	subnet-0349e57ecdc25dd7c	10.0.4.0/24	–	Main (rtb-05e294f35e88bd78e)
☑	demo-public2	subnet-0a0f886dae440ac4e	10.0.2.0/25	–	Main (rtb-05e294f35e88bd78e)
□	demo-private1	subnet-04d0e4a496dfa5939	10.0.3.0/24	–	Main (rtb-05e294f35e88bd78e)

Selected subnets

subnet-003e9b8a8b29f29bd / demo-public-1 ✕ subnet-0a0f886dae440ac4e / demo-public2 ✕



You have successfully updated subnet associations for rtb-0983e839ca1d8939f / Demo-private-RT.

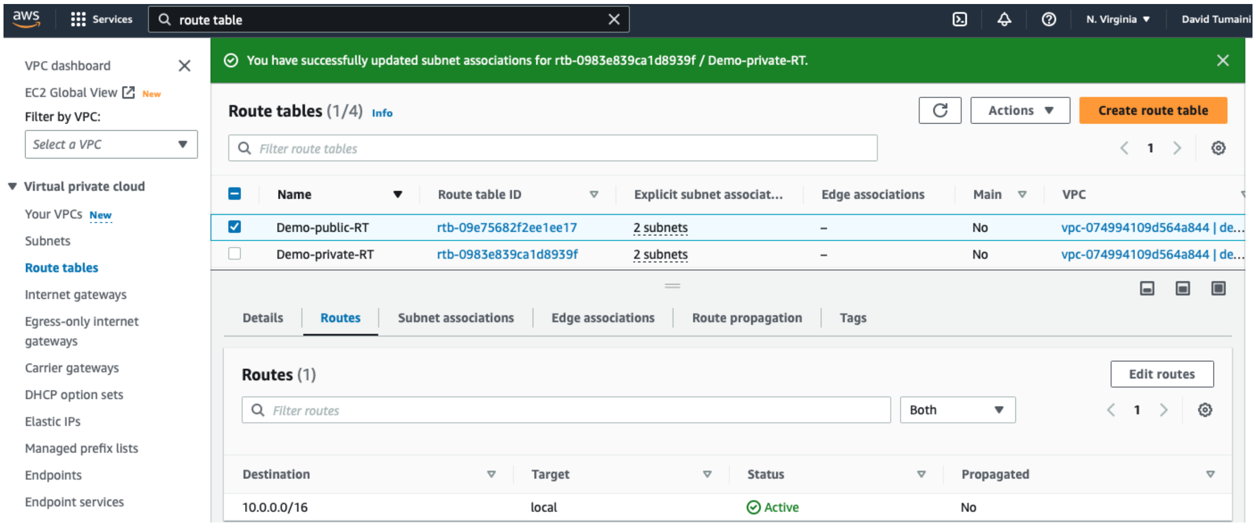
Route tables (1/4) Info

	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC
□	Demo-public-RT	rtb-09e75682f2ee1ee17	2 subnets	–	No	vpc-074994109d564a844 | de...
☑	Demo-private-RT	rtb-0983e839ca1d8939f	2 subnets	–	No	vpc-074994109d564a844 | de...

Explicit subnet associations (2)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
demo-private2	subnet-0349e57ecdc25dd7c	10.0.4.0/24	–
demo-private1	subnet-04d0e4a496dfa5939	10.0.3.0/24	–

We need to route the traffic to the internet through the internet gateway for our public route table. To do this, select the public route table and choose routes tab.



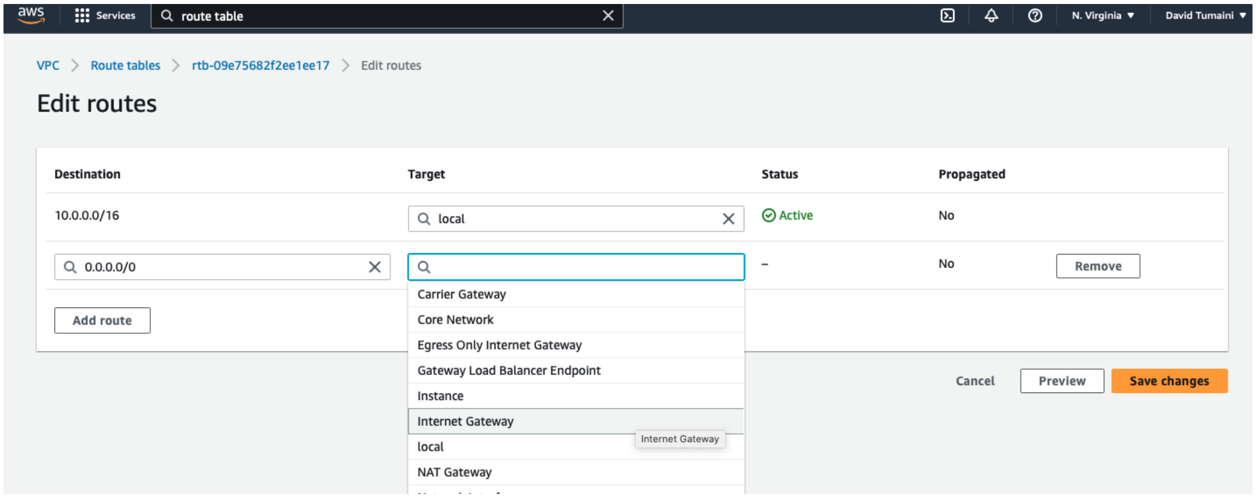
You have successfully updated subnet associations for rtb-0983e839ca1d8939f / Demo-private-RT.

Route tables (1/4) Info

	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC
☑	Demo-public-RT	rtb-09e75682f2ee1ee17	2 subnets	–	No	vpc-074994109d564a844 | de...
□	Demo-private-RT	rtb-0983e839ca1d8939f	2 subnets	–	No	vpc-074994109d564a844 | de...

Routes (1)

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

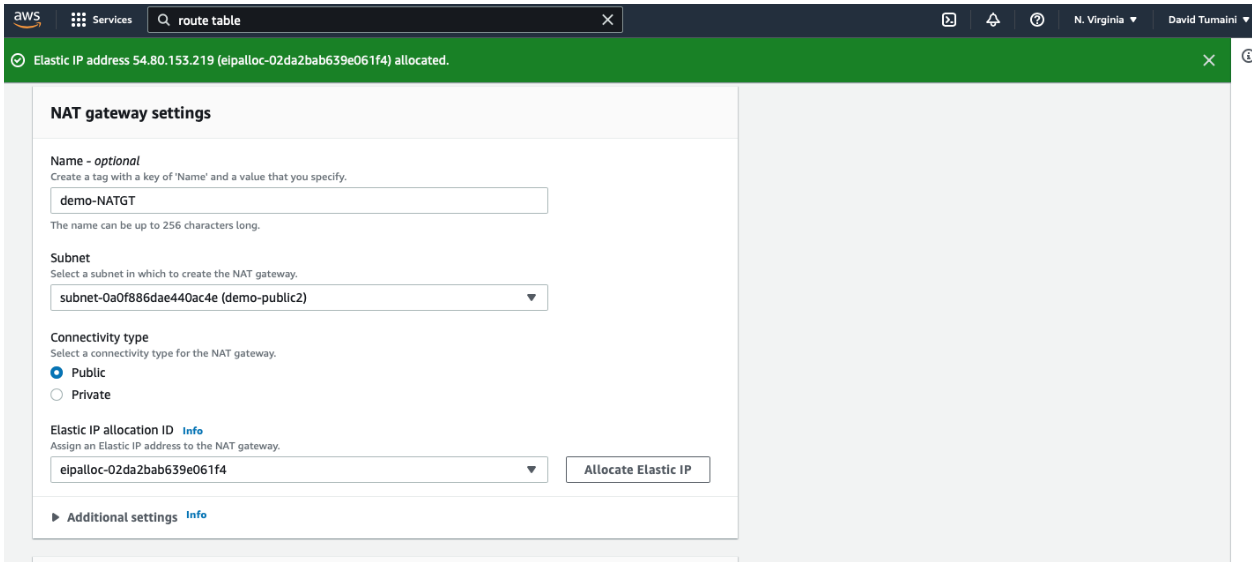


VPC > Route tables > rtb-09e75682f2ee1ee17 > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0		-	No

5. Create NAT Gateway. Navigate to NAT Gateways page and then click on 'Create NAT Gateway'. Please ensure that you know the subnet ID for the demo-public2. This will be needed when creating the NAT Gateway...



Elastic IP address 54.80.153.219 (eipalloc-02da2bab639e061f4) allocated.

NAT gateway settings

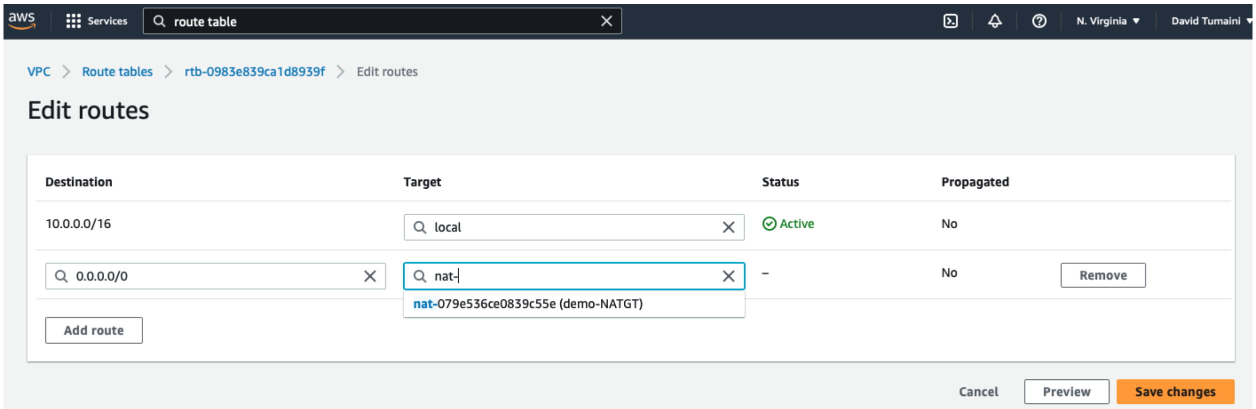
Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
demo-NATGT
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
subnet-0a0f886dae440ac4e (demo-public2)

Connectivity type
Select a connectivity type for the NAT gateway.
● Public
○ Private

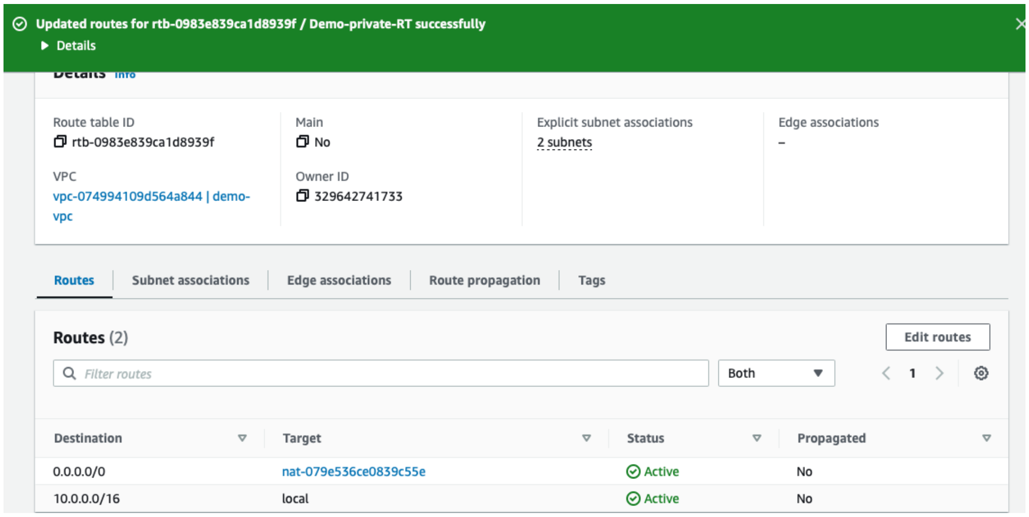
Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.
eipalloc-02da2bab639e061f4

Now that we have the NAT Gateway, lets edit the private route table to make use of that gateway to access the internet.



Edit routes

Destination	Target	Status	Propagated	
10.0.0.0/16	local	Active	No	
0.0.0.0/0	nat- nat-079e536ce0839c55e (demo-NATGT)	-	No	Remove



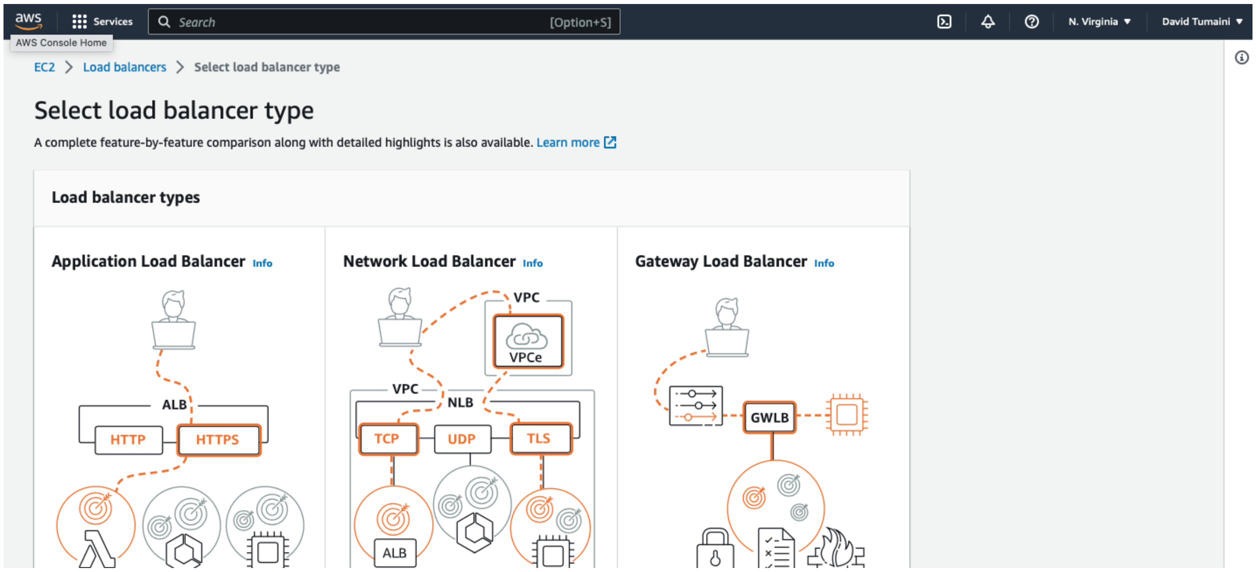
Updated routes for rtb-0983e839ca1d8939f / Demo-private-RT successfully

Route table ID: rtb-0983e839ca1d8939f
Main: No
Explicit subnet associations: 2 subnets
Edge associations: –
VPC: vpc-074994109d564a844 | demo-vpc
Owner ID: 329642741733

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	nat-079e536ce0839c55e	Active	No
10.0.0.0/16	local	Active	No

6. Create Load balancer - From our architecture, our front end tier can only accept traffic from the elastic load balancer, which connects directly to the internet gateway. To create load balancers, we navigate to load balancer page (click services and select EC2 under compute), in the left navigation page click ‘create load balancer’. Select the application load balancer.



EC2 > Load balancers > Select load balancer type

Select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. Learn more

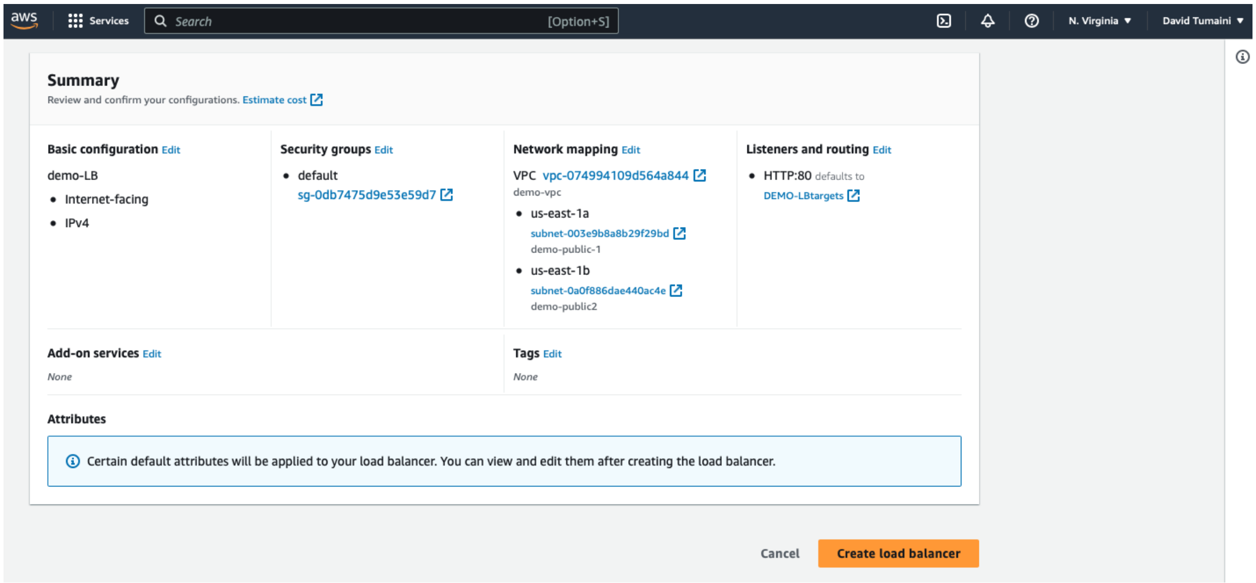
Load balancer types

Application Load Balancer Info

Network Load Balancer Info

Gateway Load Balancer Info

Configure the load balancer with a name, select internet facing. Under availability zone, select the two public subnets. Under the security group, we only need to allow the ports the application needs (HTTP port 80 and/or HTTPS port 443) on our internet facing load balancers.
Under the configure routing, we will give the Target group a name, this will be needed when we create our Auto Scaling group.



Summary

Review and confirm your configurations. Estimate cost

Basic configuration Edit

demo-LB

- Internet-facing
- IPv4

Security groups Edit

- default
  sg-0db7475d9e53e59d7

Network mapping Edit

VPC vpc-074994109d564a844
demo-vpc

- us-east-1a
  subnet-003e9b8a8b29f29bd
  demo-public-1
- us-east-1b
  subnet-0a0f886dae440ac4e
  demo-public2

Listeners and routing Edit

- HTTP:80 defaults to
  DEMO-LBtargets

Add-on services Edit

None

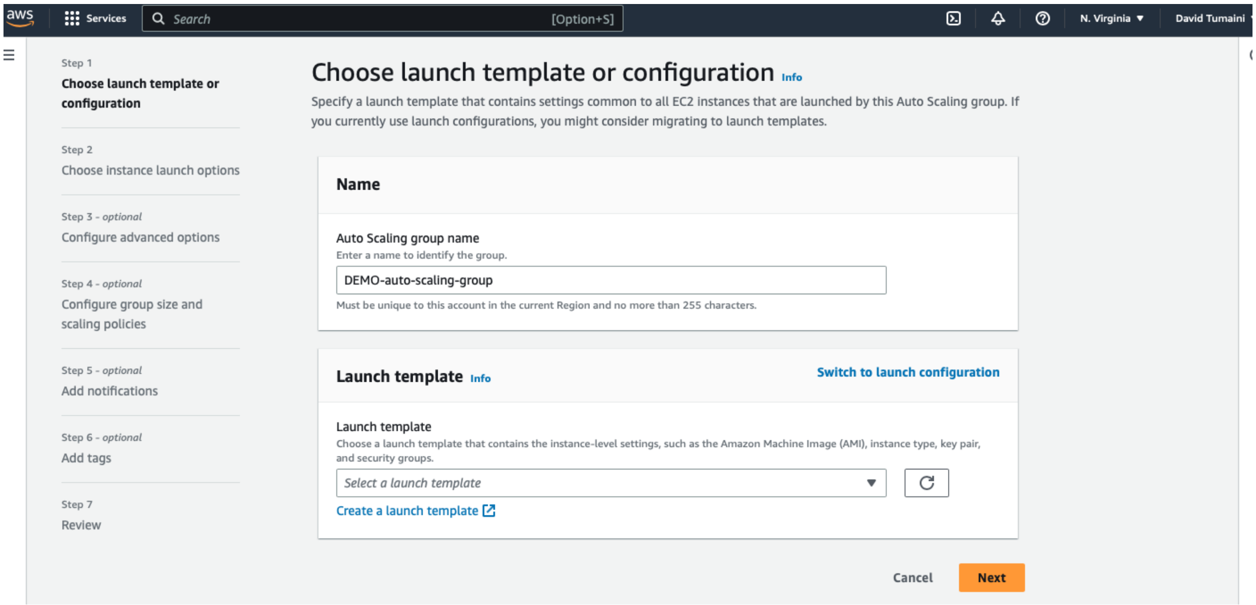
Tags Edit

None

Attributes

Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

7. Auto Scaling Group - With auto scaling, our application will be able to accomodate additional traffic or shring when there's low demand to save cost. To create and auto scaling group, click on 'Auto Scaling Groups' under EC2, and click the 'Create Auto Scaling Group' button.
Note: Instances within an auto scaling group need to have a common configuration. This is made possible with the help of a Launch Configuration.
In our Launch configuration, choose the AMI, choose the approriate instance type, give the launch configuration a name. Also, under the 'Advanced Details' dropdown, the user data is provided for you to type in a command that is needed to install dependencies and start the application.
Under security group, we will only allow the ports that are necessary for our application. Review and click 'Create Launch Configuration. Create a new security pair and download it before proceeding.



Choose launch template or configuration

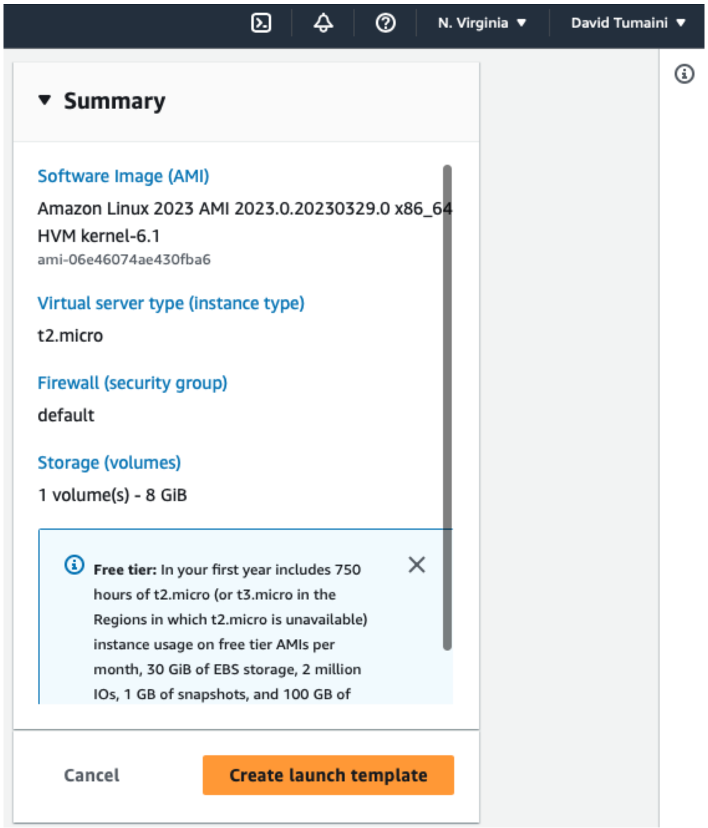
Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Name

Auto Scaling group name
Enter a name to identify the group.
DEMO-auto-scaling-group
Must be unique to this account in the current Region and no more than 255 characters.

Launch template

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
Select a launch template ▼



▼ **Summary**

Software Image (AMI)
Amazon Linux 2023 AMI 2023.0.20230329.0 x86_64 HVM kernel-6.1
ami-06e46074ae430fba6

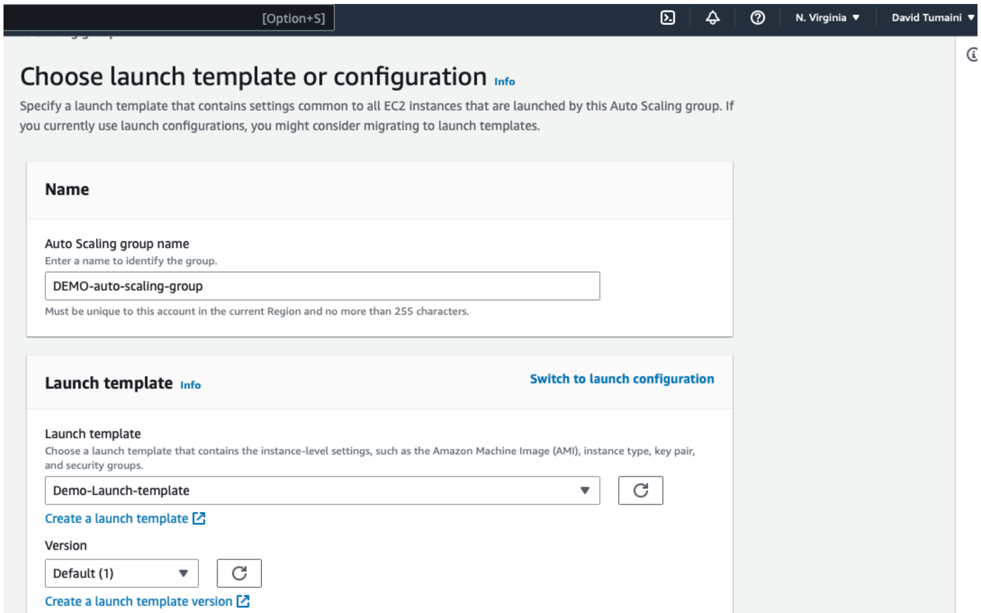
Virtual server type (instance type)
t2.micro

Firewall (security group)
default

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of

Now that we have our launch configuration, we can finish up with creating our Auto Scaling Group.



Choose launch template or configuration Info

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Name

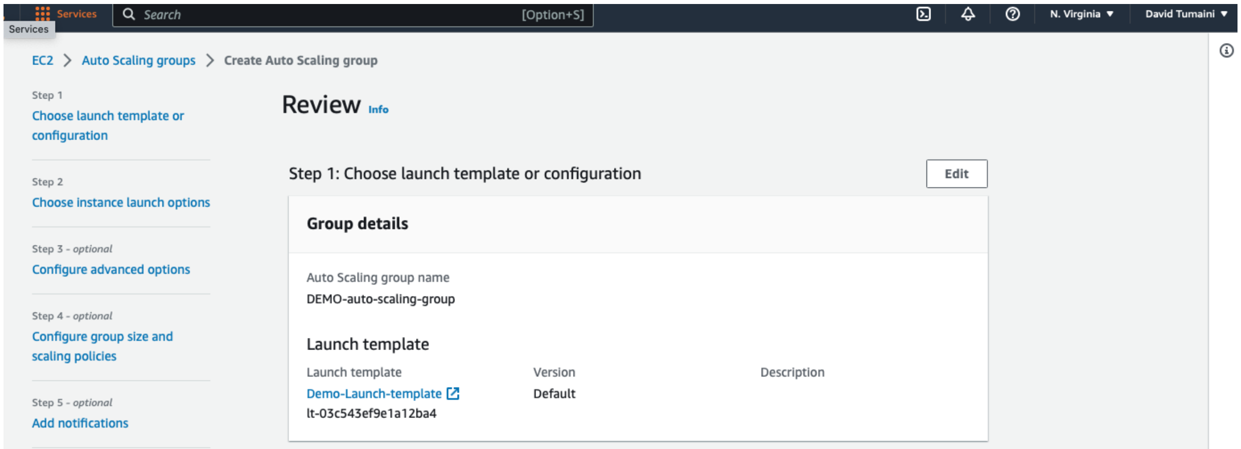
Auto Scaling group name
Enter a name to identify the group.
DEMO-auto-scaling-group
Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info

Switch to launch configuration

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
Demo-Launch-template ▼
Create a launch template

Version
Default (1) ▼
Create a launch template version


Services
Search
[Option+S]
N. Virginia
David Tumaini
EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template or configuration
Step 2
Choose instance launch options
Step 3 - optional
Configure advanced options
Step 4 - optional
Configure group size and scaling policies
Step 5 - optional
Add notifications
Info
Edit

Review

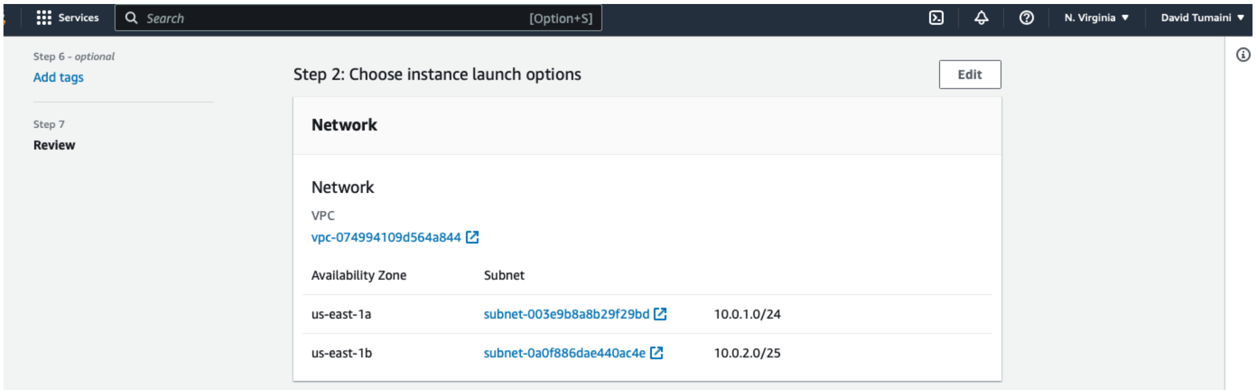
Step 1: Choose launch template or configuration

Group details

Auto Scaling group name
DEMO-auto-scaling-group

Launch template

Launch template	Version	Description
Demo-Launch-template lt-03c543ef9e1a12ba4	Default	

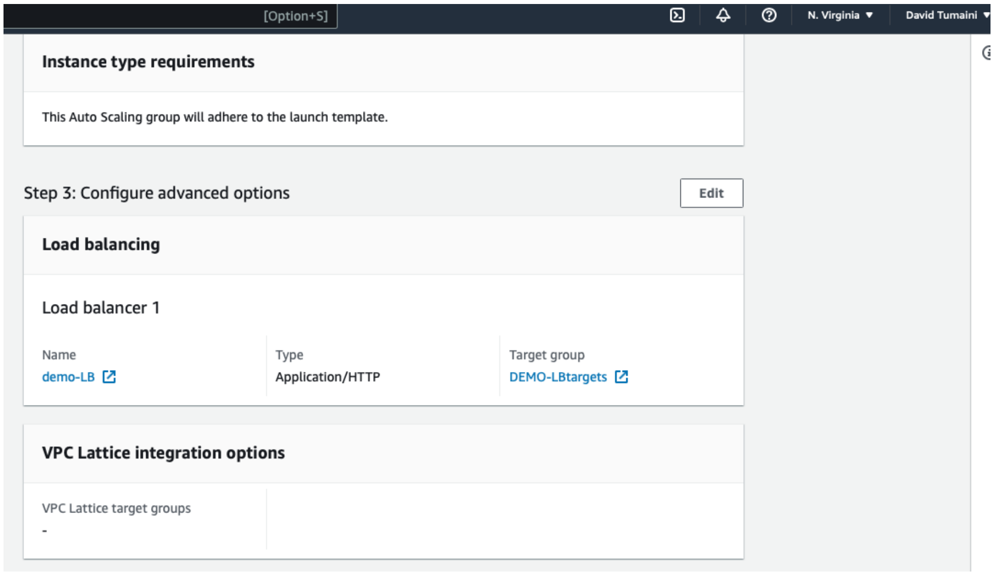


Step 2: Choose instance launch options

Network

Network
VPC
vpc-074994109d564a844

Availability Zone	Subnet	
us-east-1a	subnet-003e9b8a8b29f29bd	10.0.1.0/24
us-east-1b	subnet-0a0f886dae440ac4e	10.0.2.0/25



Instance type requirements

This Auto Scaling group will adhere to the launch template.

Step 3: Configure advanced options

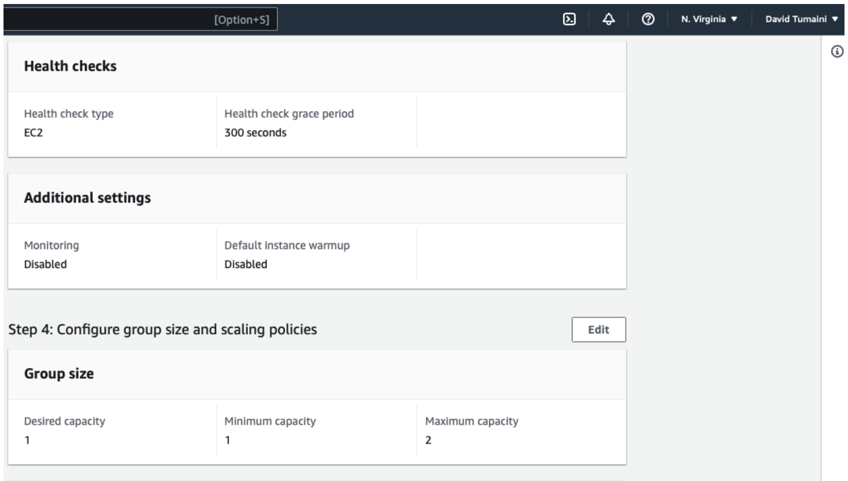
Load balancing

Load balancer 1

Name	Type	Target group
demo-LB	Application/HTTP	DEMO-LBtargets

VPC Lattice integration options

VPC Lattice target groups
-



Health checks

Health check type: EC2

Health check grace period: 300 seconds

Additional settings

Monitoring: Disabled

Default instance warmup: Disabled

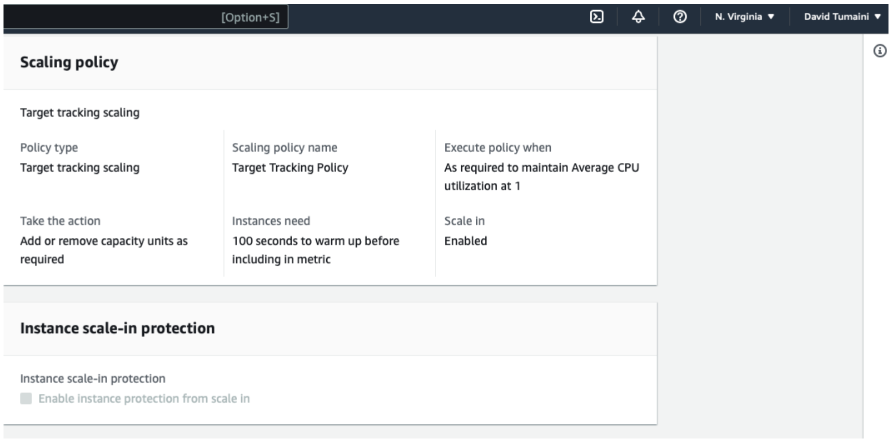
Step 4: Configure group size and scaling policies

Group size

Desired capacity: 1

Minimum capacity: 1

Maximum capacity: 2



Scaling policy

Target tracking scaling

Policy type: Target tracking scaling

Scaling policy name: Target Tracking Policy

Execute policy when: As required to maintain Average CPU utilization at 1

Take the action: Add or remove capacity units as required

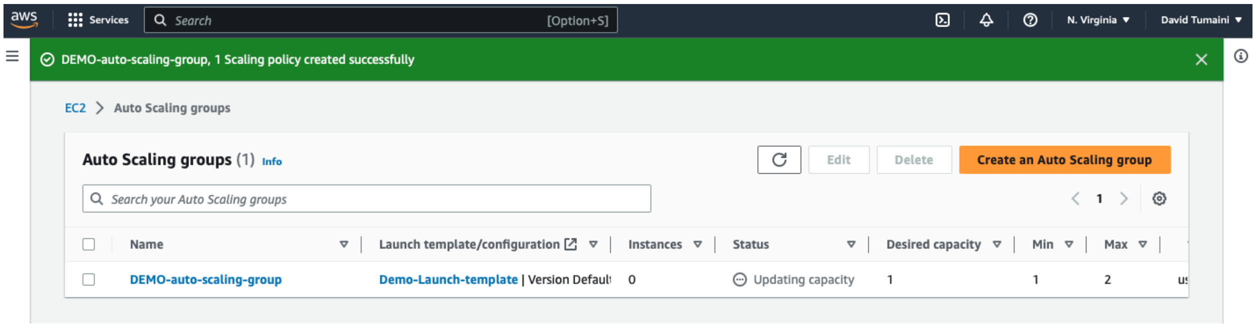
Instances need: 100 seconds to warm up before including in metric

Scale in: Enabled

Instance scale-in protection

Instance scale-in protection

□ Enable instance protection from scale in


aws
Services
Search
[Option+S]
N. Virginia
David Tumaini
EC2 > Auto Scaling groups
Edit
Delete
Create an Auto Scaling group
Search your Auto Scaling groups
1

DEMO-auto-scaling-group, 1 Scaling policy created successfully

Auto Scaling groups (1) Info

□	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	
□	DEMO-auto-scaling-group	Demo-Launch-template | Version Default	0	Updating capacity	1	1	2	us

Conclusion

There were alot of clicking and and configurations when using the console to set up a 3-tier architecture in AWS. It is, however necessary to go through this process so that its easier to move towards automation.