THREE-TIER ARCHITECTURE

PRESENTED BY DHAVANISHA JEGANNATHAN

INTRODUCTION

3-tier architecture is a widely used software design pattern that organizes applications into separate layers or tiers, each responsible for a specific functionality. This approach enhances modularity, scalability, and maintainability, making it ideal for modern web and enterprise applications. By leveraging AWS cloud services, this architecture becomes even more robust and scalable, allowing seamless integration of tools and technologies.

Definition

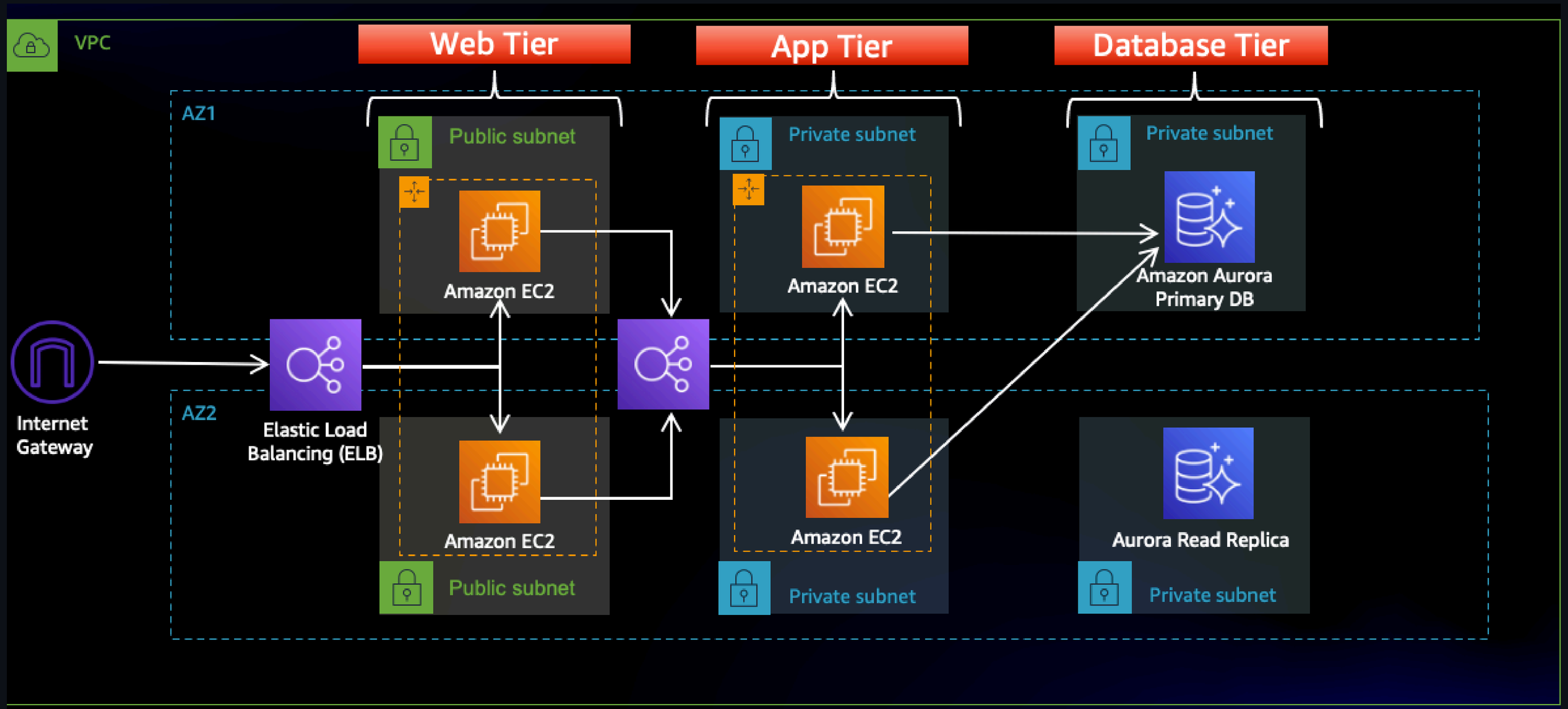
3-tier architecture separates an application into layers

- **Presentation Tier:** The user interface (UI) or front-end of the application.
- **Logic/Application Tier:** Handles business logic, application rules, and processing.
- **Data Tier:** Manages data storage and retrieval, typically involving databases.

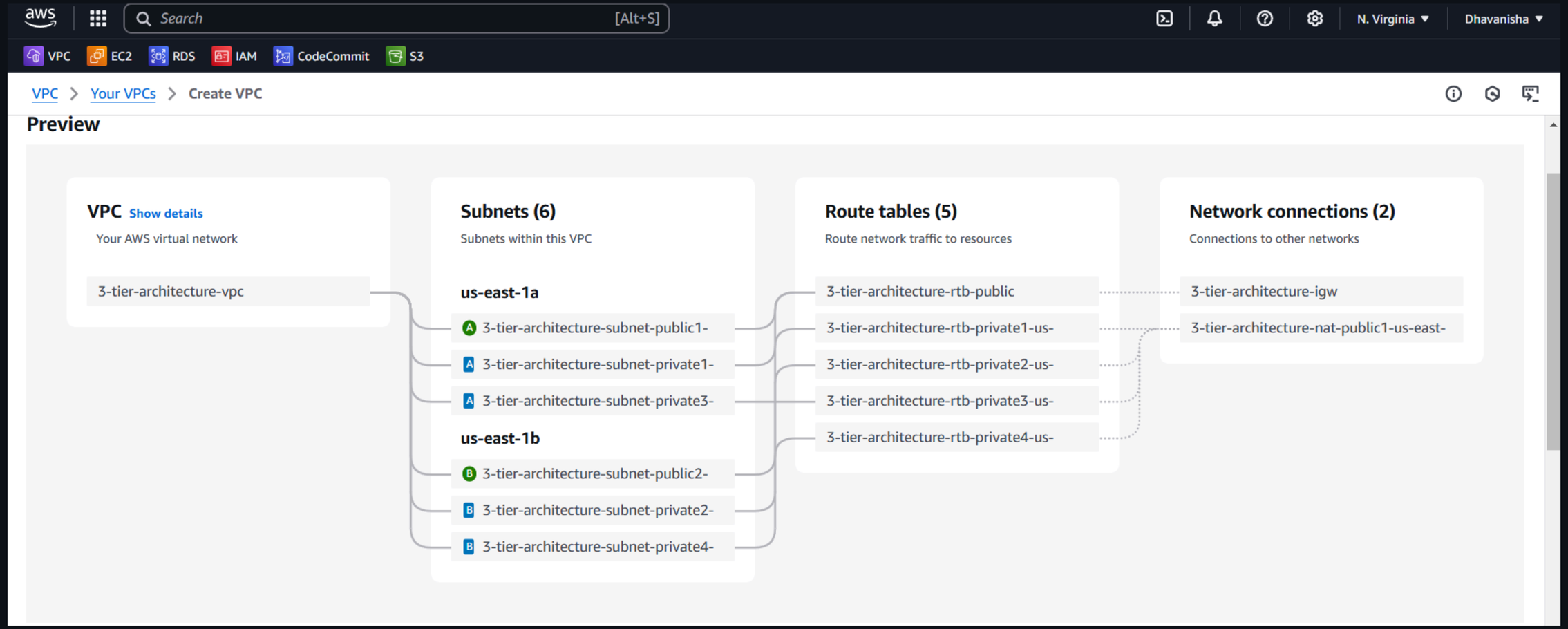
TOOLS USED IN THE PROJECT

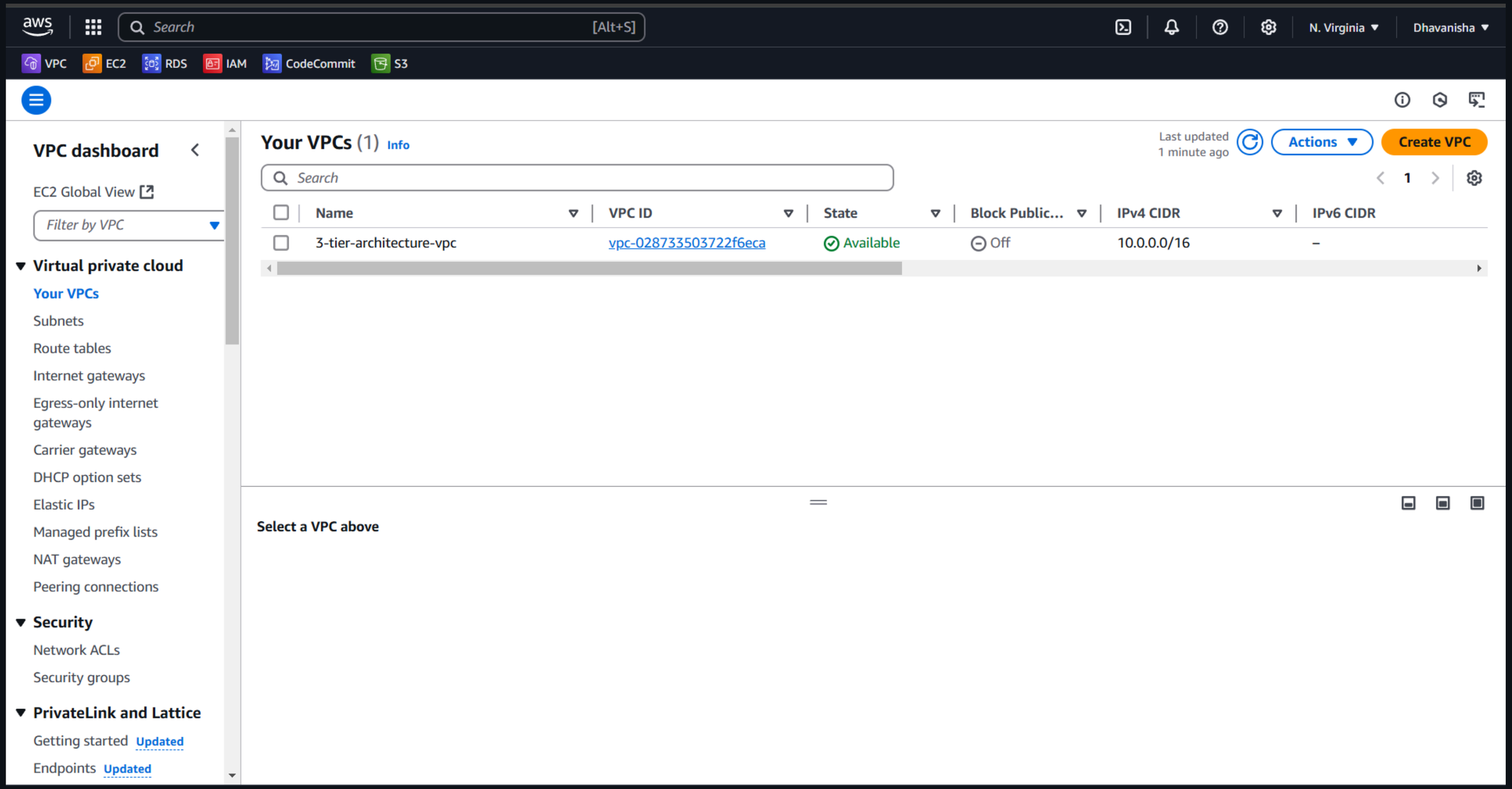
- **VPC (Virtual Private Cloud):** Provides isolated networking for secure communication between tiers.
- **EC2 (Elastic Compute Cloud):** Hosts the application logic and ensures scalability.
- **S3 (Simple Storage Service):** Stores static files and media securely.
- **RDS (Relational Database Service):** Manages relational databases for data persistence.
- **IAM (Identity and Access Management):** Ensures secure access and permissions for resources.
- **Load Balancer:** Distributes incoming traffic across multiple EC2 instances for fault tolerance.
- **Route 53:** Manages DNS for efficient traffic routing.
- **AWS Certificate Manager:** Provides SSL certificates for secure communication.
- **Auto Scaling:** Automatically adjusts resources to handle traffic surges.
- **CloudWatch:** Monitors application performance and resource usage.

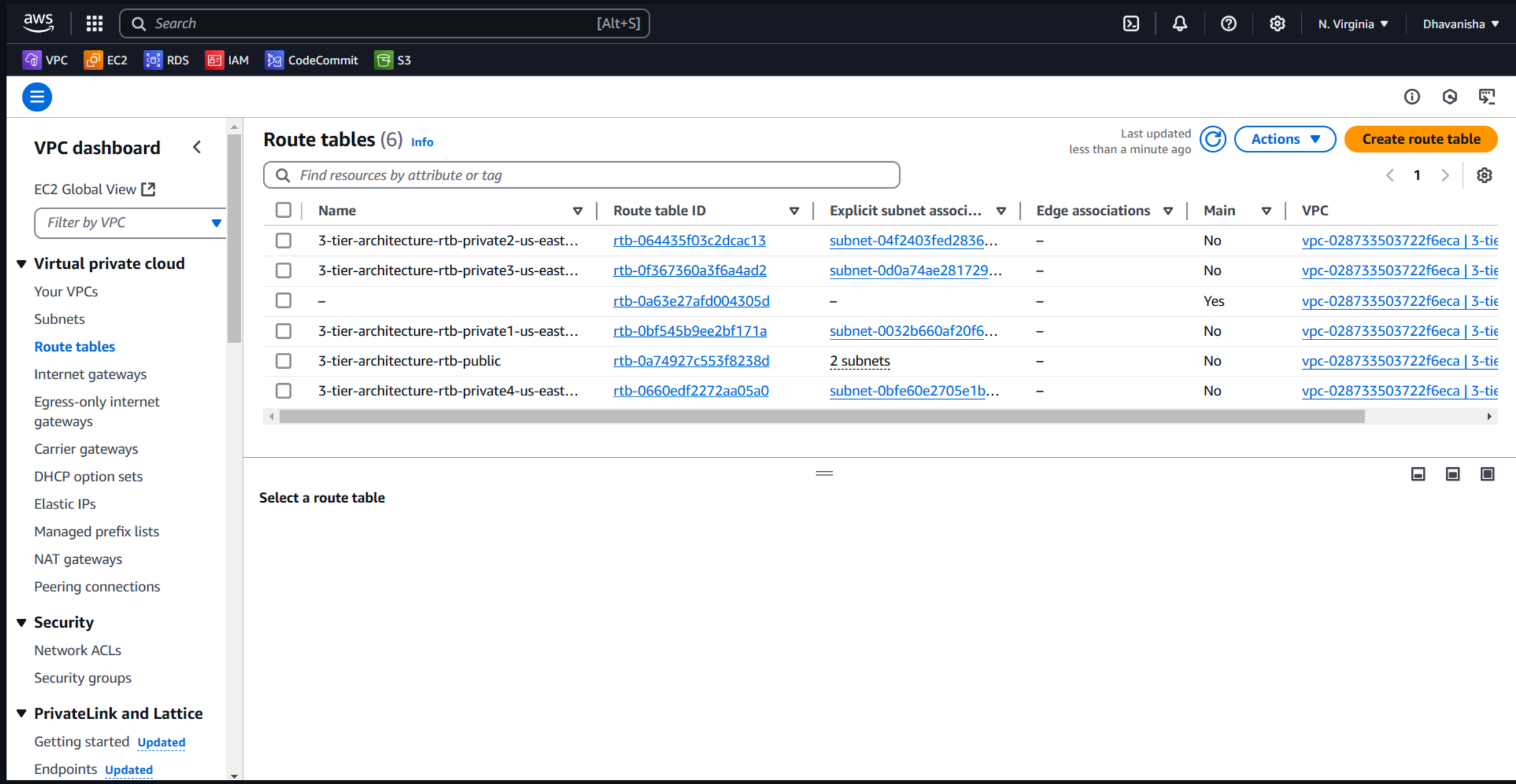
ARCHITECTURE OF THREE-TIER



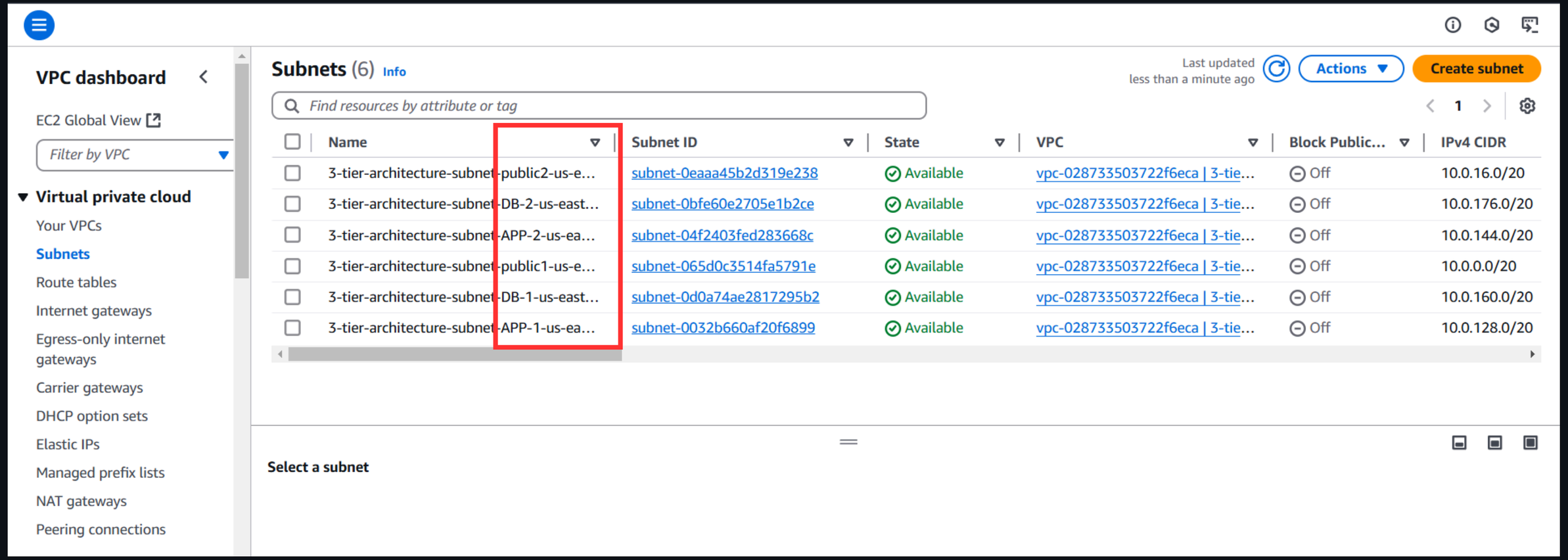
3-tier architecture VPC setup in AWS with public and private subnets across two availability zones, route tables, an internet gateway, and a NAT gateway for secure and scalable infrastructure.



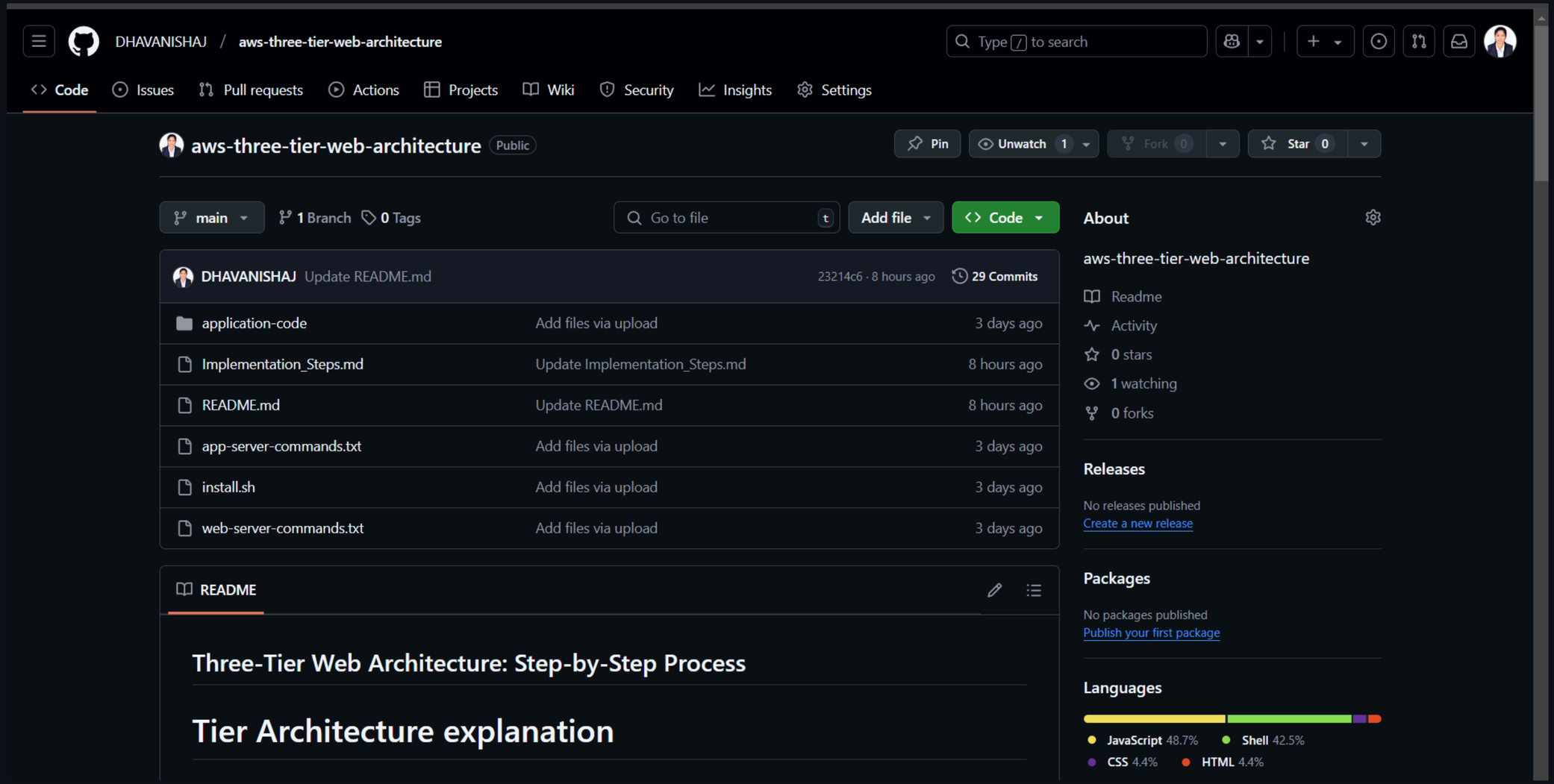

aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
VPC dashboard
EC2 Global View
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Your VPCs (1) Info
Last updated 1 minute ago
Actions
Create VPC
Search
1
Name
VPC ID
State
Block Public...
IPv4 CIDR
IPv6 CIDR
3-tier-architecture-vpc
vpc-028733503722f6eca
Available
Off
10.0.0.0/16
–
Select a VPC above


aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
VPC dashboard
EC2 Global View
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Route tables (6) Info
Last updated less than a minute ago
Actions
Create route table
Find resources by attribute or tag
1
Name
Route table ID
Explicit subnet associ...
Edge associations
Main
VPC
3-tier-architecture-rtb-private2-us-east...
rtb-064435f03c2dcac13
subnet-04f2403fed2836...
–
No
vpc-028733503722f6eca | 3-tie
3-tier-architecture-rtb-private3-us-east...
rtb-0f367360a3f6a4ad2
subnet-0d0a74ae281729...
–
No
vpc-028733503722f6eca | 3-tie
–
rtb-0a63e27afd004305d
–
–
Yes
vpc-028733503722f6eca | 3-tie
3-tier-architecture-rtb-private1-us-east...
rtb-0bf545b9ee2bf171a
subnet-0032b660af20f6...
–
No
vpc-028733503722f6eca | 3-tie
3-tier-architecture-rtb-public
rtb-0a74927c553f8238d
2 subnets
–
No
vpc-028733503722f6eca | 3-tie
3-tier-architecture-rtb-private4-us-east...
rtb-0660edf2272aa05a0
subnet-0bfe60e2705e1b...
–
No
vpc-028733503722f6eca | 3-tie
Select a route table

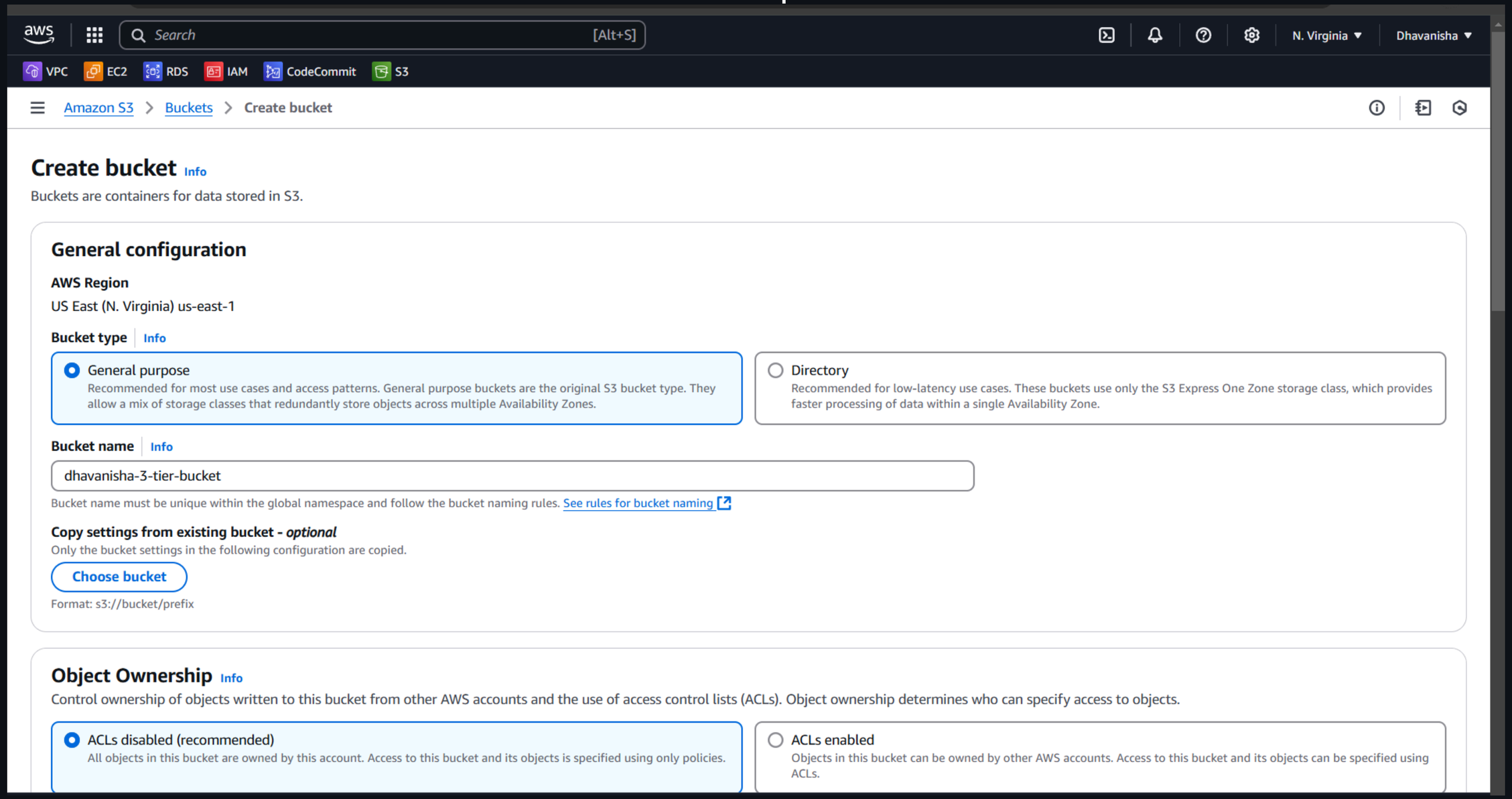
I modified private1,2 to APP-1 AND APP-2 for the application tier, private3,4 to DB1,2 for the database tier, and public for the web tier.

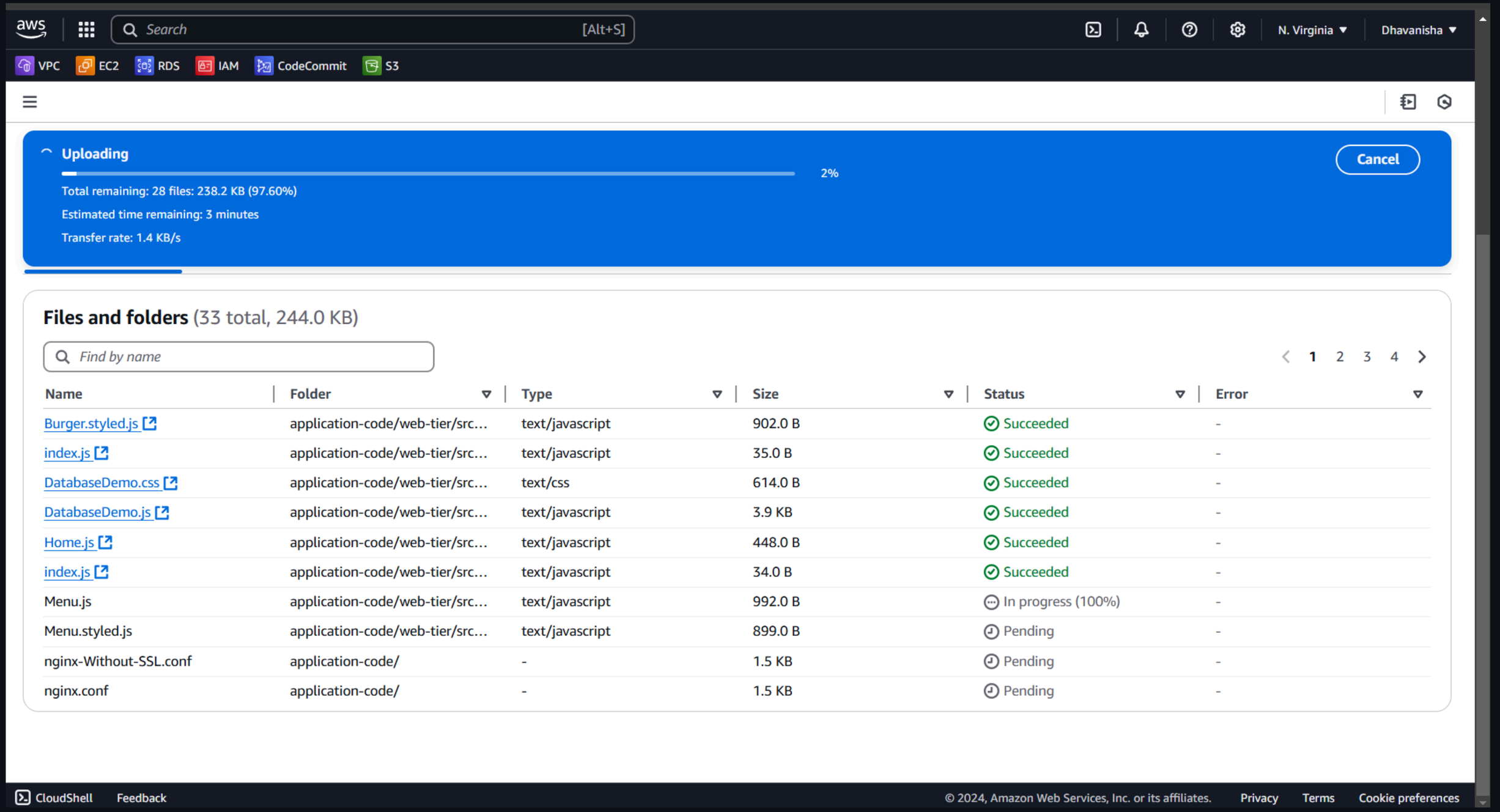


I transferred my files from git[https://github.com/DHAVANISHAJ/aws-three-tier-web-architecture.git]to s3 bucket. The files contain javascript, CSS, HTML, and shell.



Create an S3 bucket and upload a file from Git.




aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Uploading
2%
Cancel
Total remaining: 28 files: 238.2 KB (97.60%)
Estimated time remaining: 3 minutes
Transfer rate: 1.4 KB/s
Files and folders (33 total, 244.0 KB)
Find by name
1 2 3 4
Name
Folder
Type
Size
Status
Error
Burger.styled.js application-code/web-tier/src... text/javascript 902.0 B Succeeded -
index.js application-code/web-tier/src... text/javascript 35.0 B Succeeded -
DatabaseDemo.css application-code/web-tier/src... text/css 614.0 B Succeeded -
DatabaseDemo.js application-code/web-tier/src... text/javascript 3.9 KB Succeeded -
Home.js application-code/web-tier/src... text/javascript 448.0 B Succeeded -
index.js application-code/web-tier/src... text/javascript 34.0 B Succeeded -
Menu.js application-code/web-tier/src... text/javascript 992.0 B In progress (100%) -
Menu.styled.js application-code/web-tier/src... text/javascript 899.0 B Pending -
nginx-Without-SSL.conf application-code/ - 1.5 KB Pending -
nginx.conf application-code/ - 1.5 KB Pending -
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

dhavanisha-3-tier-bucket Info

Objects | Metadata - *Preview* | Properties | Permissions | Metrics | Management | Access Points

Objects (1) Info

Copy S3 URI | Copy URL | Download | Open | Delete | Actions | Create folder | Upload

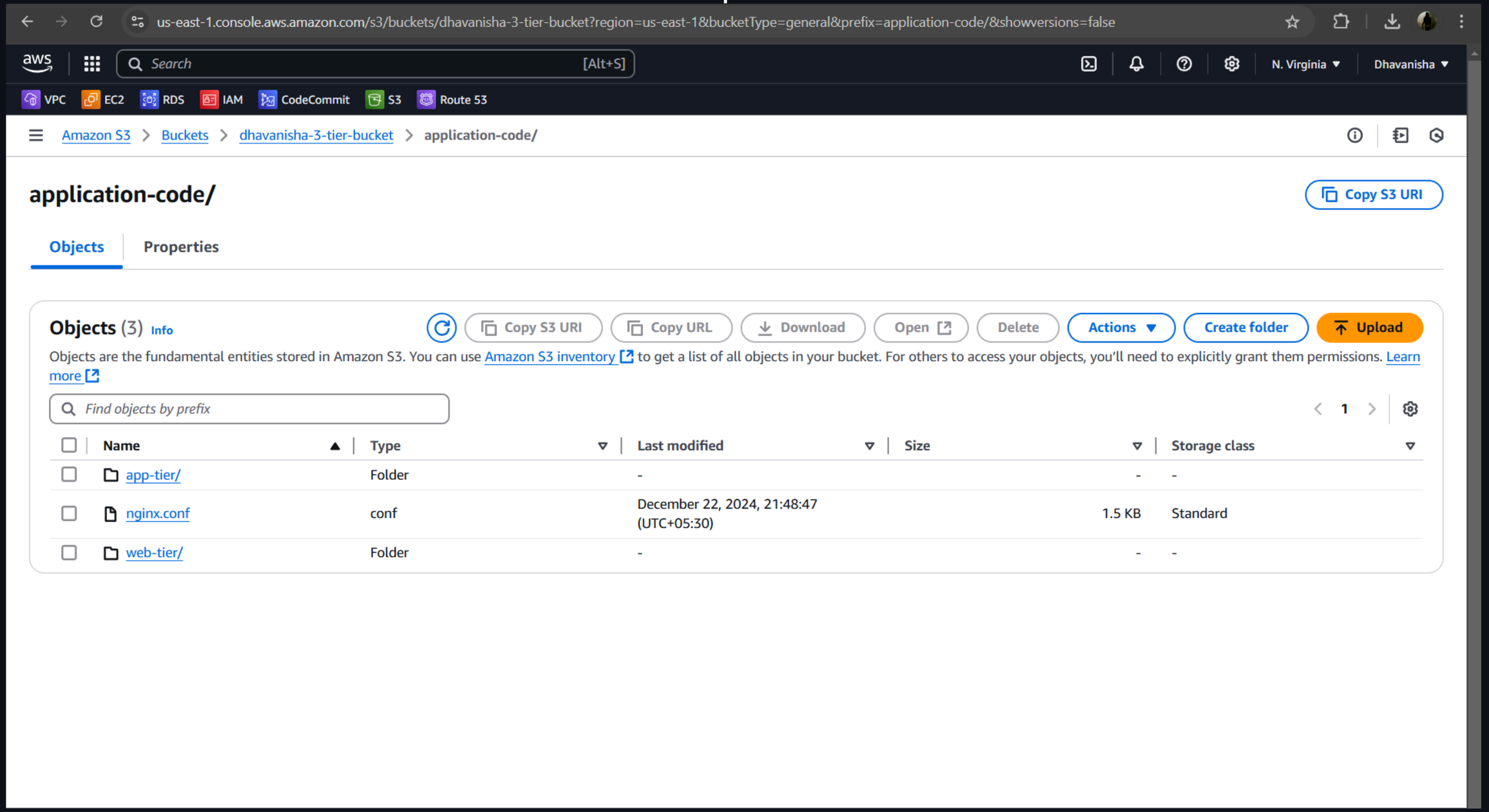
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Find objects by prefix

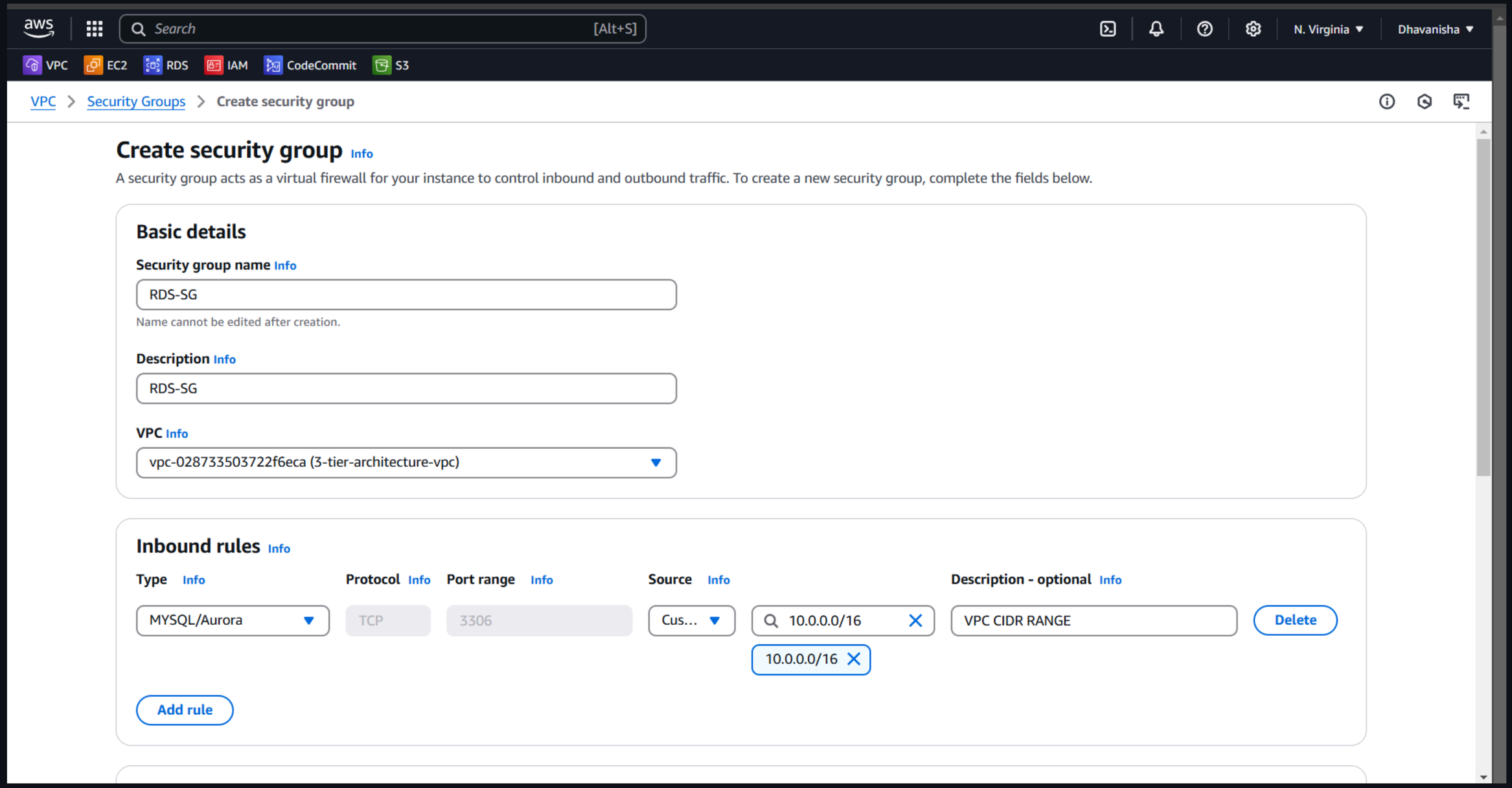
1

Name	Type	Last modified	Size	Storage class
application-code/	Folder	-	-	-

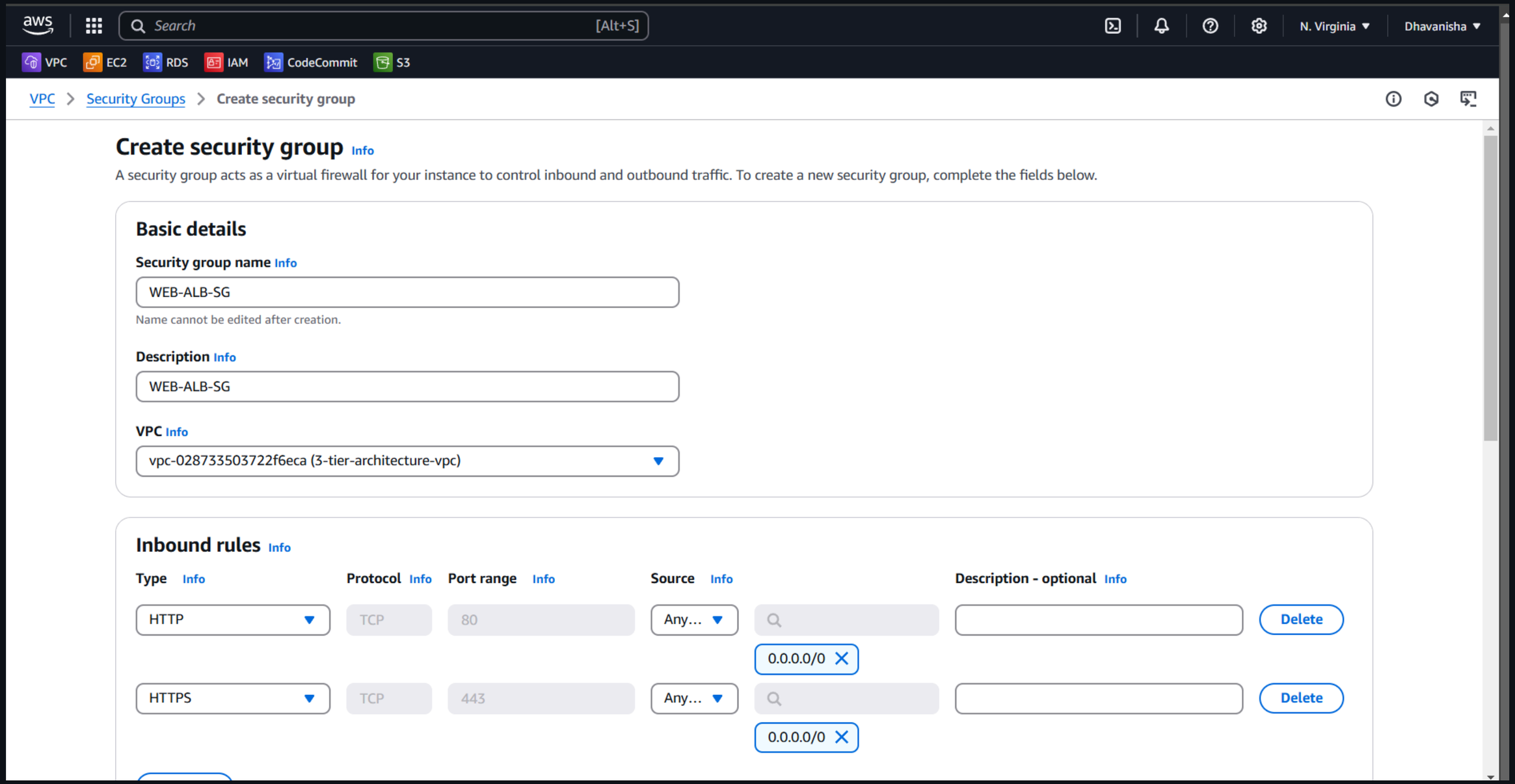
Files are uploaded.



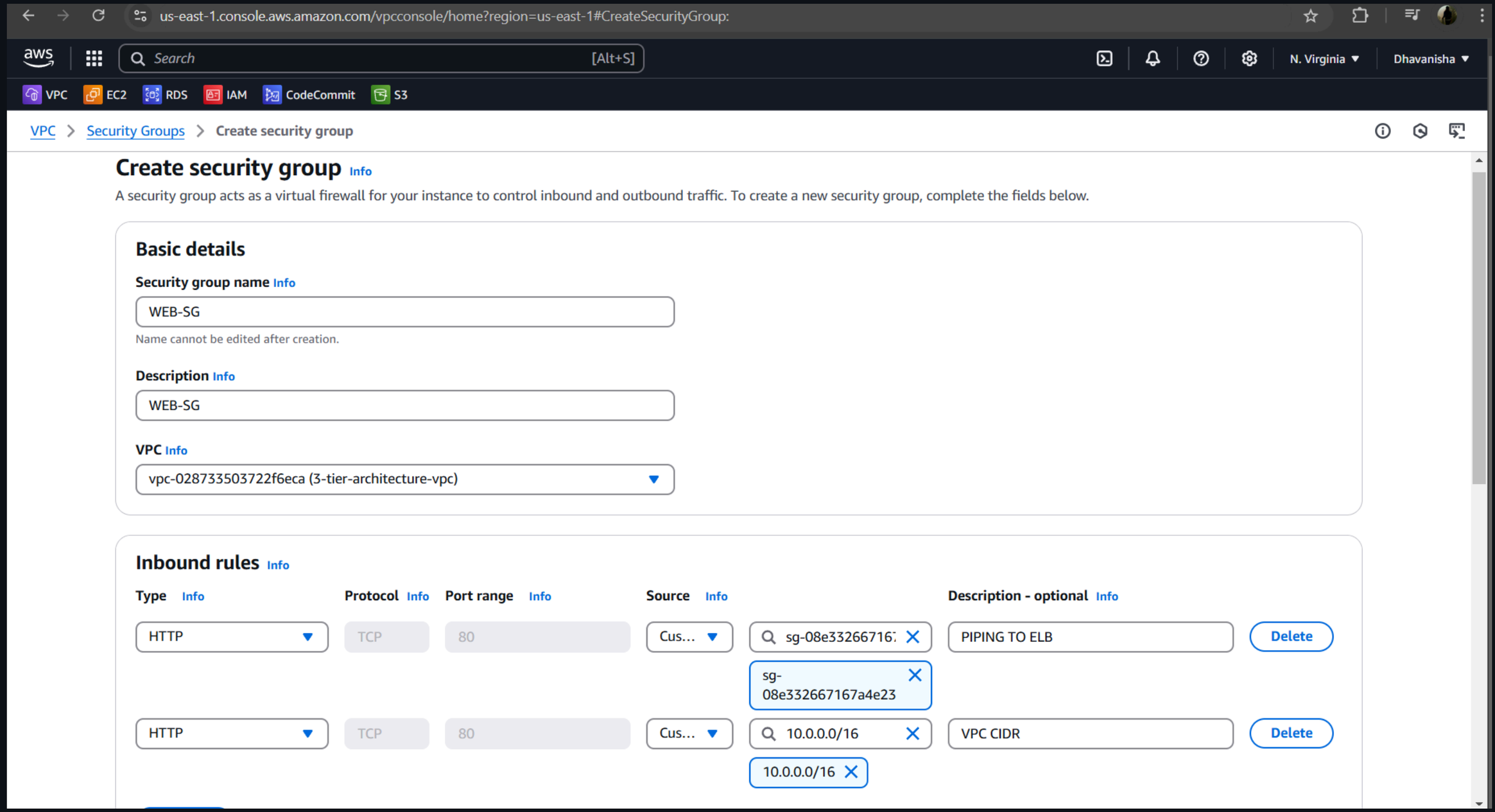
I created a security group for database (rds).



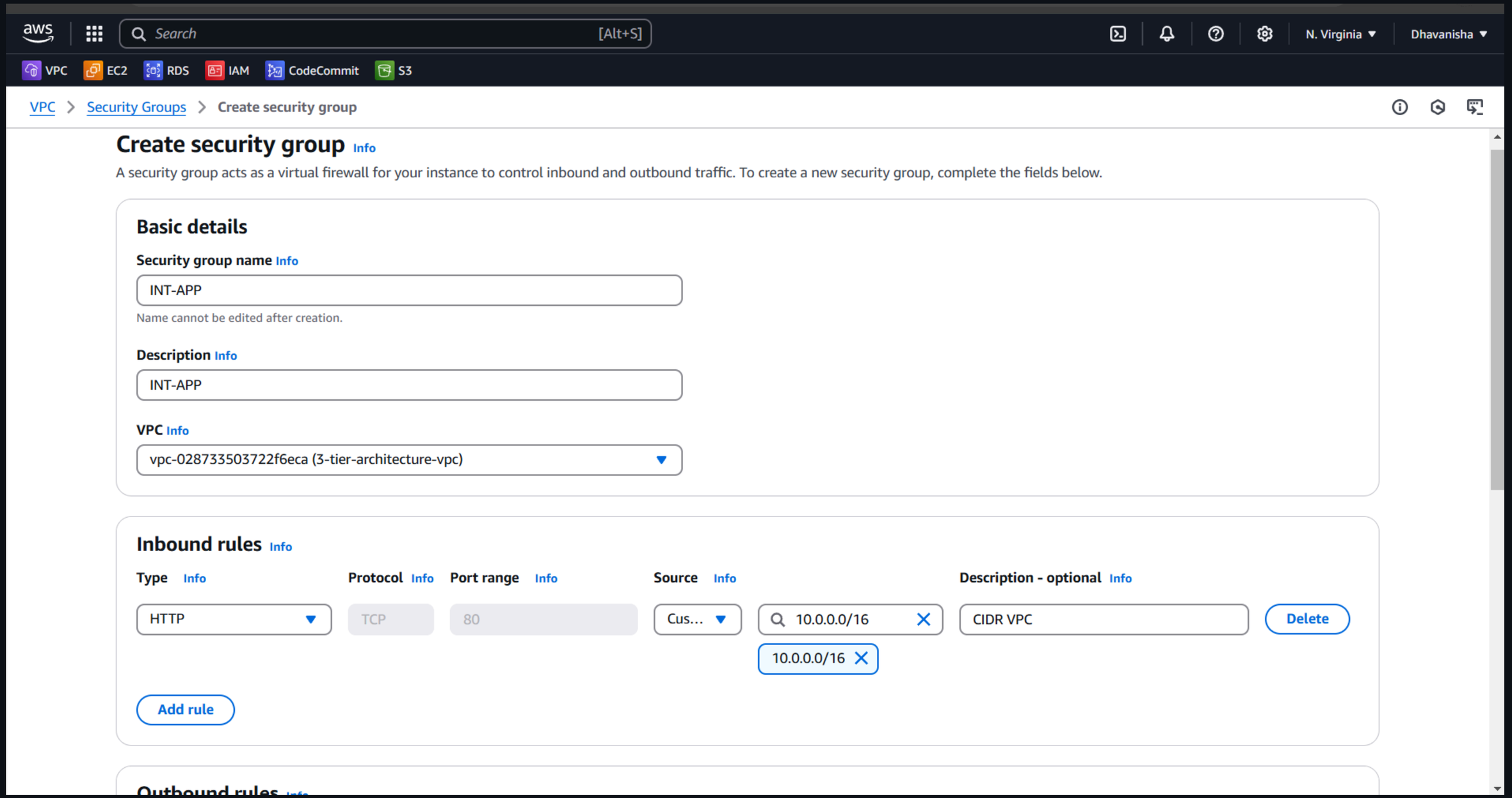
Created a security group for External load balancer.



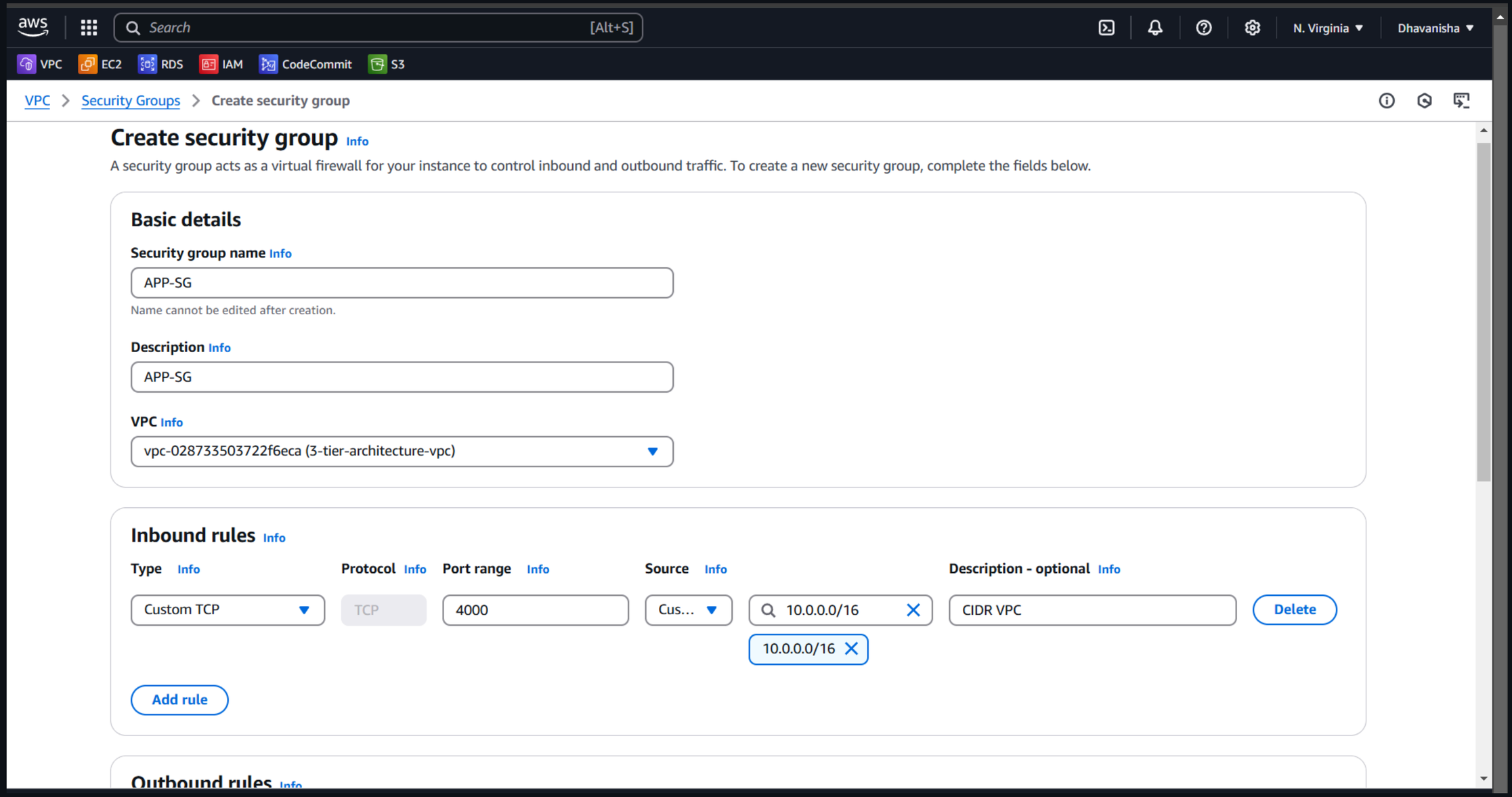
Created a security group for Web Tier(instance).

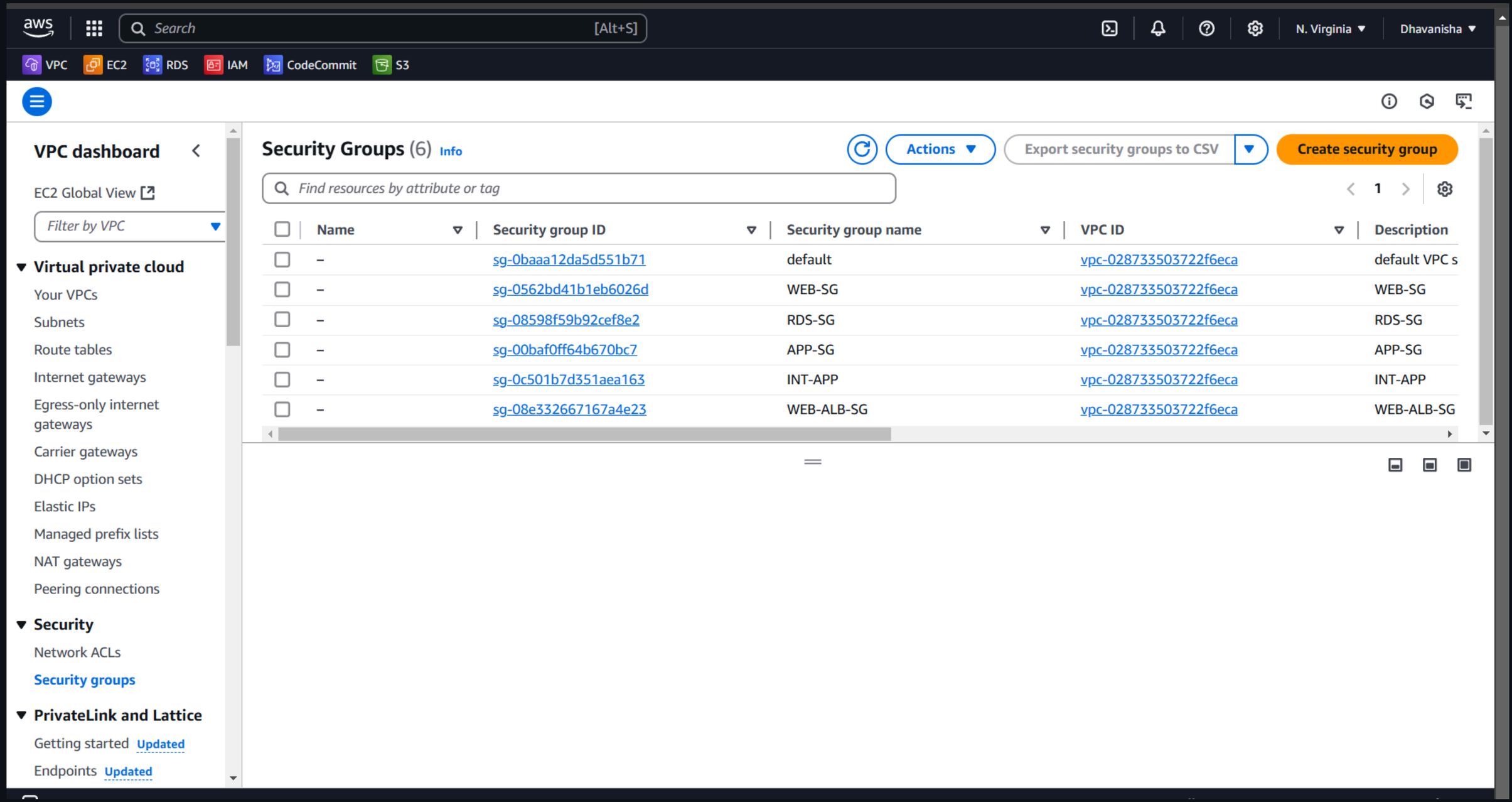


Created a security group for Internal load balancer.

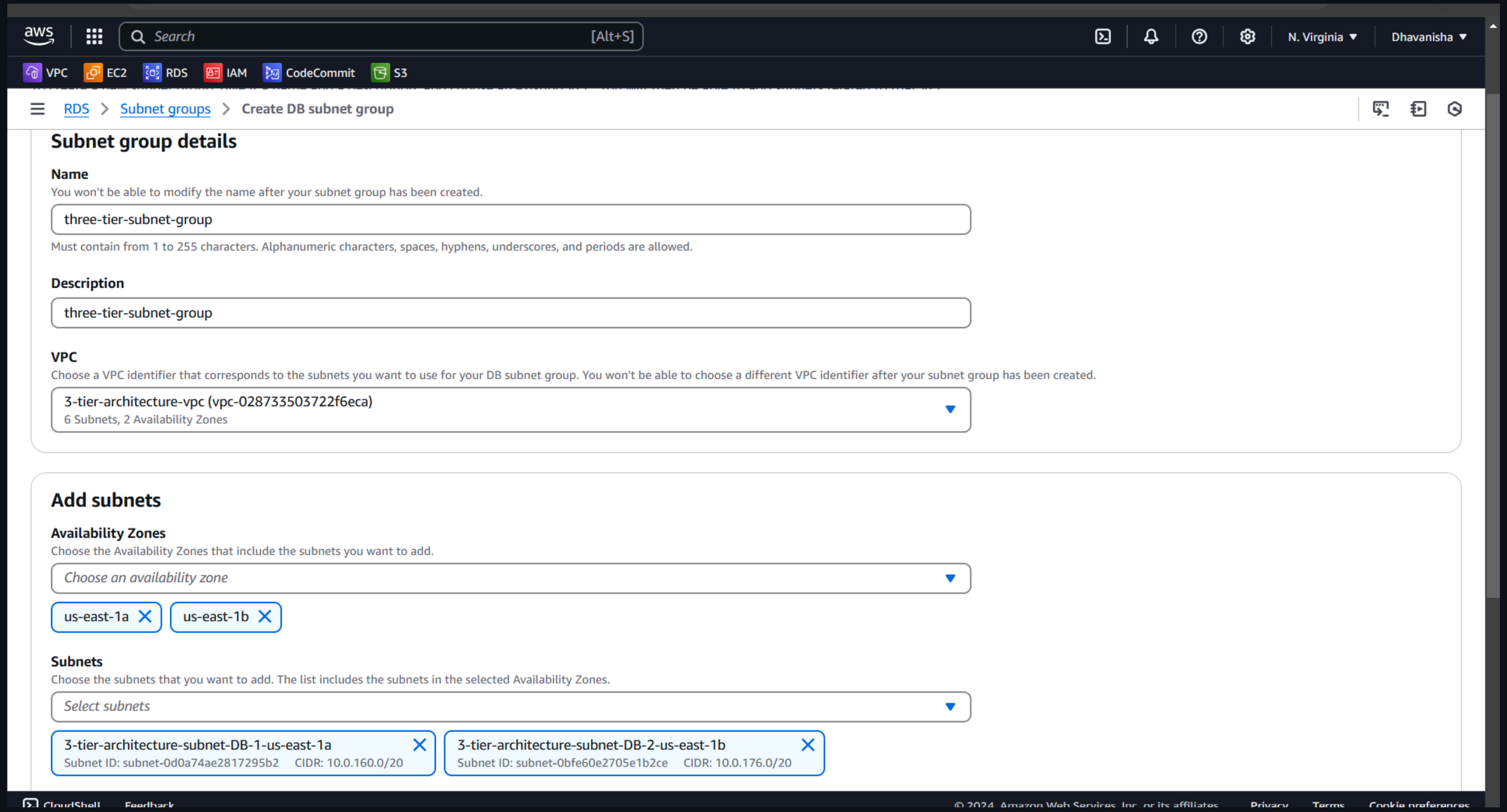


Created a security group for App tier(instance).

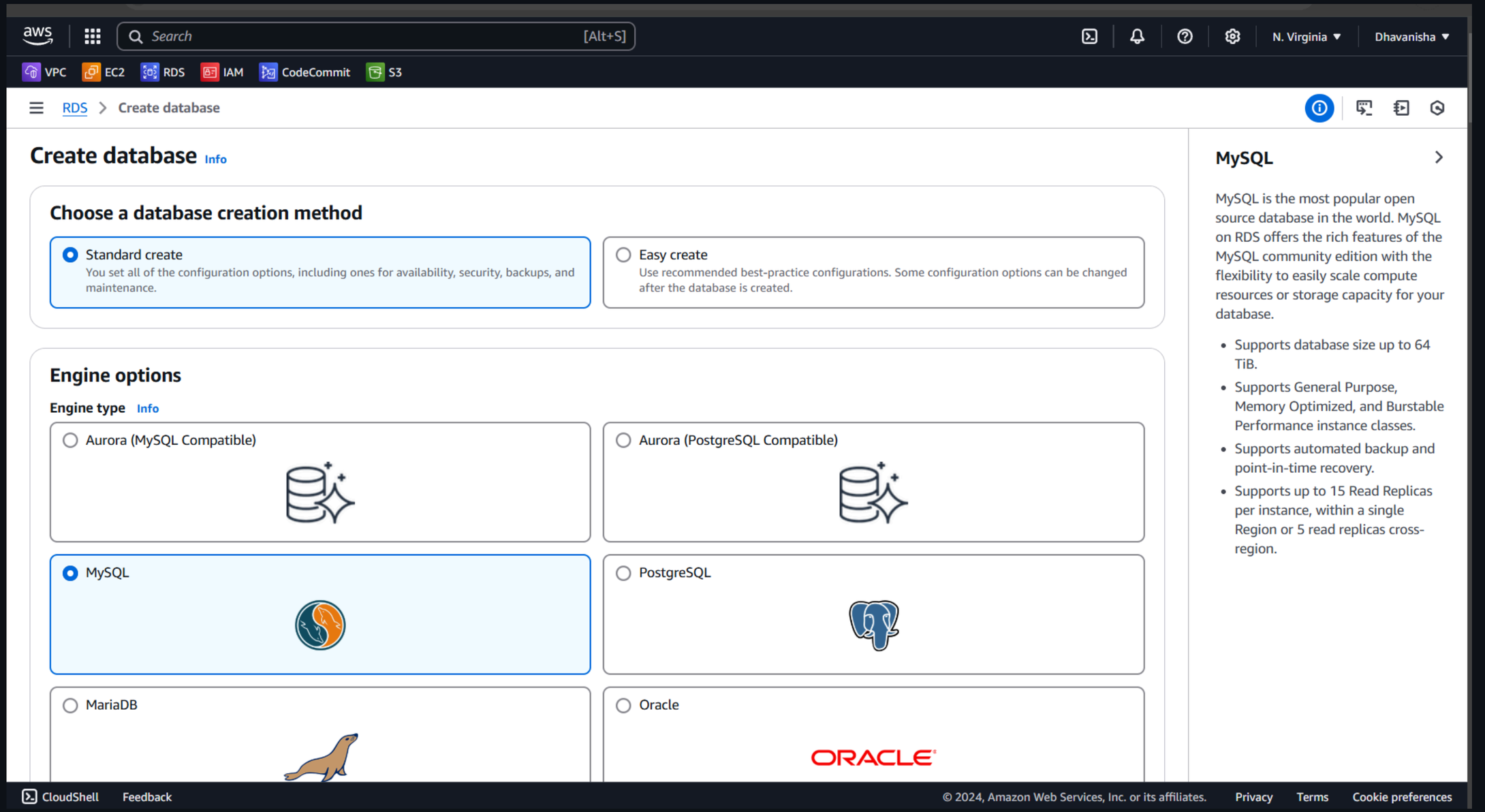



aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
VPC dashboard
EC2 Global View
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Security Groups (6) Info
Actions
Export security groups to CSV
Create security group
Find resources by attribute or tag
1
Name
Security group ID
Security group name
VPC ID
Description
– sg-0baaa12da5d551b71 default vpc-028733503722f6eca default VPC s
– sg-0562bd41b1eb6026d WEB-SG vpc-028733503722f6eca WEB-SG
– sg-08598f59b92cef8e2 RDS-SG vpc-028733503722f6eca RDS-SG
– sg-00baf0ff64b670bc7 APP-SG vpc-028733503722f6eca APP-SG
– sg-0c501b7d351aea163 INT-APP vpc-028733503722f6eca INT-APP
– sg-08e332667167a4e23 WEB-ALB-SG vpc-028733503722f6eca WEB-ALB-SG

Created a subnet group to attach in the RDs database.



After creating a subnet group, navigate to database and build a database.



Settings

DB instance identifier Info

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

three-tier-database

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Credentials Settings

Master username Info

Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

- ○ Managed in AWS Secrets Manager - *most secure*
  RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.
- ◉ Self managed
  Create your own password or have RDS create a password that you manage.

☐ Auto generate password

Amazon RDS can generate a password for you, or you can specify your own password.

Master password Info

••••••••••••

Password strength Very strong

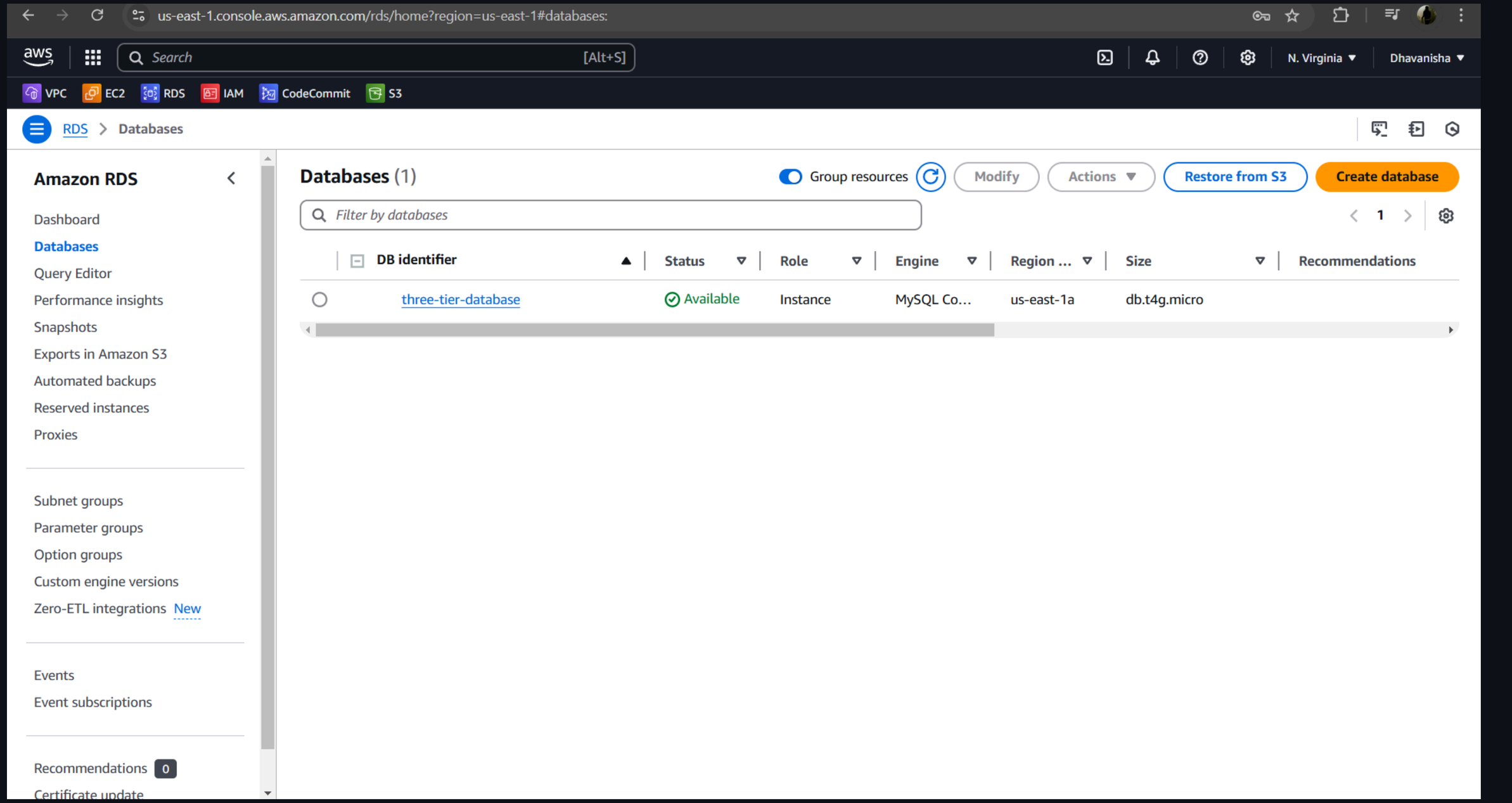
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @

Confirm master password Info

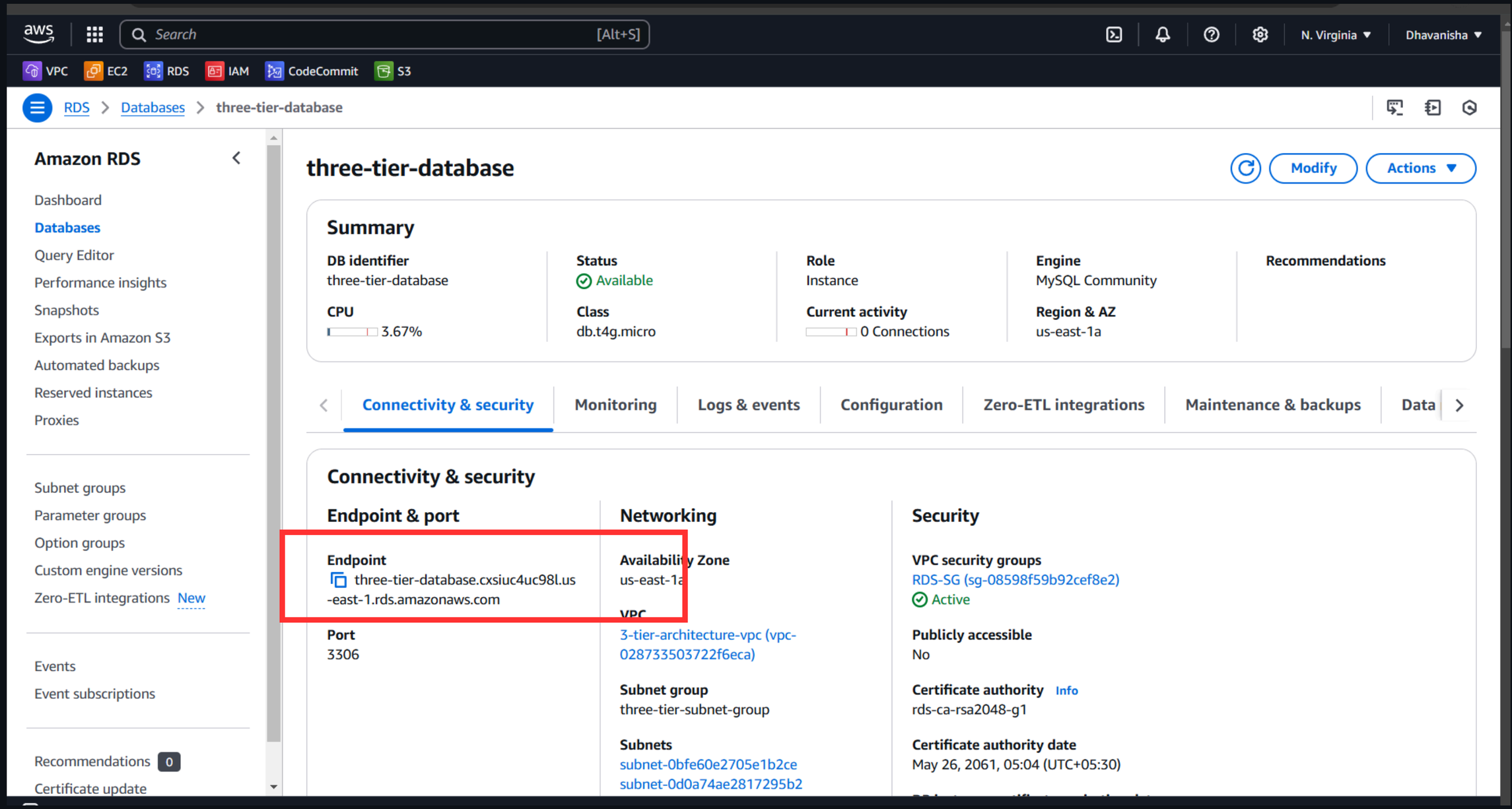
MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

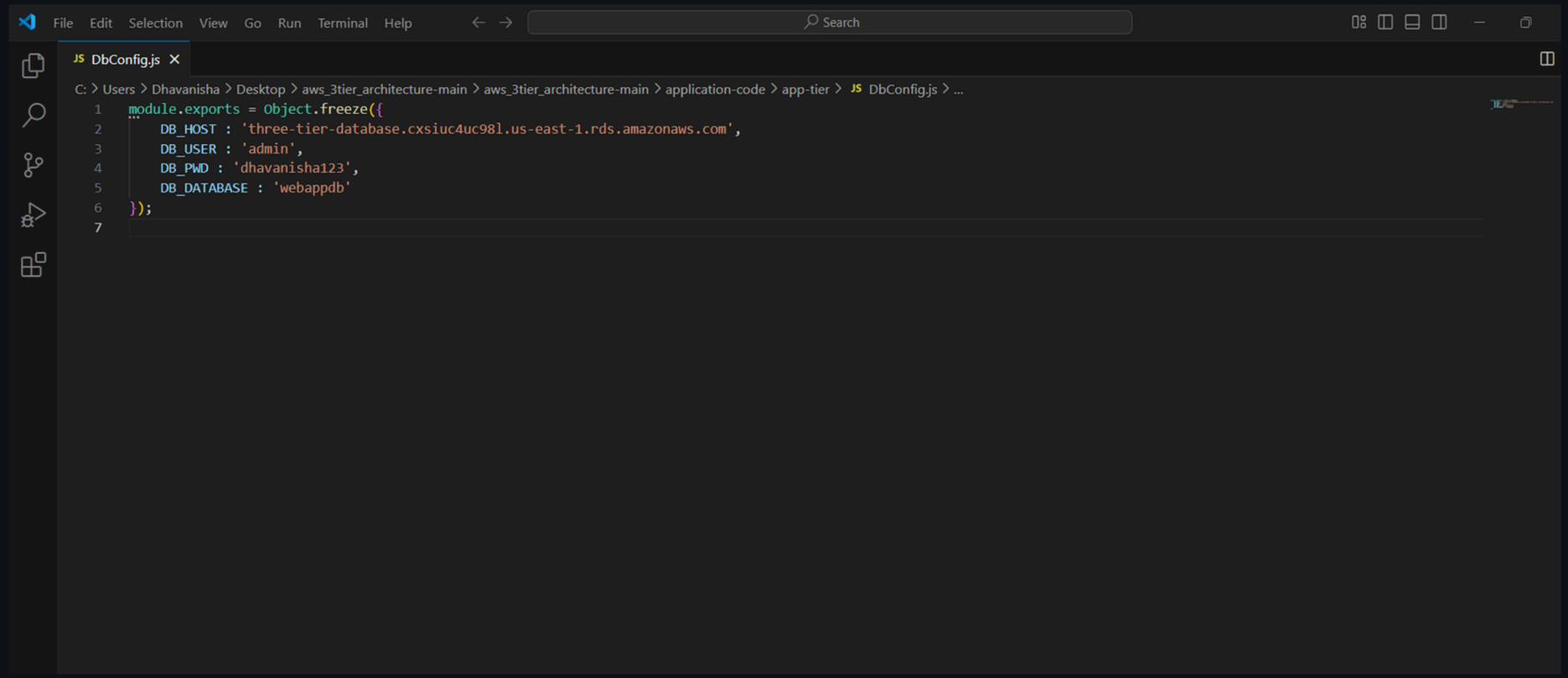
- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.


us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databases:
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
RDS
Databases
Amazon RDS
Dashboard
Databases
Query Editor
Performance insights
Snapshots
Exports in Amazon S3
Automated backups
Reserved instances
Proxies
Subnet groups
Parameter groups
Option groups
Custom engine versions
Zero-ETL integrations
New
Events
Event subscriptions
Recommendations
0
Certificate update
Databases (1)
Group resources
Modify
Actions
Restore from S3
Create database
Filter by databases
1
DB identifier
Status
Role
Engine
Region ...
Size
Recommendations
three-tier-database
Available
Instance
MySQL Co...
us-east-1a
db.t4g.micro

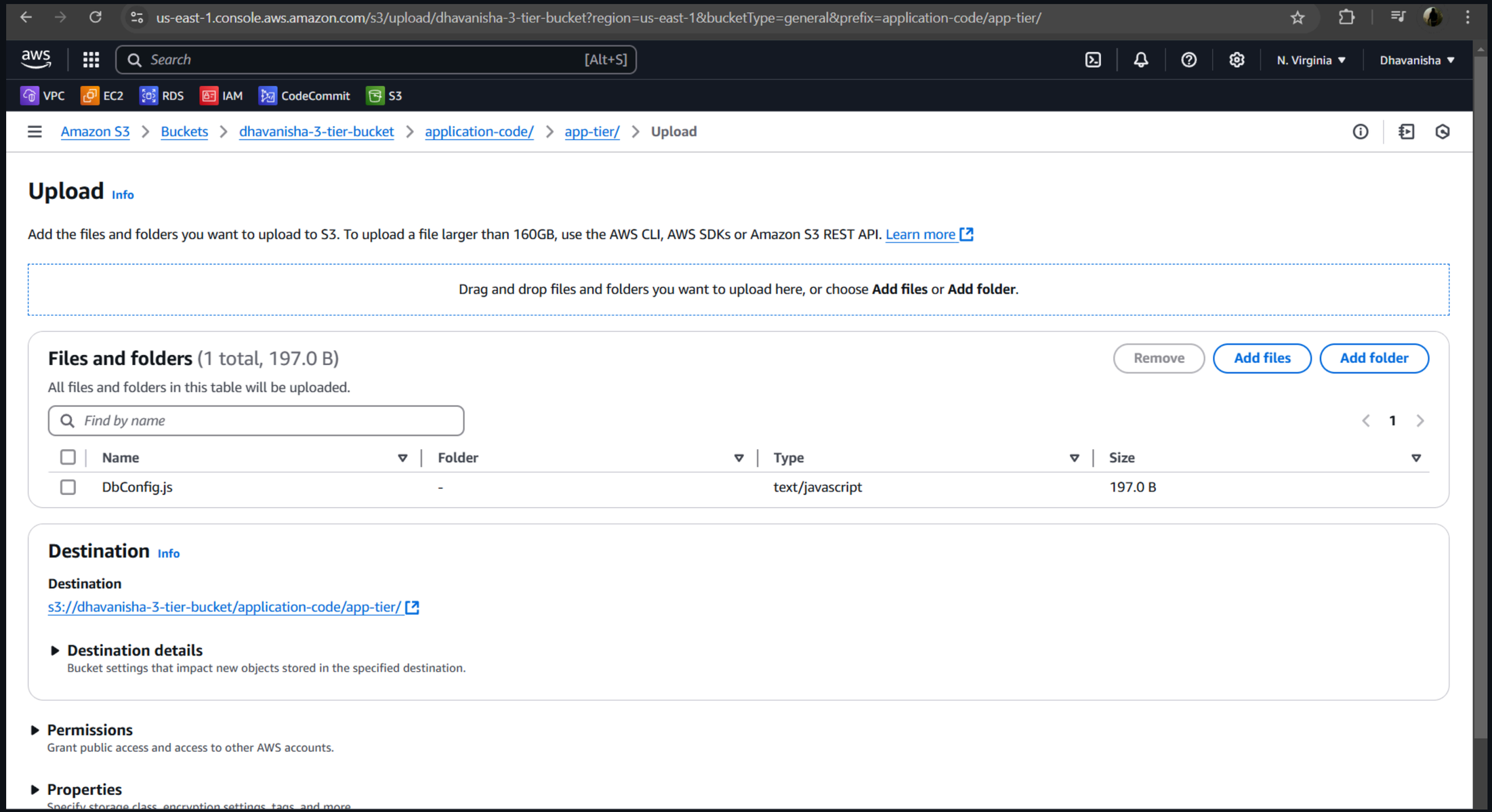
RDS and endpoints were successfully created.



Copy and paste the endpoint in the Dbconfig file where you have already uploaded in the S3 bucket the path was application code/app-tier/.



updated the Dbconfig file in the s3 bucket



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.

Service or use case

EC2

Choose a use case for the specified service.

Use case

- **EC2**
  Allows EC2 instances to call AWS services on your behalf.
- **EC2 Role for AWS Systems Manager**
  Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Add permissions Info

Permissions policies (1/1026) Info

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

Filter by Type

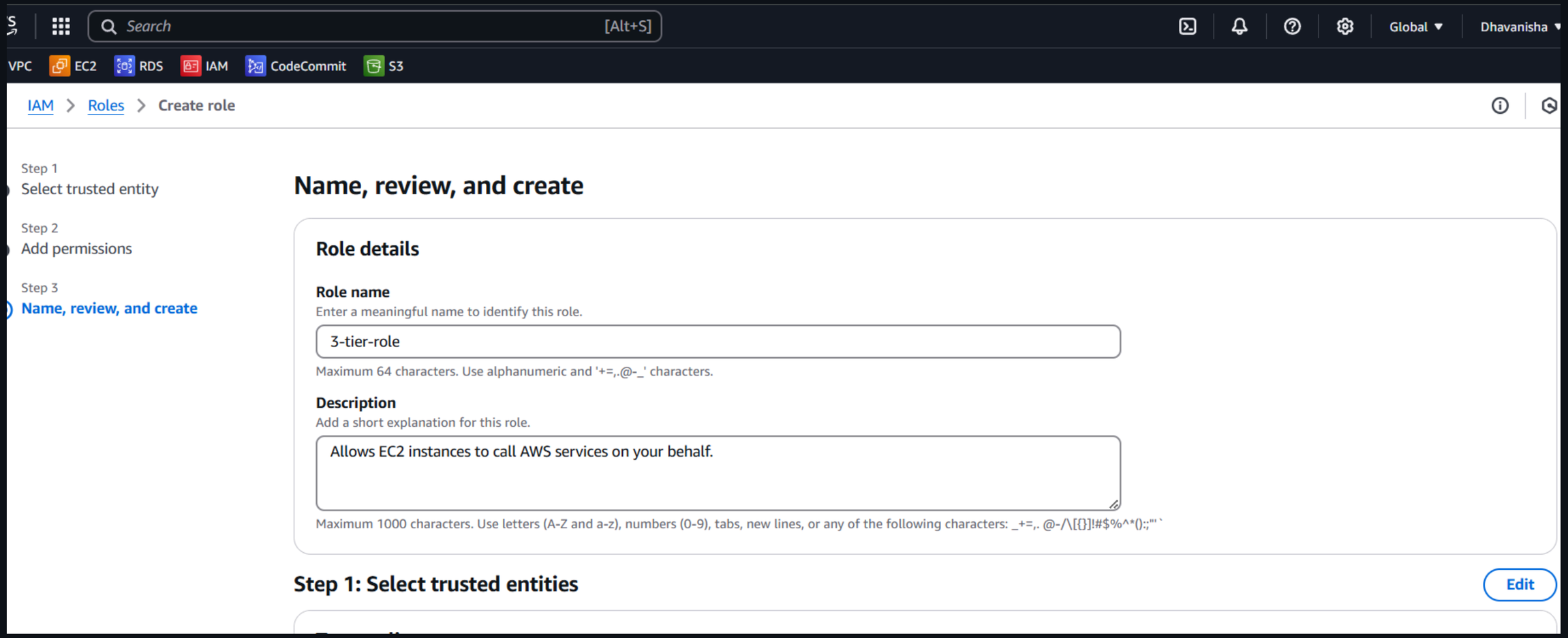
ec2roleforss | All types | 1 match | 1

Policy name	Type	Description
AmazonEC2RoleforSSM	AWS managed	This policy will soon be deprecated. Pl…

▶ **Set permissions boundary - *optional***

Cancel | Previous | Next

Create a role for session manager to have access.



Launch an instance Info

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

Name and tags Info

Name

APP-TIER-INSTANCE

Add additional tags

Application and OS Images (Amazon Machine Image) Info

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

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

Recents | Quick Start

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE Linux | D

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type — Free tier eligible

ami-0166fe664262f664c (64-bit (x86)) / ami-07bc5cc4add81dad9 (64-bit (Arm))

Summary

Number of instances Info

1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-0166fe664262f664c

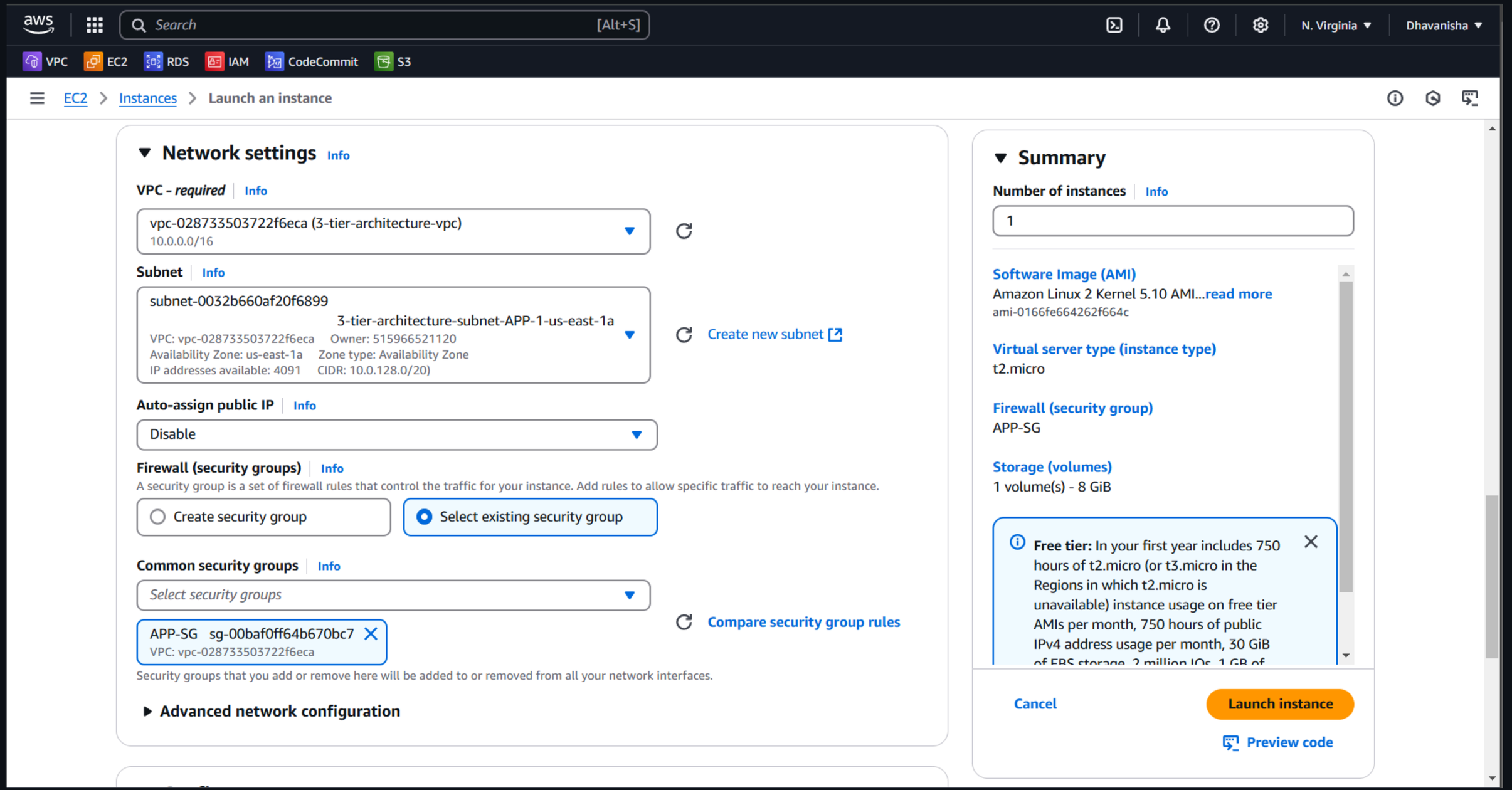
Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

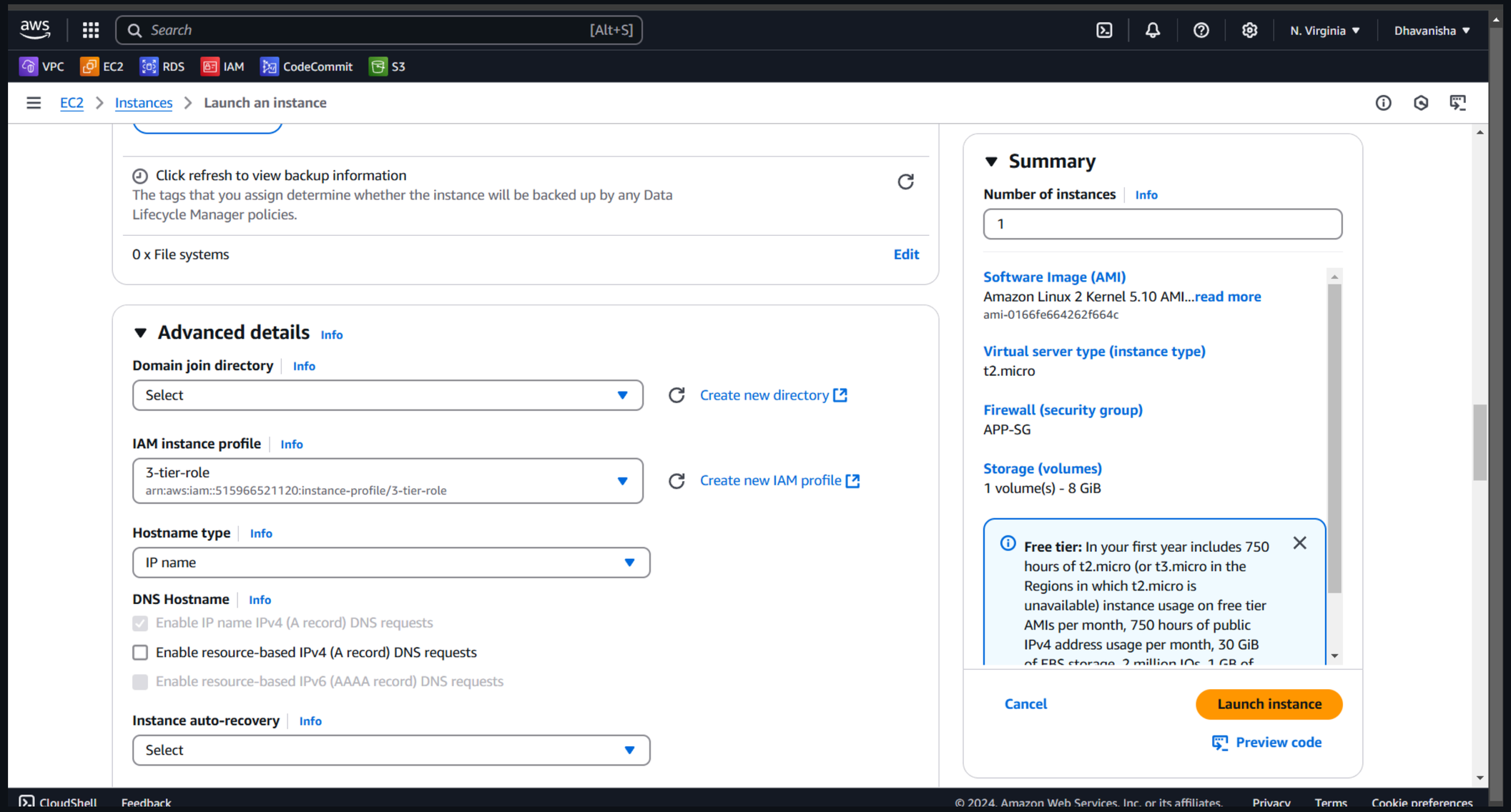
Storage (volumes)
1 volume(s) - 8 GiB

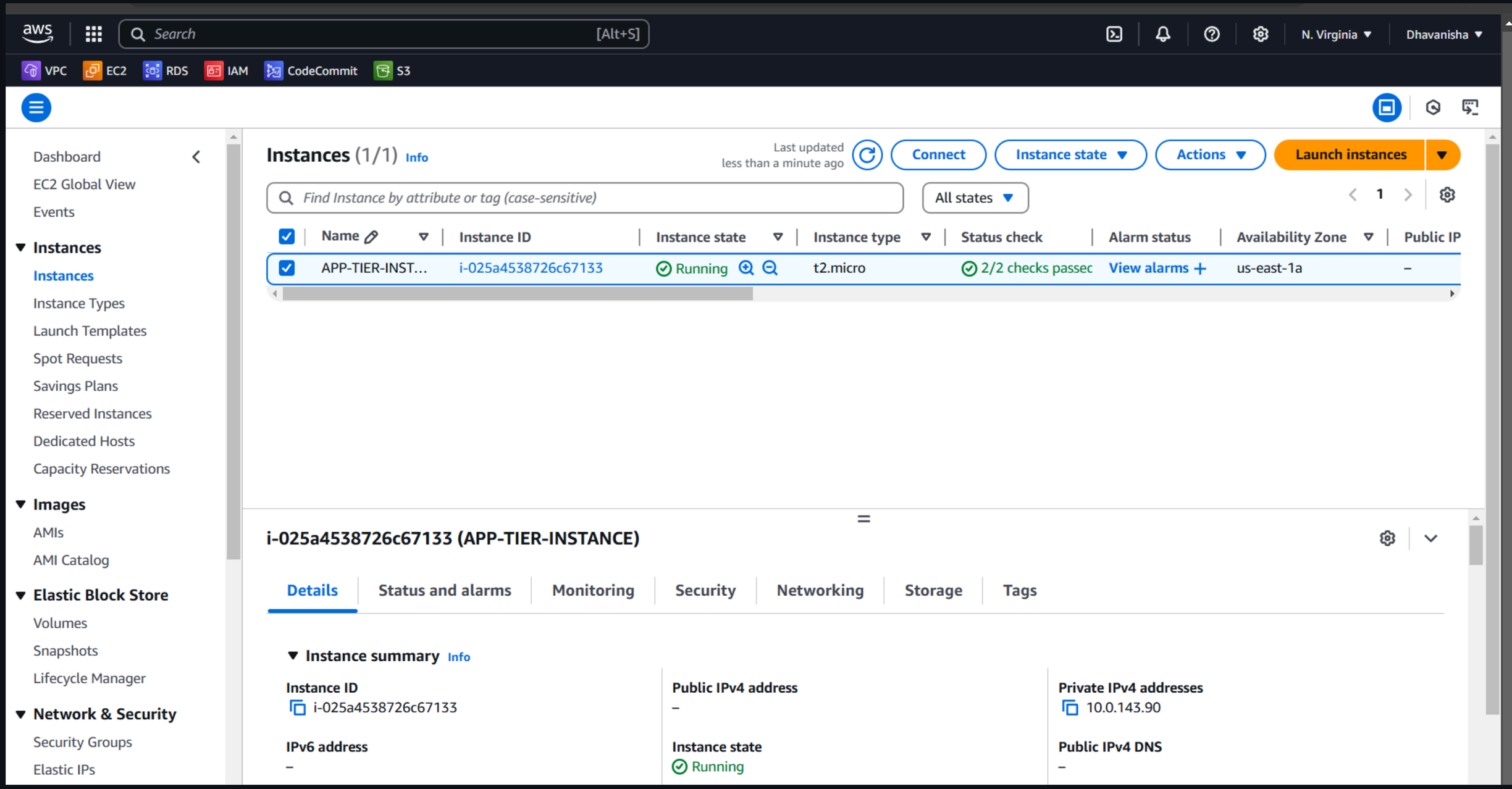
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of

Cancel | Launch instance | Preview code


aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
EC2 > Instances > Launch an instance
Network settings Info
VPC - required Info
vpc-028733503722f6eca (3-tier-architecture-vpc)
10.0.0.0/16
Subnet Info
subnet-0032b660af20f6899
3-tier-architecture-subnet-APP-1-us-east-1a
VPC: vpc-028733503722f6eca Owner: 515966521120
Availability Zone: us-east-1a Zone type: Availability Zone
IP addresses available: 4091 CIDR: 10.0.128.0/20)
Create new subnet
Auto-assign public IP Info
Disable
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
Common security groups Info
Select security groups
APP-SG sg-00baf0ff64b670bc7
VPC: vpc-028733503722f6eca
Compare security group rules
Security groups that you add or remove here will be added to or removed from all your network interfaces.
Advanced network configuration
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-0166fe664262f664c
Virtual server type (instance type)
t2.micro
Firewall (security group)
APP-SG
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, 750 hours of public IPv4 address usage per month, 30 GiB
Cancel
Launch instance
Preview code

Launch an instance for the app tier.




aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Instances (1/1) Info
Last updated
less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IP
APP-TIER-INST...
i-025a4538726c67133
Running
t2.micro
2/2 checks passec
View alarms
us-east-1a
–
i-025a4538726c67133 (APP-TIER-INSTANCE)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-025a4538726c67133
IPv6 address
–
Public IPv4 address
–
Instance state
Running
Private IPv4 addresses
10.0.143.90
Public IPv4 DNS
–

Connect to instance Info

Connect to your instance i-025a4538726c67133 (APP-TIER-INSTANCE) using any of these options

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager Preferences page.

Cancel Connect

Session ID: root-4pndfsrbff64bpy3vvf6u4sxv4 Instance ID: i-025a4538726c67133 Terminate

```
sh-4.2$ sudo su
[root@ip-10-0-143-90 bin]# whoami
root
[root@ip-10-0-143-90 bin]# cd /home/ec2-user
[root@ip-10-0-143-90 ec2-user]# sudo yum install mysql -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                | 3.6 kB  00:00:00
Resolving Dependencies
--> Running transaction check
---> Package mariadb.x86_64 1:5.5.68-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================================================================================
 Package                          Arch                          Version                                Repository                          Size
================================================================================================================================================
Installing:
 mariadb                          x86_64                        1:5.5.68-1.amzn2.0.1                   amzn2-core                         8.8 M

Transaction Summary
================================================================================================================================================
Install  1 Package

Total download size: 8.8 M
Installed size: 49 M
Downloading packages:
mariadb-5.5.68-1.amzn2.0.1.x86_64.rpm                                     | 8.8 MB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : 1:mariadb-5.5.68-1.amzn2.0.1.x86_64                                          1/1
  Verifying  : 1:mariadb-5.5.68-1.amzn2.0.1.x86_64                                          1/1

Installed:
  mariadb.x86_64 1:5.5.68-1.amzn2.0.1

Complete!
[root@ip-10-0-143-90 ec2-user]#
```

I opened a mysqlclient with rds endpoint and password.

```
[root@ip-10-0-143-90 ec2-user]# mysql -h three-tier-database.cxsiuc4uc981.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 34
Server version: 8.0.39 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> CREATE DATABASE webappdb;
Query OK, 1 row affected (0.02 sec)

MySQL [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| webappdb           |
+--------------------+
5 rows in set (0.00 sec)

MySQL [(none)]> USE webappdb;
Database changed
MySQL [webappdb]> CREATE TABLE IF NOT EXISTS transactions(
    ->   id INT NOT NULL AUTO_INCREMENT,
    ->   amount DECIMAL(10,2),
    ->   description VARCHAR(100),
    ->   PRIMARY KEY(id)
    -> );
Query OK, 0 rows affected (0.04 sec)

MySQL [webappdb]> SHOW TABLES;
+--------------------+
| Tables_in_webappdb |
+--------------------+
```

Session ID: root-4pndfsrbff64bpy3vvf6u4sxv4 Instance ID: i-025a4538726c67133

```
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| webappdb           |
+--------------------+
5 rows in set (0.00 sec)

MySQL [(none)]> USE webappdb;
Database changed
MySQL [webappdb]> CREATE TABLE IF NOT EXISTS transactions(
    ->   id INT NOT NULL AUTO_INCREMENT,
    ->   amount DECIMAL(10,2),
    ->   description VARCHAR(100),
    ->   PRIMARY KEY(id)
    -> );
Query OK, 0 rows affected (0.04 sec)

MySQL [webappdb]> SHOW TABLES;
+--------------------+
| Tables_in_webappdb |
+--------------------+
| transactions       |
+--------------------+
1 row in set (0.01 sec)

MySQL [webappdb]> INSERT INTO transactions (amount, description) VALUES ('400', 'Dhavanisha');
Query OK, 1 row affected (0.00 sec)

MySQL [webappdb]> SELECT * FROM transactions;
+----+--------+-------------+
| id | amount | description |
+----+--------+-------------+
|  1 | 400.00 | Dhavanisha  |
+----+--------+-------------+
1 row in set (0.00 sec)

MySQL [webappdb]>
```

```
[root@ip-10-0-143-90 ec2-user]# curl -o- https://raw.githubusercontent.com/DHAVANISHAJ/aws-three-tier-web-architecture/main/install.sh | bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 14926  100 14926    0     0   114k      0 --:--:-- --:--:-- --:--:--  114k
=> Downloading nvm as script to '/root/.nvm'

=> Appending nvm source string to /root/.bashrc
=> Appending bash_completion source string to /root/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"  # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion"  # This loads nvm bash_completion
[root@ip-10-0-143-90 ec2-user]# source ~/.bashrc
[root@ip-10-0-143-90 ec2-user]# nvm install 16
Downloading and installing node v16.20.2...
Downloading https://nodejs.org/dist/v16.20.2/node-v16.20.2-linux-x64.tar.xz...
######################################################################################################################################################################## 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v16.20.2 (npm v8.19.4)
Creating default alias: default -> 16 (-> v16.20.2)
[root@ip-10-0-143-90 ec2-user]# nvm use 16
Now using node v16.20.2 (npm v8.19.4)
[root@ip-10-0-143-90 ec2-user]# npm install -g pm2

added 137 packages, and audited 138 packages in 7s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New major version of npm available! 8.19.4 -> 11.0.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.0.0
npm notice Run npm install -g npm@11.0.0 to update!
npm notice
[root@ip-10-0-143-90 ec2-user]#
[root@ip-10-0-143-90 ec2-user]#
[root@ip-10-0-143-90 ec2-user]#
```

Session ID: root-4pndfsrbff64bpy3vvf6u4sxv4 Instance ID: i-025a4538726c67133 Terminate

```
[root@ip-10-0-143-90 ec2-user]# cd ~/
[root@ip-10-0-143-90 ~]# aws s3 cp s3://dhavanisha-3-tier-bucket/application-code/app-tier/ app-tier --recursive
download: s3://dhavanisha-3-tier-bucket/application-code/app-tier/README.md to app-tier/README.md
download: s3://dhavanisha-3-tier-bucket/application-code/app-tier/TransactionService.js to app-tier/TransactionService.js
download: s3://dhavanisha-3-tier-bucket/application-code/app-tier/package.json to app-tier/package.json
download: s3://dhavanisha-3-tier-bucket/application-code/app-tier/package-lock.json to app-tier/package-lock.json
download: s3://dhavanisha-3-tier-bucket/application-code/app-tier/index.js to app-tier/index.js
download: s3://dhavanisha-3-tier-bucket/application-code/app-tier/DbConfig.js to app-tier/DbConfig.js
[root@ip-10-0-143-90 ~]# ls
app-tier
[root@ip-10-0-143-90 ~]# cd ~/app-tier
[root@ip-10-0-143-90 app-tier]# ls
DbConfig.js  README.md  TransactionService.js  index.js  package-lock.json  package.json
[root@ip-10-0-143-90 app-tier]# npm install

added 68 packages, and audited 69 packages in 2s

2 packages are looking for funding
  run `npm fund` for details

7 vulnerabilities (3 low, 4 high)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
[root@ip-10-0-143-90 app-tier]# cat DbConfig.js
module.exports = Object.freeze({
    DB_HOST : 'three-tier-database.cxsiuc4uc98l.us-east-1.rds.amazonaws.com',
    DB_USER : 'admin',
    DB_PWD : 'dhavanisha123',
    DB_DATABASE : 'webappdb'
});
[root@ip-10-0-143-90 app-tier]#
```

Session ID: root-4pndfsrbff64bpy3vvf6u4sxv4 Instance ID: i-025a4538726c67133 Terminate

```
 _\/\\\\\\\\\____\/\\\__\///\\\/___\/\\\_____/\\\//_____
  _\/\\\_____\/\\\____\///_____\/\\\_____/\\\//________
   _\/\\\_____\/\\\_____________\/\\\___/\\\/___________
    _\/\\\_____\/\\\_____________\/\\\__/\\\\\\\\\\\\\\\_
     _\///______\///______________\///__\///////////////__

                      Runtime Edition

       PM2 is a Production Process Manager for Node.js applications
                    with a built-in Load Balancer.

              Start and Daemonize any application:
              $ pm2 start app.js

              Load Balance 4 instances of api.js:
              $ pm2 start api.js -i 4

              Monitor in production:
              $ pm2 monitor

              Make pm2 auto-boot at server restart:
              $ pm2 startup

              To go further checkout:
              http://pm2.io/


                      -------------

[PM2] Spawning PM2 daemon with pm2_home=/root/.pm2
[PM2] PM2 Successfully daemonized
[PM2] Starting /root/app-tier/index.js in fork_mode (1 instance)
[PM2] Done.
```

id	name	namespace	version	mode	pid	uptime	↺	status	cpu	mem	user	watching
0	index	default	1.0.0	fork	4401	0s	0	online	0%	23.6mb	root	disabled

```
[root@ip-10-0-143-90 app-tier]#
```

Session ID: root-4pndfsrbff64bpy3vvf6u4sxv4 Instance ID: i-025a4538726c67133 Terminate

```
[root@ip-10-0-143-90 app-tier]# pm2 status
```

id	name	namespace	version	mode	pid	uptime	↺	status	cpu	mem	user	watching
0	index	default	1.0.0	fork	4401	2m	0	online	0%	51.4mb	root	disabled

```
[root@ip-10-0-143-90 app-tier]# pm2 list
```

id	name	namespace	version	mode	pid	uptime	↺	status	cpu	mem	user	watching
0	index	default	1.0.0	fork	4401	3m	0	online	0%	51.7mb	root	disabled

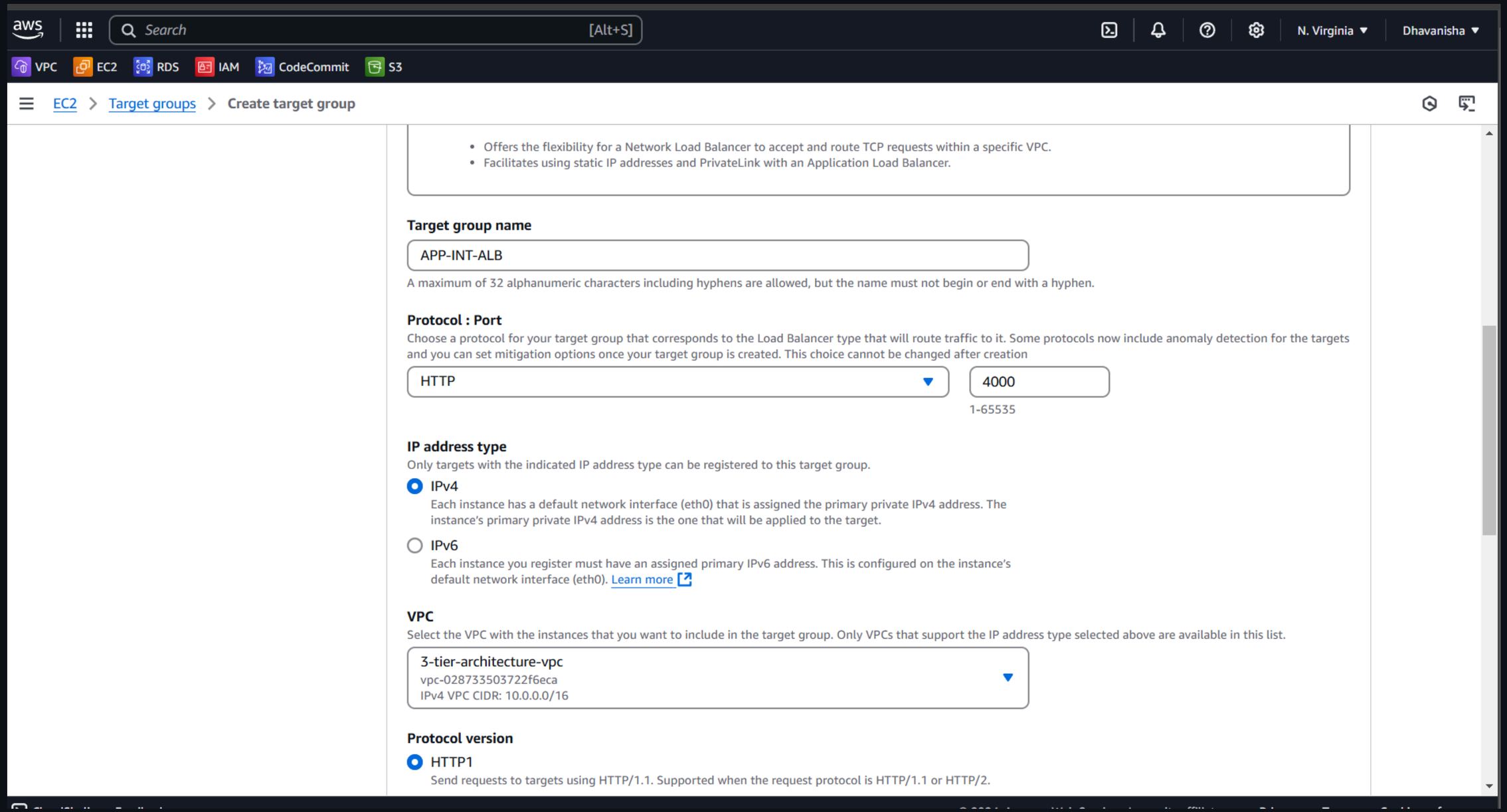
```
[root@ip-10-0-143-90 app-tier]# pm2 logs
[TAILING] Tailing last 15 lines for [all] processes (change the value with --lines option)
/root/.pm2/pm2.log last 15 lines:
PM2        | 2024-12-19T12:19:55: PM2 log: PM2 version         : 5.4.3
PM2        | 2024-12-19T12:19:55: PM2 log: Node.js version     : 16.20.2
PM2        | 2024-12-19T12:19:55: PM2 log: Current arch        : x64
PM2        | 2024-12-19T12:19:55: PM2 log: PM2 home            : /root/.pm2
PM2        | 2024-12-19T12:19:55: PM2 log: PM2 PID file        : /root/.pm2/pm2.pid
PM2        | 2024-12-19T12:19:55: PM2 log: RPC socket file     : /root/.pm2/rpc.sock
PM2        | 2024-12-19T12:19:55: PM2 log: BUS socket file     : /root/.pm2/pub.sock
PM2        | 2024-12-19T12:19:55: PM2 log: Application log path : /root/.pm2/logs
PM2        | 2024-12-19T12:19:55: PM2 log: Worker Interval     : 30000
PM2        | 2024-12-19T12:19:55: PM2 log: Process dump file   : /root/.pm2/dump.pm2
PM2        | 2024-12-19T12:19:55: PM2 log: Concurrent actions  : 2
PM2        | 2024-12-19T12:19:55: PM2 log: SIGTERM timeout     : 1600
PM2        | 2024-12-19T12:19:55: PM2 log: ================================================================================
PM2        | 2024-12-19T12:19:55: PM2 log: App [index:0] starting in -fork mode-
PM2        | 2024-12-19T12:19:55: PM2 log: App [index:0] online

/root/.pm2/logs/index-error.log last 15 lines:
/root/.pm2/logs/index-out.log last 15 lines:
0|index    | AB3 backend app listening at http://localhost:4000
```

Session ID: root-4pndfsrbff64bpy3vvf6u4sxv4 Instance ID: i-025a4538726c67133 Terminate

```
[root@ip-10-0-143-90 app-tier]# ls
DbConfig.js  README.md  TransactionService.js  index.js  node_modules  package-lock.json  package.json
[root@ip-10-0-143-90 app-tier]# cat index.js
const transactionService = require('./TransactionService');
const express = require('express');
const bodyParser = require('body-parser');
const cors = require('cors');
const os = require('os');
const fetch = require('node-fetch');

const app = express();
const port = 4000;

app.use(bodyParser.urlencoded({ extended: true }));
app.use(bodyParser.json());
app.use(cors());

// ROUTES FOR OUR API
// =============================================================

//Health Checking
app.get('/health',(req,res)=>{
    res.json("This is the health check");
});

// ADD TRANSACTION
app.post('/transaction', (req,res)=>{
    var response = "";
    try{
        console.log(req.body);
        console.log(req.body.amount);
        console.log(req.body.desc);
        var success = transactionService.addTransaction(req.body.amount,req.body.desc);
        if (success = 200) res.json({ message: 'added transaction successfully'});
    }catch (err){
        res.json({ message: 'something went wrong', error : err.message});
    }
});

// GET ALL TRANSACTIONS
```

```
After=network.target

[Service]
Type=forking
User=root
LimitNOFILE=infinity
LimitNPROC=infinity
LimitCORE=infinity
Environment=PATH=/root/.nvm/versions/node/v16.20.2/bin:/sbin:/bin:/usr/sbin:/usr/bin:/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
Environment=PM2_HOME=/root/.pm2
PIDFile=/root/.pm2/pm2.pid
Restart=on-failure

ExecStart=/root/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 resurrect
ExecReload=/root/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 reload all
ExecStop=/root/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 kill

[Install]
WantedBy=multi-user.target

Target path
/etc/systemd/system/pm2-root.service
Command list
[ 'systemctl enable pm2-root' ]
[PM2] Writing init configuration in /etc/systemd/system/pm2-root.service
[PM2] Making script booting at startup...
[PM2] [-] Executing: systemctl enable pm2-root...
Created symlink from /etc/systemd/system/multi-user.target.wants/pm2-root.service to /etc/systemd/system/pm2-root.service.
[PM2] [v] Command successfully executed.
+---------------------------------------+
[PM2] Freeze a process list on reboot via:
$ pm2 save

[PM2] Remove init script via:
$ pm2 unstartup systemd
[root@ip-10-0-143-90 app-tier]# pm2 save
[PM2] Saving current process list...
[PM2] Successfully saved in /root/.pm2/dump.pm2
[root@ip-10-0-143-90 app-tier]# curl http://localhost:4000/health
"This is the health check"[root@ip-10-0-143-90 app-tier]#
```

I setup a target group for internal load balancer.



0 selected

Ports for the selected instances

Ports for routing traffic to the selected instances.

4000

1-65535 (separate multiple ports with commas)

Include as pending below

1 selection is now pending below. Include more or register targets when ready.

Review targets

Targets (1)

Remove all pending

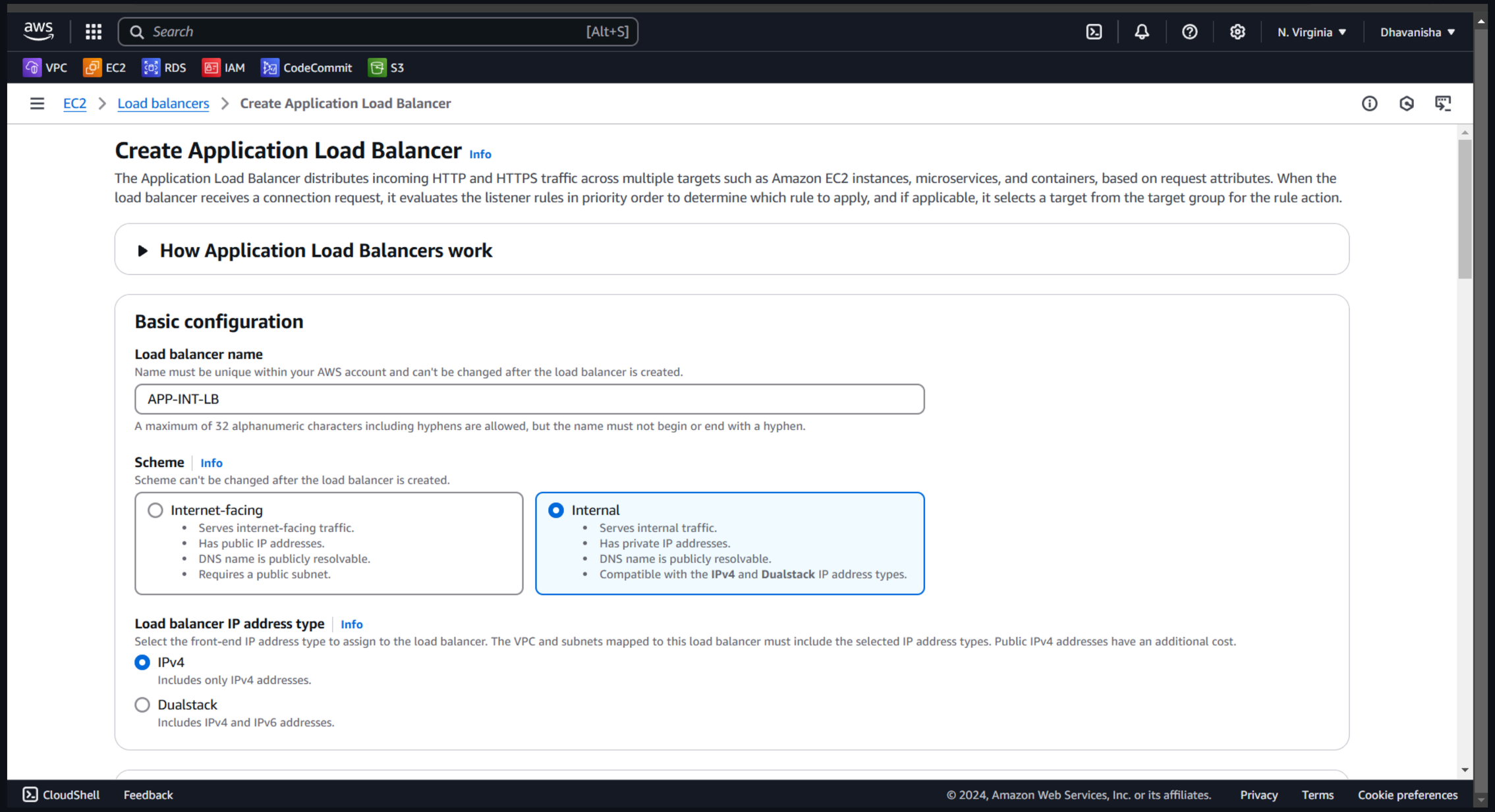
Filter targets | Show only pending | 1

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-025a4538726c67133	APP-TIER-INSTANCE	4000	Running	APP-SG	us-east-1a	10.0.143.90	subnet-0032b660

1 pending

Cancel | Previous | Create target group

Creating a Load Balancer with Target Group



Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

Availability Zones

us-east-1a (use1-az2)

Subnet

subnet-0032b660af20f6899 3-tier-architecture-subnet-APP-1-us-east-1a
IPv4 subnet CIDR: 10.0.128.0/20

IPv4 address
Assigned from CIDR 10.0.128.0/20

us-east-1b (use1-az4)

Subnet

subnet-04f2403fed283668c 3-tier-architecture-subnet-APP-2-us-east-1b
IPv4 subnet CIDR: 10.0.144.0/20

IPv4 address
Assigned from CIDR 10.0.144.0/20

Security groups Info

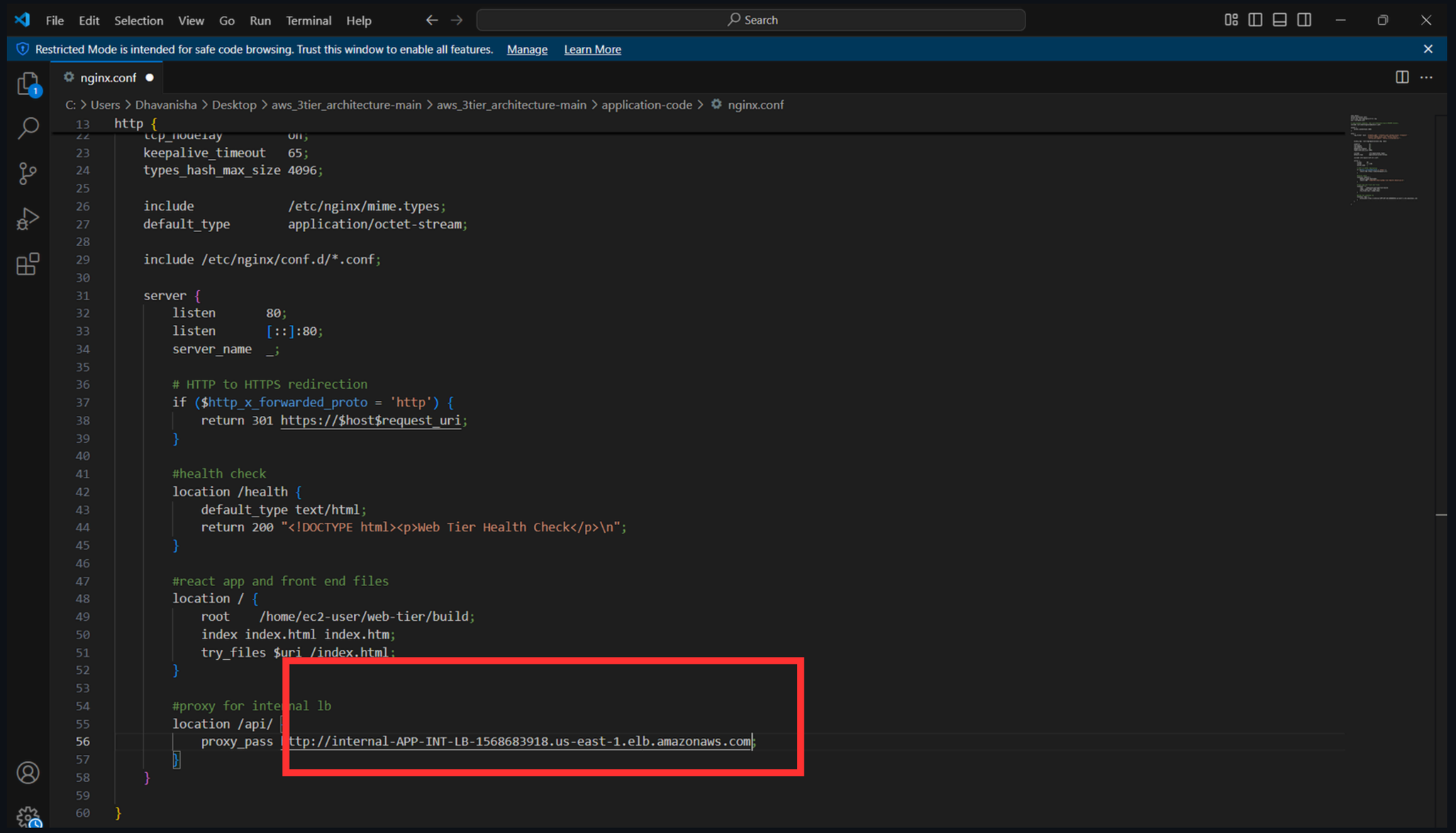
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can create a new security group.

Security groups

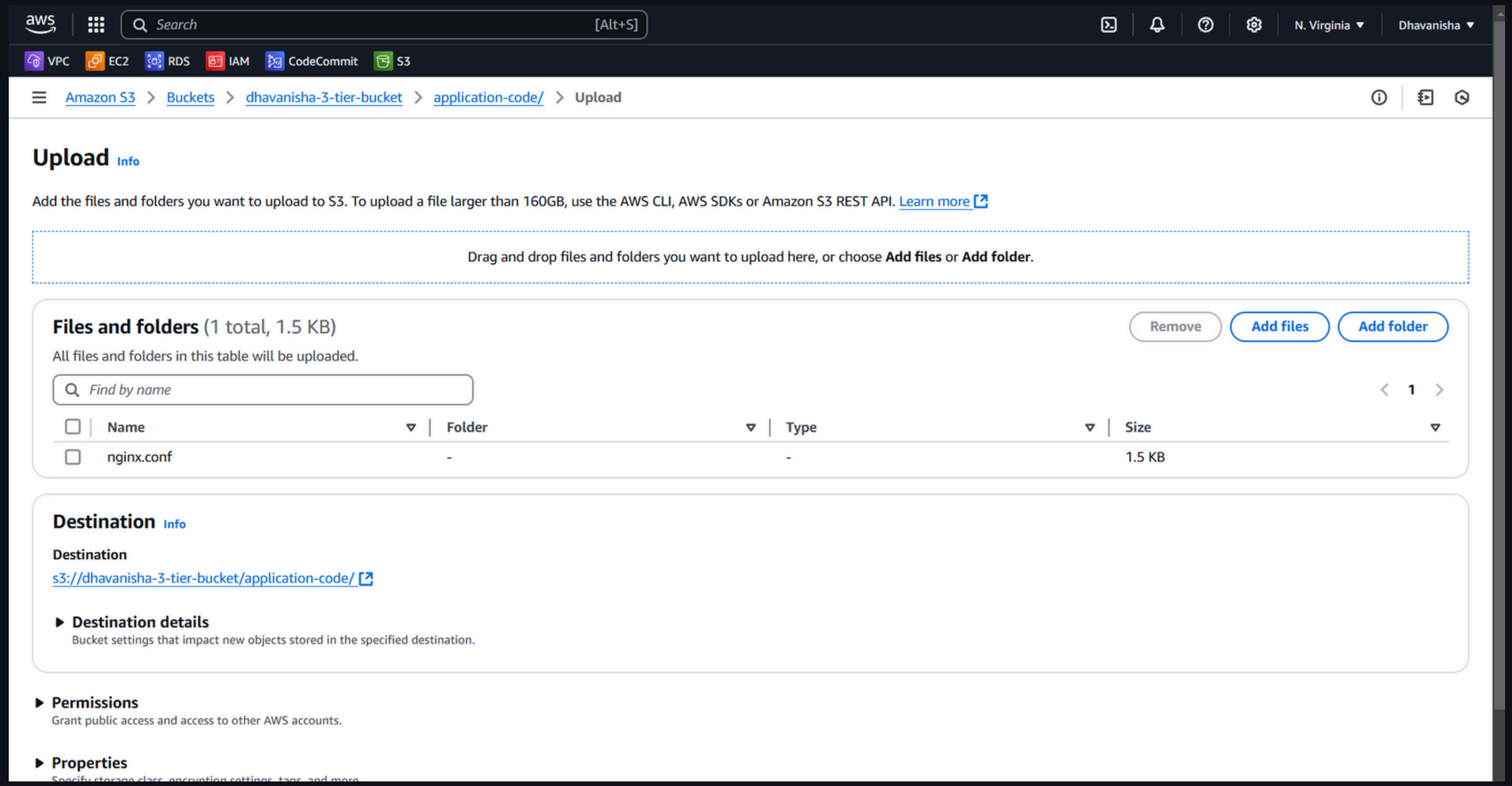
Select up to 5 security groups

INT-APP
sg-0c501b7d351aea163 VPC: vpc-028733503722f6eca

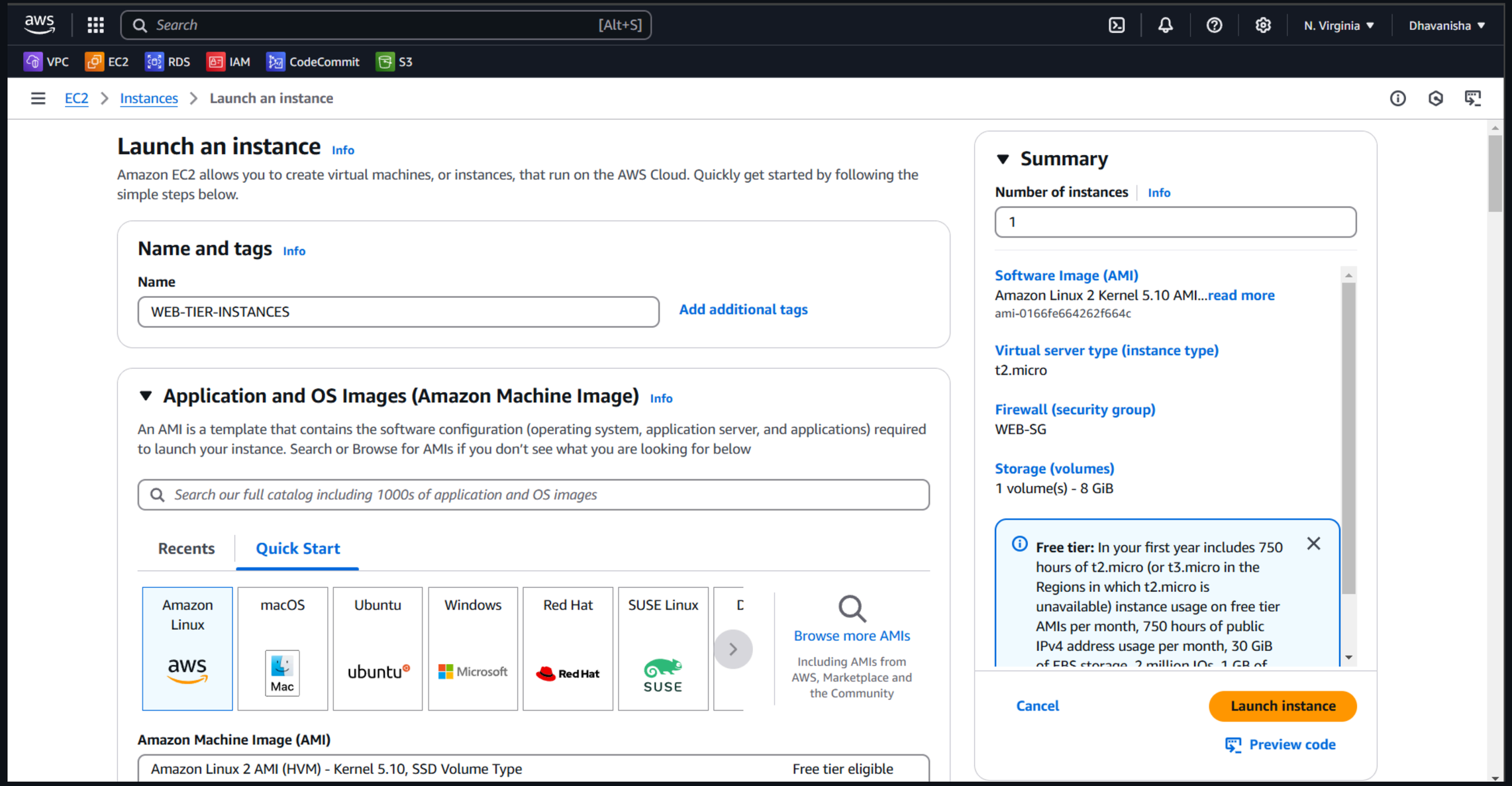
After creating an internal load balancer, copy the DNS name and modify it in the nginx.conf file where you have already uploaded the file to the S3 bucket path:/application.



modified the nginx.config file in the s3 bucket that I have already edited.



Launch an instance for the web tier.



VPC EC2 RDS IAM CodeCommit S3

EC2 > Instances > Launch an instance

Network settings Info

VPC - *required* Info

vpc-028733503722f6eca (3-tier-architecture-vpc)
10.0.0.0/16

Subnet Info

subnet-065d0c3514fa5791e
3-tier-architecture-subnet-public1-us-east-1a
VPC: vpc-028733503722f6eca Owner: 515966521120
Availability Zone: us-east-1a Zone type: Availability Zone
IP addresses available: 4090 CIDR: 10.0.0.0/20)

Create new subnet

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

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

- Create security group
- Select existing security group

Common security groups Info

Select security groups

WEB-SG sg-0562bd41b1eb6026d
VPC: vpc-028733503722f6eca

Compare security group rules

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

▶ **Advanced network configuration**

Summary

Number of instances Info

1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-0166fe664262f664c

Virtual server type (instance type)
t2.micro

Firewall (security group)
WEB-SG

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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of

Cancel

Launch instance

Preview code

Advanced details Info

Domain join directory Info

Select

Create new directory

IAM instance profile Info

3-tier-role
arn:aws:iam::515966521120:instance-profile/3-tier-role

Create new IAM profile

Hostname type Info

IP name

DNS Hostname Info

- [x] Enable IP name IPv4 (A record) DNS requests
- [ ] Enable resource-based IPv4 (A record) DNS requests
- [ ] Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery Info

Select

Shutdown behavior Info

Stop

Stop - Hibernate behavior Info

Select

Termination protection Info

Select

Summary

Number of instances Info

1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-0166fe664262f664c

Virtual server type (instance type)
t2.micro

Firewall (security group)
WEB-SG

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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of

Cancel

Launch instance

Preview code

Connect to instance Info

Connect to your instance i-025a4538726c67133 (APP-TIER-INSTANCE) using any of these options

EC2 Instance Connect | **Session Manager** | SSH client | EC2 serial console

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager Preferences page.

Cancel **Connect**

Session ID: root-isve7szfjv7cqsju3bbvi5g4n4 Instance ID: i-025a4538726c67133 Terminate

```
sh-4.2$ sudo -su ec2-user
[ec2-user@ip-10-0-143-90 bin]$ cd /home/ec2-user/
[ec2-user@ip-10-0-143-90 ~]$ pwd
/home/ec2-user
[ec2-user@ip-10-0-143-90 ~]$ curl -o- https://raw.githubusercontent.com/DHAVANISHAJ/aws-three-tier-web-architecture/main/install.sh | bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 14926  100 14926    0     0   106k      0 --:--:-- --:--:-- --:--:--  107k
=> Downloading nvm as script to '/home/ec2-user/.nvm'

=> Appending nvm source string to /home/ec2-user/.bashrc
=> Appending bash_completion source string to /home/ec2-user/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"  # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion"  # This loads nvm bash_completion
[ec2-user@ip-10-0-143-90 ~]$ source ~/.bashrc
[ec2-user@ip-10-0-143-90 ~]$ nvm install 16
Downloading and installing node v16.20.2...
Downloading https://nodejs.org/dist/v16.20.2/node-v16.20.2-linux-x64.tar.xz...
######################################################################################################################################################################## 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v16.20.2 (npm v8.19.4)
Creating default alias: default -> 16 (-> v16.20.2)
[ec2-user@ip-10-0-143-90 ~]$ nvm use 16
Now using node v16.20.2 (npm v8.19.4)
[ec2-user@ip-10-0-143-90 ~]$ cd ~/
[ec2-user@ip-10-0-143-90 ~]$ aws s3 cp s3://dhavanisha-3-tier-bucket/application-code/web-tier/ web-tier --recursive
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/public/index.html to web-tier/public/index.html
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/README.md to web-tier/README.md
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/components/Menu/Menu.styled.js to web-tier/src/components/Menu/Menu.styled.js
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/components/Menu/Menu.js to web-tier/src/components/Menu/Menu.js
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/components/Burger/index.js to web-tier/src/components/Burger/index.js
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/package.json to web-tier/package.json
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/components/Burger/Burger.js to web-tier/src/components/Burger/Burger.js
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/components/Menu/index.js to web-tier/src/components/Menu/index.js
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/components/index.js to web-tier/src/components/index.js
download: s3://dhavanisha-3-tier-bucket/application-code/web-tier/src/App.js to web-tier/src/App.js
```

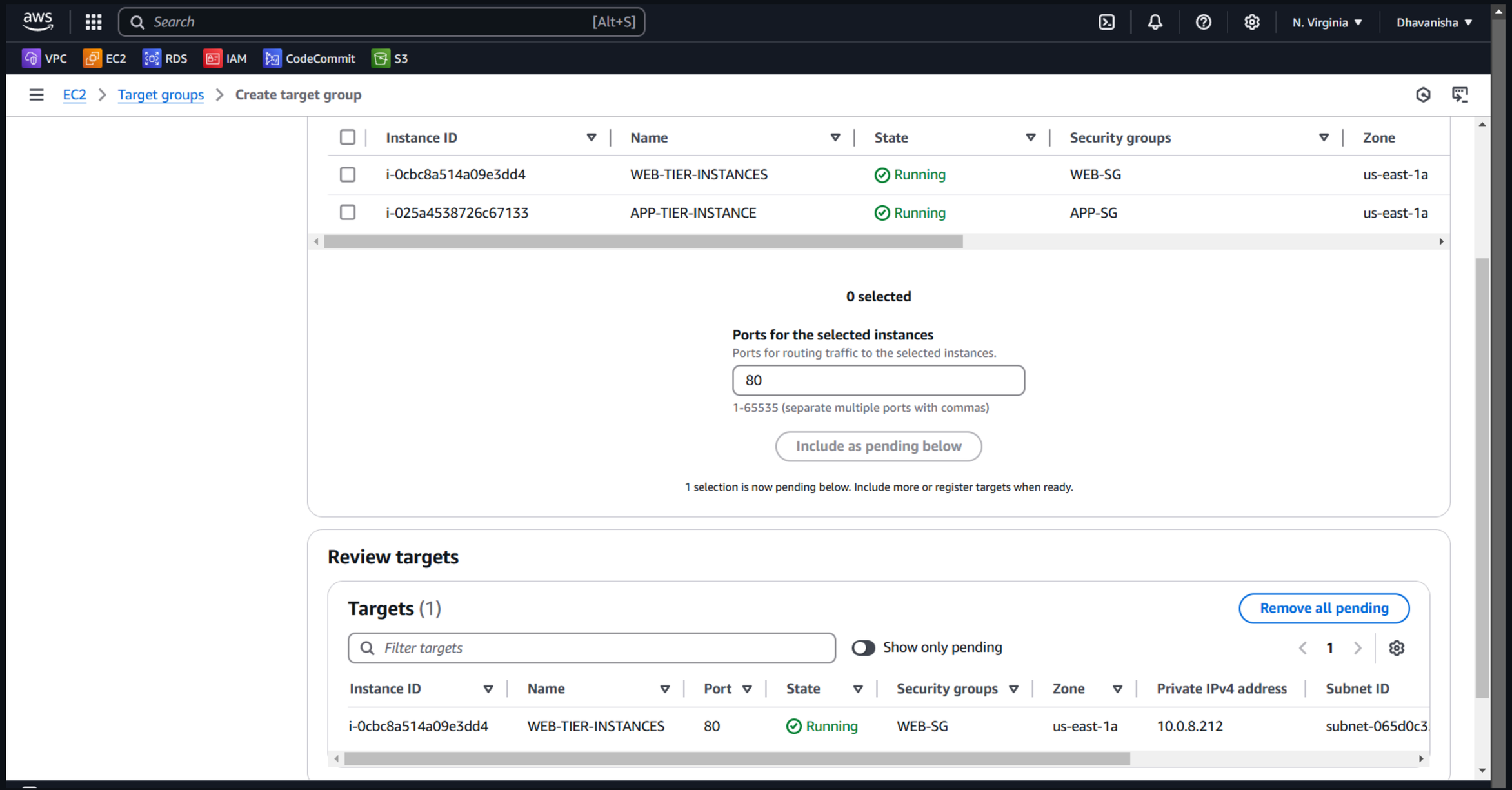
Session ID: root-isve7szfjv7cqsju3bbvi5g4n4 Instance ID: i-025a4538726c67133 Terminate

```
[ec2-user@ip-10-0-143-90 ~]$ ls
web-tier
[ec2-user@ip-10-0-143-90 ~]$ cd ~/web-tier
[ec2-user@ip-10-0-143-90 web-tier]$ npm install
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: '@testing-library/dom@10.4.0',
npm WARN EBADENGINE   required: { node: '>=18' },
npm WARN EBADENGINE   current: { node: 'v16.20.2', npm: '8.19.4' }
npm WARN EBADENGINE }
npm WARN deprecated sourcemap-codec@1.4.8: Please use @jridgewell/sourcemap-codec instead
npm WARN deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plugin-terser
npm WARN deprecated workbox-cacheable-response@6.6.0: workbox-background-sync@6.6.0
npm WARN deprecated workbox-google-analytics@6.6.0: It is not compatible with newer versions of GA starting with v4, as long as you are using GAv3 it should be ok, but the
 package is not longer being maintained
npm WARN deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the compatibility table on MDN: https://de
veloper.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort#browser_compatibility
npm WARN deprecated q@1.5.1: You or someone you depend on is using Q, the JavaScript Promise library that gave JavaScript developers strong feelings about promises. They c
an almost certainly migrate to the native JavaScript promise now. Thank you literally everyone for joining me in this bet against the odds. Be excellent to each other.
npm WARN deprecated
npm WARN deprecated (For a CapTP with native promises, see @endo/eventual-send and @endo/captp)
npm WARN deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm WARN deprecated domexception@2.0.1: Use your platform's native DOMException instead
npm WARN deprecated abab@2.0.6: Use your platform's native atob() and btoa() methods instead
npm WARN deprecated inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out lru-cache if you want a good and tested way to coalesce async
requests by a key value, which is much more comprehensive and powerful.
npm WARN deprecated glob@7.2.3: Glob versions prior to v9 are no longer supported
npm WARN deprecated rimraf@3.0.2: Rimraf versions prior to v4 are no longer supported
npm WARN deprecated @humanwhocodes/object-schema@2.0.3: Use @eslint/object-schema instead
npm WARN deprecated @humanwhocodes/config-array@0.13.0: Use @eslint/config-array instead
npm WARN deprecated @babel/plugin-proposal-private-methods@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintained. P
lease use @babel/plugin-transform-private-methods instead.
npm WARN deprecated @babel/plugin-proposal-optional-chaining@7.21.0: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintained.
 Please use @babel/plugin-transform-optional-chaining instead.
npm WARN deprecated @babel/plugin-proposal-numeric-separator@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintained.
 Please use @babel/plugin-transform-numeric-separator instead.
npm WARN deprecated @babel/plugin-proposal-class-properties@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintained.
Please use @babel/plugin-transform-class-properties instead.
npm WARN deprecated @babel/plugin-proposal-nullish-coalescing-operator@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer m
aintained. Please use @babel/plugin-transform-nullish-coalescing-operator instead.
```

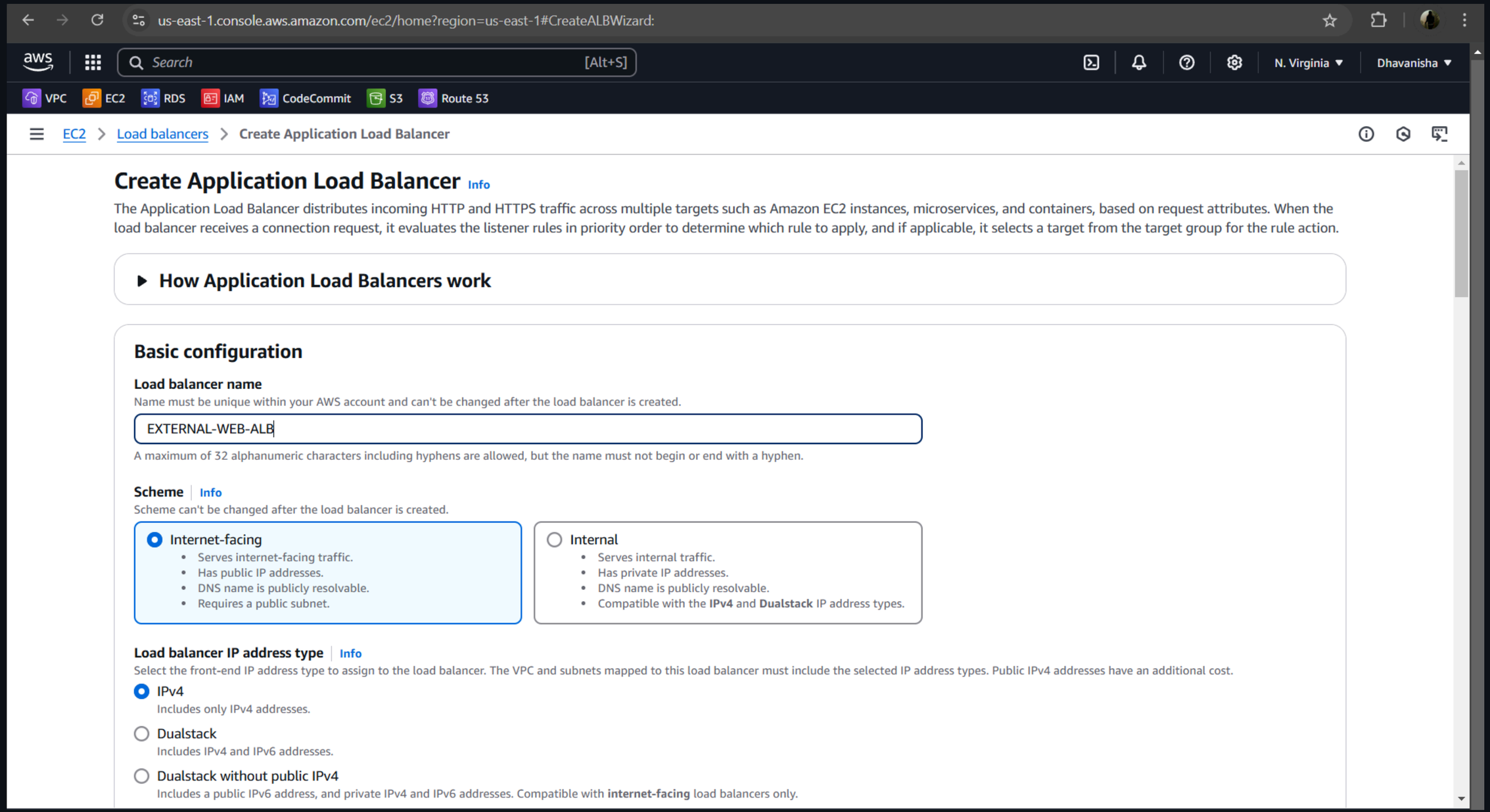
Session ID: root-949ckfr7pebhdefvu8ajebn3zi Instance ID: i-0cbc8a514a09e3dd4 Terminate

```
npm notice New major version of npm available! 8.19.4 -> 11.0.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.0.0
npm notice Run npm install -g npm@11.0.0 to update!
npm notice
[ec2-user@ip-10-0-8-212 web-tier]$ npm run build

> aws-3tier-web-layer@0.1.0 build
> react-scripts build

Creating an optimized production build...
One of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it may break at any time.

babel-preset-react-app is part of the create-react-app project, which
is not maintianed anymore. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.

Compiled successfully.

File sizes after gzip:

  75.3 kB  build/static/js/main.eed54180.js
  1.79 kB  build/static/js/453.a4ec9c9e.chunk.js
  493 B    build/static/css/main.b20b6ac4.css

The project was built assuming it is hosted at ./.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.

Find out more about deployment here:

  https://cra.link/deployment

[ec2-user@ip-10-0-8-212 web-tier]$
```

I setup a target group for internal load balancer.



Creating a Load Balancer with Target Group



Availability Zones

us-east-1a (use1-az2)

Subnet

subnet-065d0c3514fa5791e 3-tier-architecture-subnet-public1-us-east-1a
IPv4 subnet CIDR: 10.0.0.0/20

IPv4 address
Assigned by AWS

us-east-1b (use1-az4)

Subnet

subnet-0eaaa45b2d319e238 3-tier-architecture-subnet-public2-us-east-1b
IPv4 subnet CIDR: 10.0.16.0/20

IPv4 address
Assigned by AWS

Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can create a new security group.

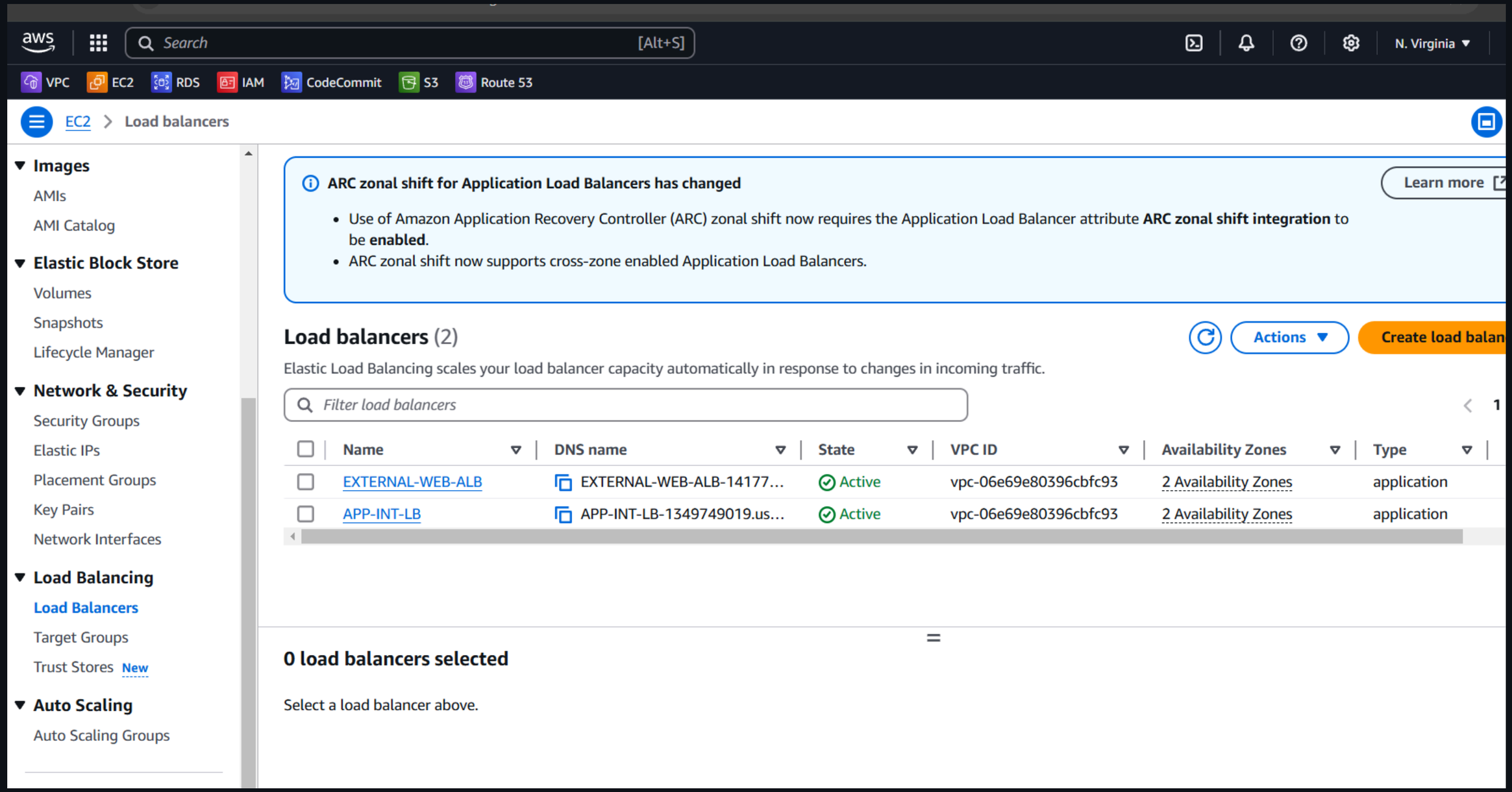
Security groups

Select up to 5 security groups

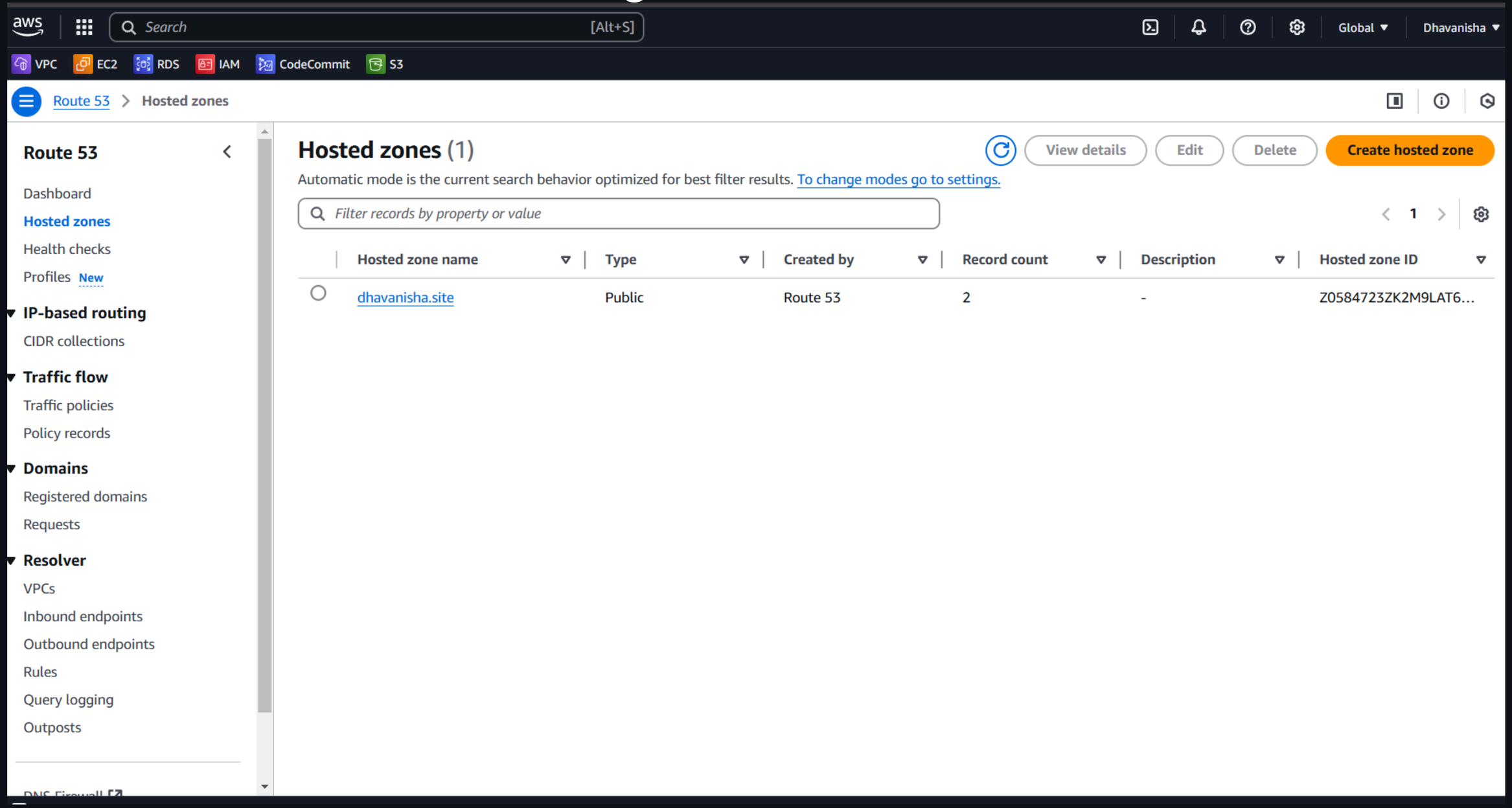
WEB-ALB-SG
sg-08e332667167a4e23 VPC: vpc-028733503722f6eca

Listeners and routing Info

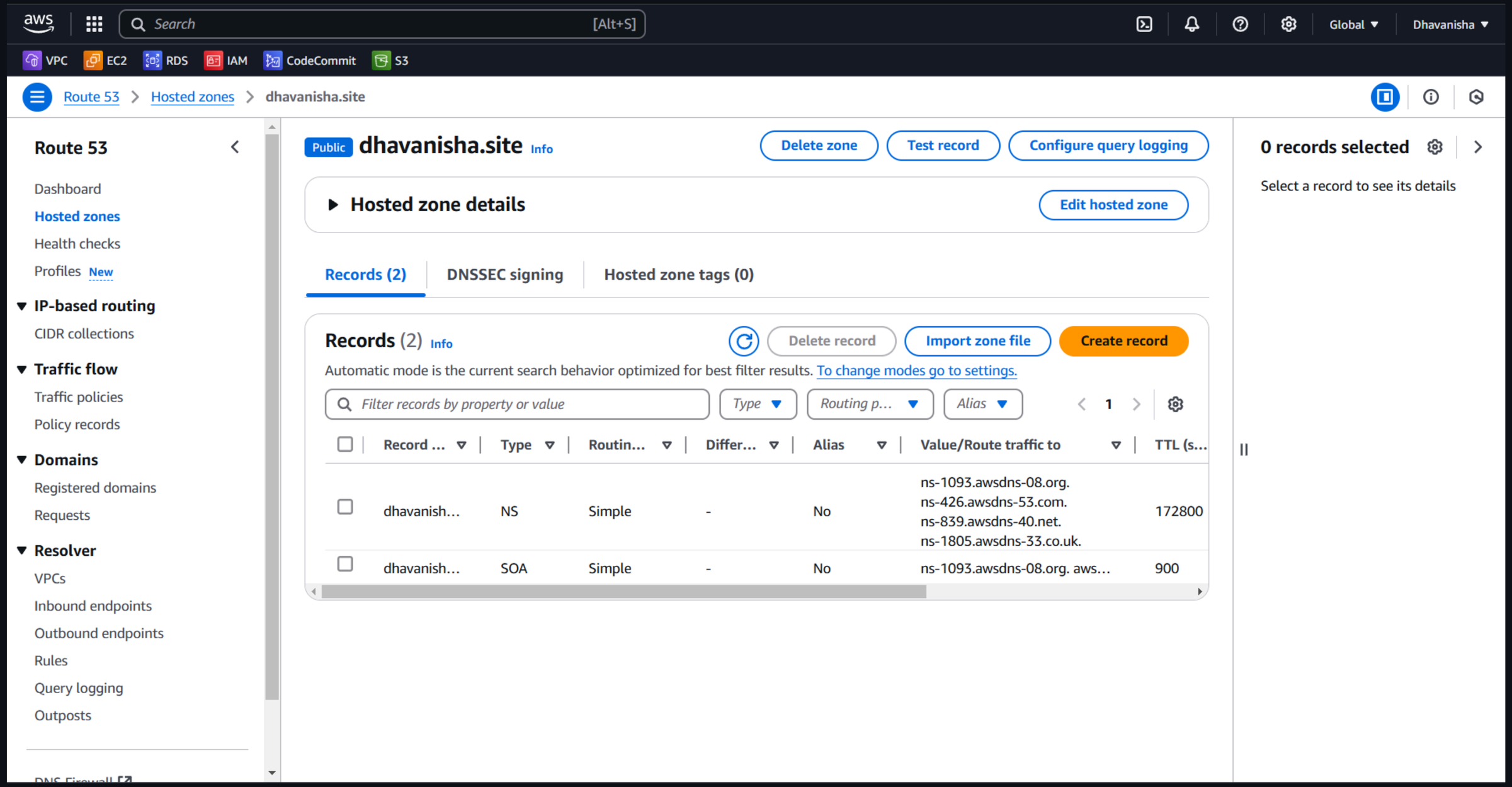
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its

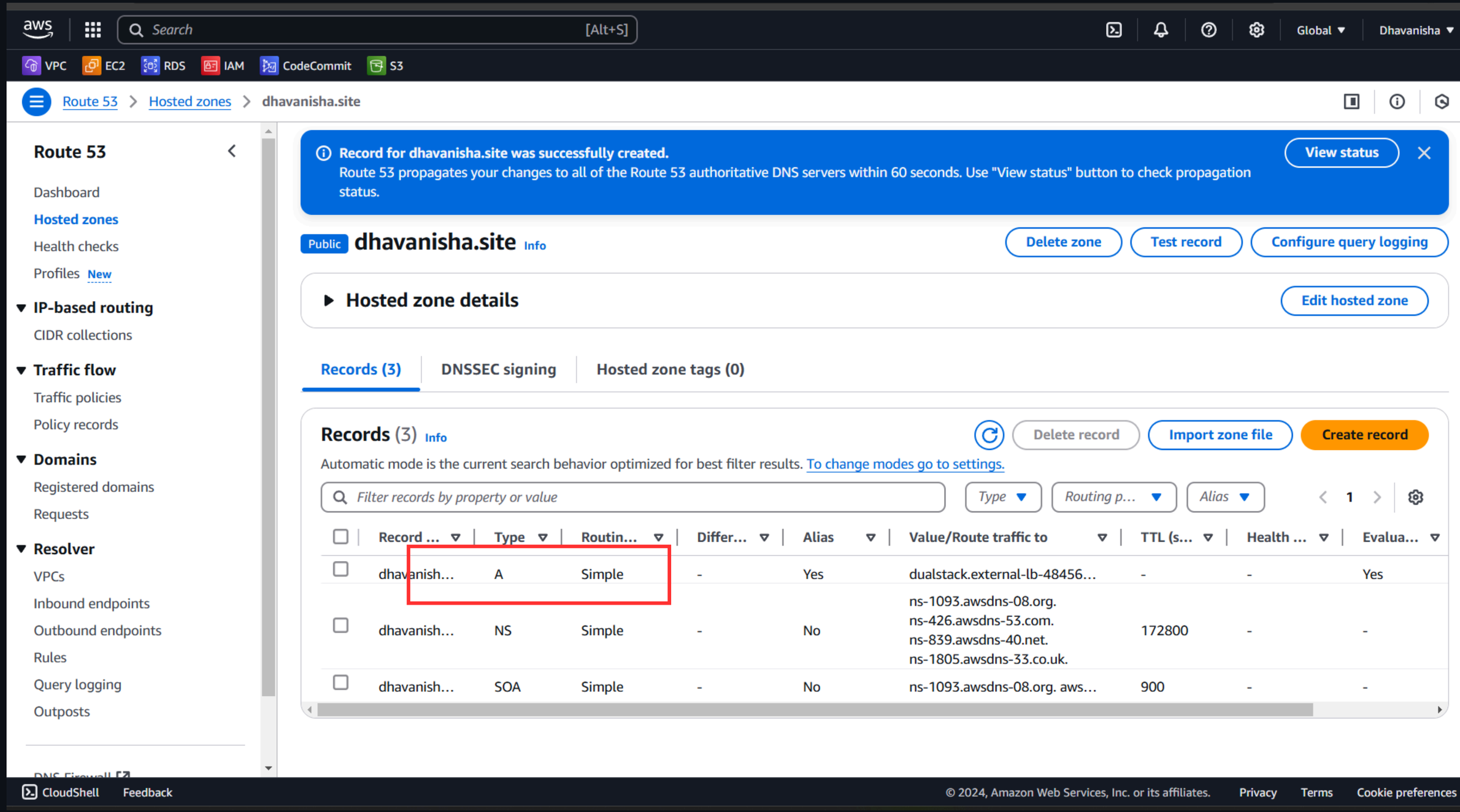

aws
Search
[Alt+S]
N. Virginia
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2 > Load balancers
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Auto Scaling
Auto Scaling Groups
ARC zonal shift for Application Load Balancers has changed
Learn more
Use of Amazon Application Recovery Controller (ARC) zonal shift now requires the Application Load Balancer attribute ARC zonal shift integration to be enabled.
ARC zonal shift now supports cross-zone enabled Application Load Balancers.
Load balancers (2)
Actions
Create load balan
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.
Filter load balancers
1
Name
DNS name
State
VPC ID
Availability Zones
Type
EXTERNAL-WEB-ALB
EXTERNAL-WEB-ALB-14177...
Active
vpc-06e69e80396cbfc93
2 Availability Zones
application
APP-INT-LB
APP-INT-LB-1349749019.us...
Active
vpc-06e69e80396cbfc93
2 Availability Zones
application
0 load balancers selected
Select a load balancer above.

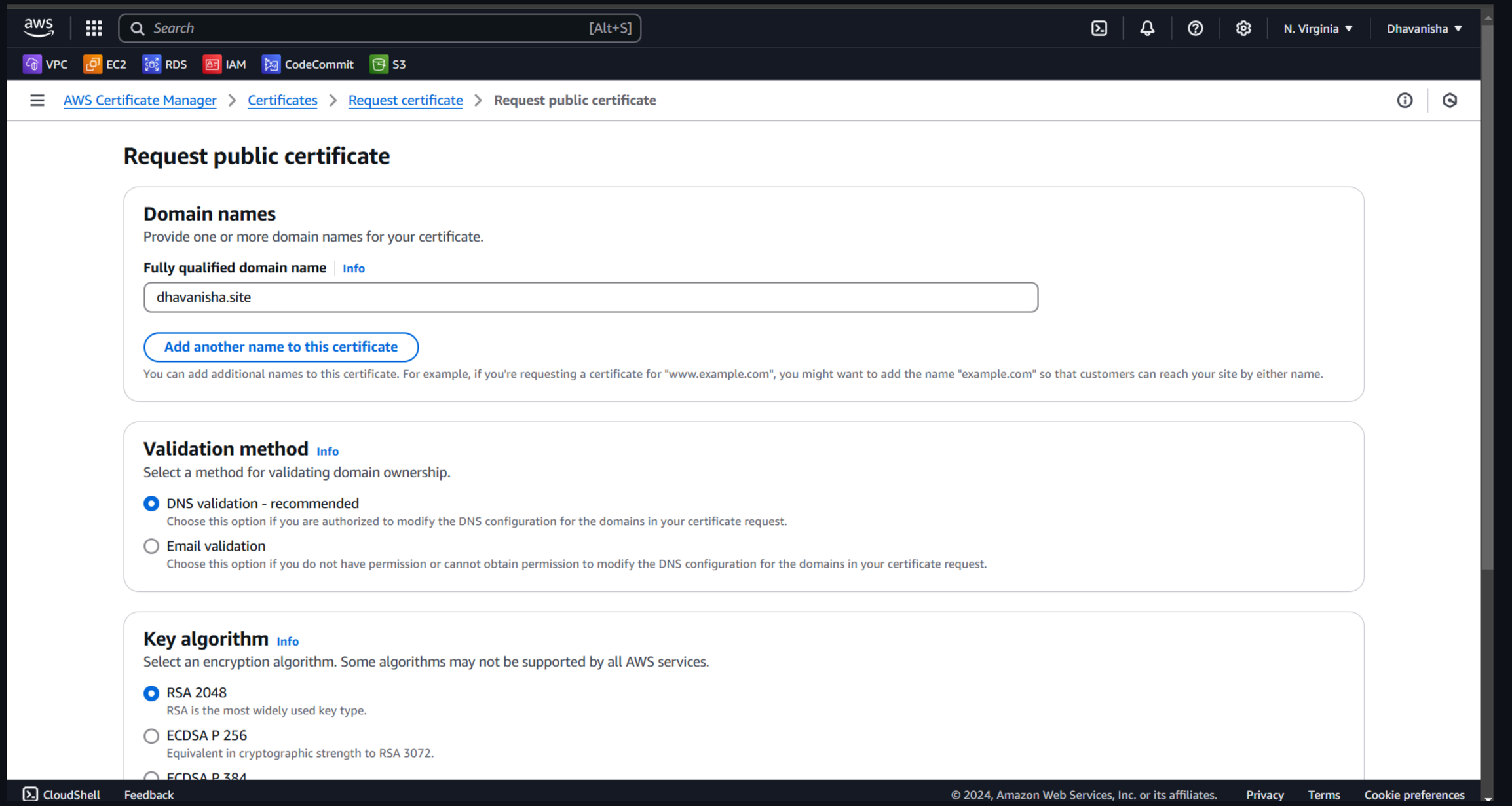
Navigate to route 53



Create A record




aws
Search
[Alt+S]
Global
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
Hosted zones
dhavanisha.site
Route 53
Dashboard
Hosted zones
Health checks
Profiles New
IP-based routing
CIDR collections
Traffic flow
Traffic policies
Policy records
Domains
Registered domains
Requests
Resolver
VPCs
Inbound endpoints
Outbound endpoints
Rules
Query logging
Outposts
Record for dhavanisha.site was successfully created.
Route 53 propagates your changes to all of the Route 53 authoritative DNS servers within 60 seconds. Use "View status" button to check propagation status.
View status
Public
dhavanisha.site
Info
Delete zone
Test record
Configure query logging
Hosted zone details
Edit hosted zone
Records (3)
DNSSEC signing
Hosted zone tags (0)
Records (3)
Info
Delete record
Import zone file
Create record
Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.
Filter records by property or value
Type
Routing p...
Alias
1
Record ...
Type
Routin...
Differ...
Alias
Value/Route traffic to
TTL (s...
Health ...
Evalua...
dhavanish...
A
Simple
-
Yes
dualstack.external-lb-48456...
-
-
Yes
dhavanish...
NS
Simple
-
No
ns-1093.awsdns-08.org.
ns-426.awsdns-53.com.
ns-839.awsdns-40.net.
ns-1805.awsdns-33.co.uk.
172800
-
-
dhavanish...
SOA
Simple
-
No
ns-1093.awsdns-08.org. aws...
900
-
-
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



Navigate to Aws certificate manager and request a public certificate

Request certificate

Certificate type Info

ACM certificates can be used to establish secure communications access across the internet or within an internal network. Choose the type of certificate for ACM to provide.

- (•) Request a public certificate
  Request a public SSL/TLS certificate from Amazon. By default, public certificates are trusted by browsers and operating systems.
- ( ) Request a private certificate
  No private CAs available for issuance.

Requesting a private certificate requires the creation of a private certificate authority (CA). To create a private CA, visit AWS Private Certificate Authority

Cancel Next

Create DNS records in Amazon Route 53 (1/1)

Search domains

1 match

Validation status = Pending validation | Validation status = Failed | Is domain in Route 53? = Yes | Clear filters

1

Domain	Validation status	Is domain in Route 53?
dhavanisha.site	Pending validation	Yes

Cancel | Create records

AWS Certificate Manager (ACM)
List certificates
Request certificate
Import certificate
AWS Private CA

Successfully requested certificate with ID fd55785c-1c43-4590-baf5-60453648f1c5
A certificate request with a status of pending validation has been created. Further action is needed to complete the validation and approval of the certificate.

View certificate

fd55785c-1c43-4590-baf5-60453648f1c5

Delete

Certificate status

Identifier
fd55785c-1c43-4590-baf5-60453648f1c5

Status
Pending validation Info

ARN
arn:aws:acm:us-east-1:515966521120:certificate/fd55785c-1c43-4590-baf5-60453648f1c5

Type
Amazon Issued

Domains (1)

Create records in Route 53 | Export to CSV

1

Domain	Status	Renewal status	Type	CNAME name
dhavanisha.site	Pending validation	-	CNAME	_129276dfb1c5c486e390faaff6e5d1db.dhavanis

Details

AWS Certificate Manager (ACM)
List certificates
Request certificate
Import certificate
AWS Private CA

92afab21-dda5-445a-b963-d06532455fbe

Delete

Certificate status

Identifier
92afab21-dda5-445a-b963-d06532455fbe

Status
Issued

ARN
arn:aws:acm:us-east-1:515966521120:certificate/92afab21-dda5-445a-b963-d06532455fbe

Type
Amazon Issued

Domains (1)

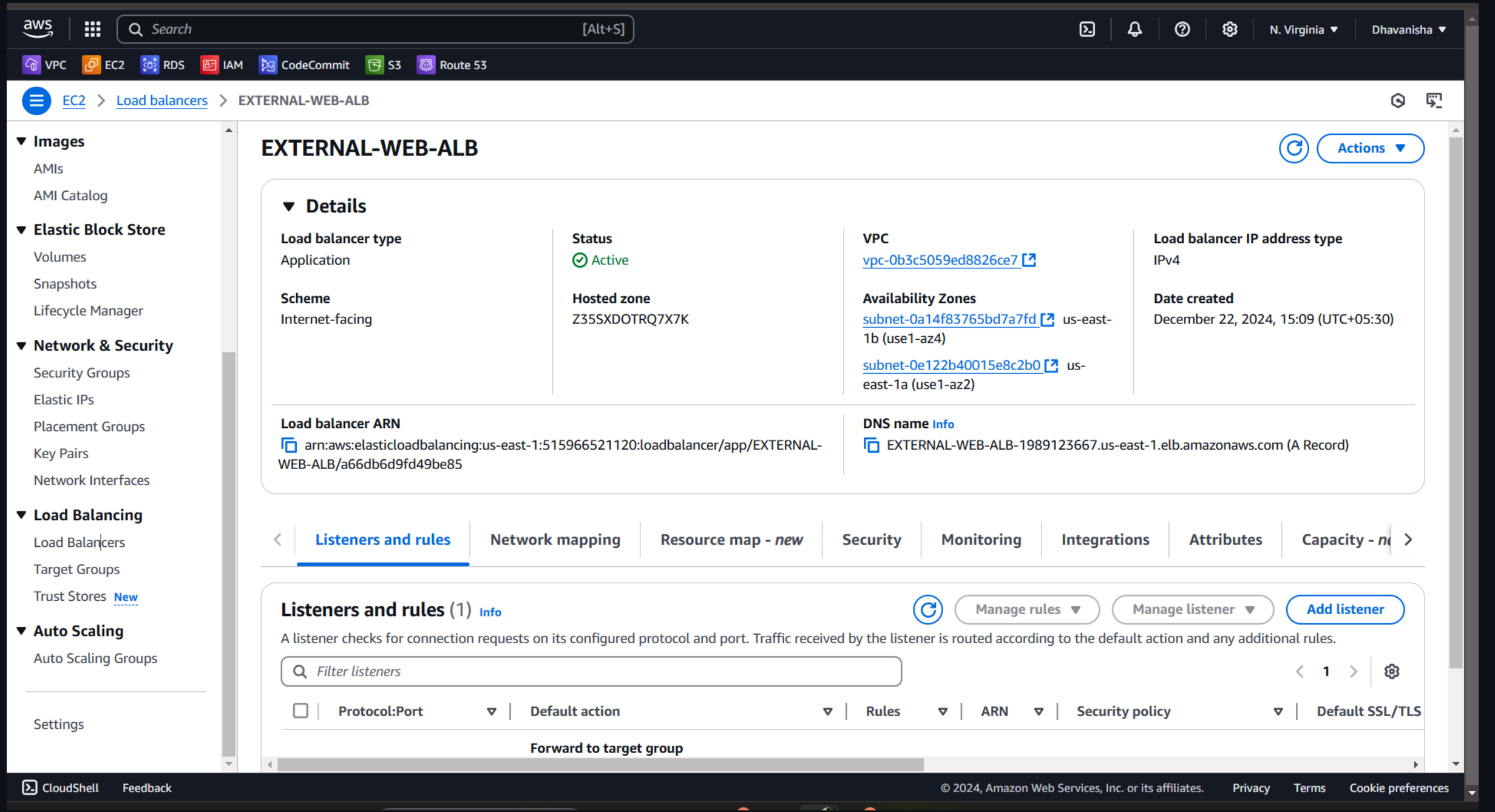
Create records in Route 53 | Export to CSV

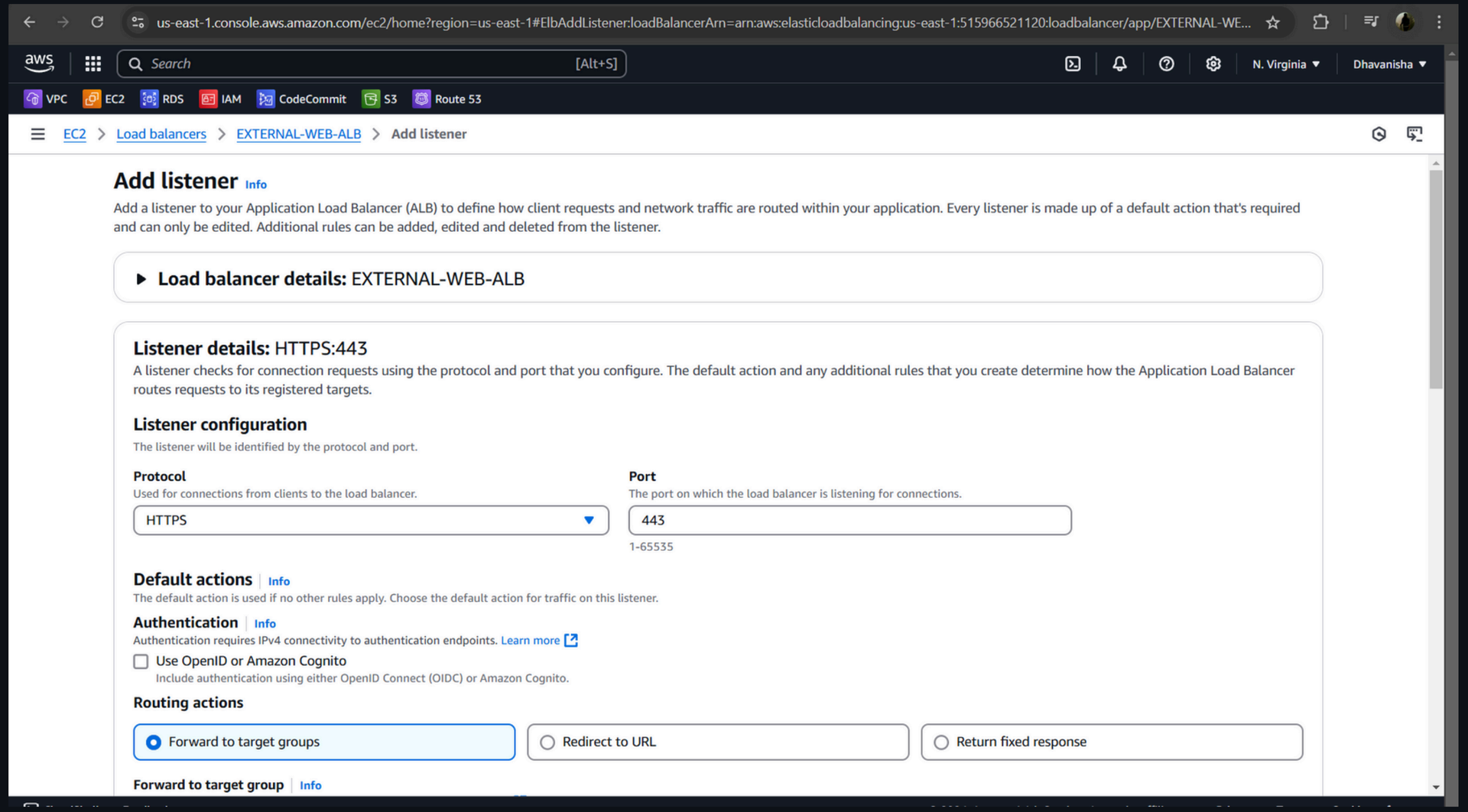
1

Domain	Status	Renewal status	Type	CNAME name
dhavanisha.site	Success	-	CNAME	_129276dfb1c5c486e390faaff6e5d1db.dhavanis

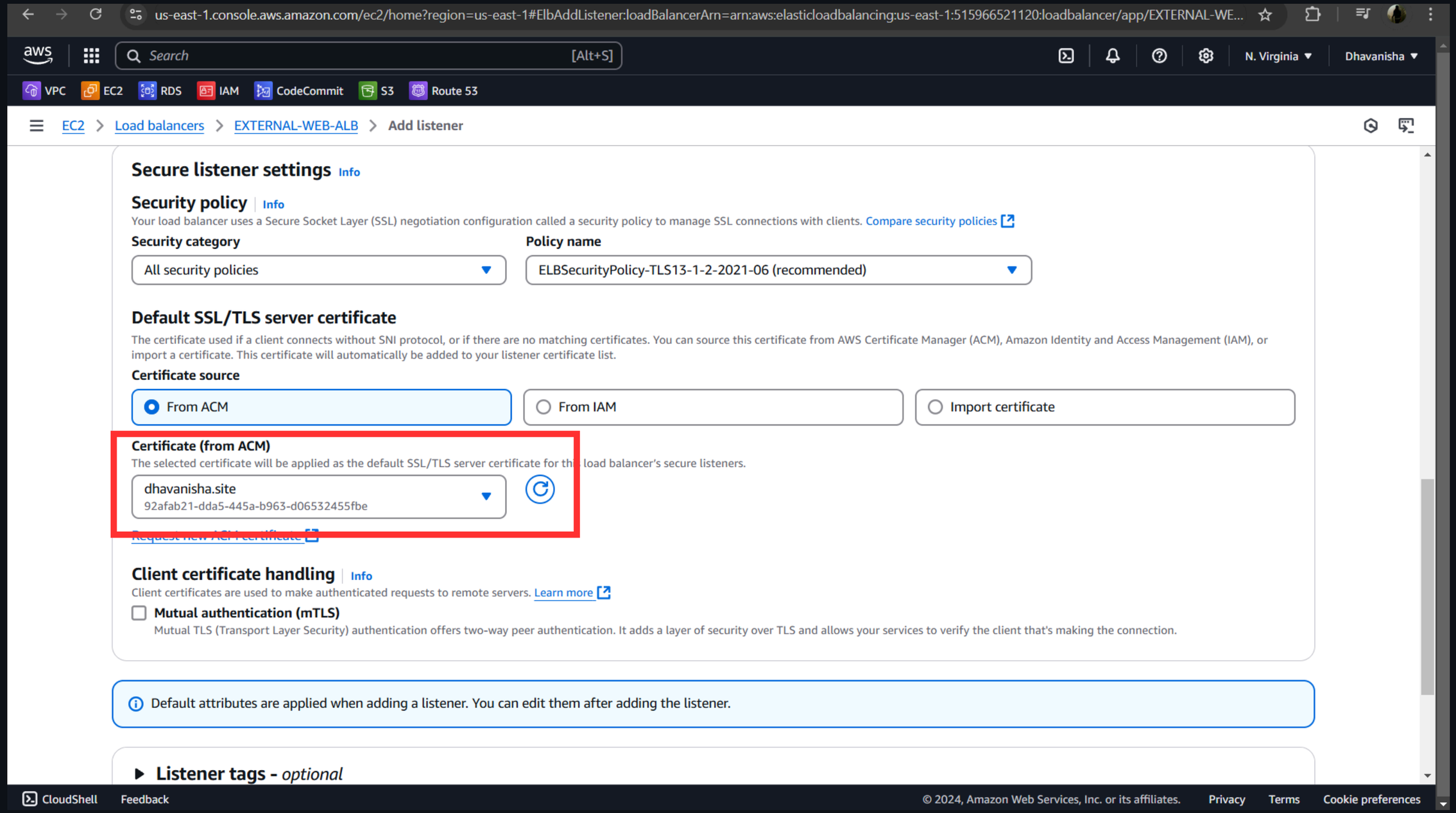
Details

In use	Serial number	Requested at	Renewal eligibility
No	0e:f8:3e:64:6a:51:5b:8f:44:ed:ef:5f:10:4e:6d:68	December 22, 2024, 15:34:27 (UTC+05:30)	Ineligible


aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2 > Load balancers > EXTERNAL-WEB-ALB
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Auto Scaling
Auto Scaling Groups
Settings
EXTERNAL-WEB-ALB
Actions
Details
Load balancer type
Application
Status
Active
VPC
vpc-0b3c5059ed8826ce7
Load balancer IP address type
IPv4
Scheme
Internet-facing
Hosted zone
Z35SXDOTRQ7X7K
Availability Zones
subnet-0a14f83765bd7a7fd us-east-1b (use1-az4)
subnet-0e122b40015e8c2b0 us-east-1a (use1-az2)
Date created
December 22, 2024, 15:09 (UTC+05:30)
Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:515966521120:loadbalancer/app/EXTERNAL-WEB-ALB/a66db6d9fd49be85
DNS name Info
EXTERNAL-WEB-ALB-1989123667.us-east-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map - new
Security
Monitoring
Integrations
Attributes
Capacity - ne
Listeners and rules (1) Info
Manage rules
Manage listener
Add listener
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.
Filter listeners
1
Protocol:Port
Default action
Rules
ARN
Security policy
Default SSL/TLS
Forward to target group
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ElbAddListener:loadBalancerArn=arn:aws:elasticloadbalancing:us-east-1:515966521120:loadbalancer/app/EXTERNAL-WE...
aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2 > Load balancers > EXTERNAL-WEB-ALB > Add listener
Add listener Info
Add a listener to your Application Load Balancer (ALB) to define how client requests and network traffic are routed within your application. Every listener is made up of a default action that's required and can only be edited. Additional rules can be added, edited and deleted from the listener.
Load balancer details: EXTERNAL-WEB-ALB
Listener details: HTTPS:443
A listener checks for connection requests using the protocol and port that you configure. The default action and any additional rules that you create determine how the Application Load Balancer routes requests to its registered targets.
Listener configuration
The listener will be identified by the protocol and port.
Protocol
Used for connections from clients to the load balancer.
HTTPS
Port
The port on which the load balancer is listening for connections.
443
1-65535
Default actions Info
The default action is used if no other rules apply. Choose the default action for traffic on this listener.
Authentication Info
Authentication requires IPv4 connectivity to authentication endpoints. Learn more
Use OpenID or Amazon Cognito
Include authentication using either OpenID Connect (OIDC) or Amazon Cognito.
Routing actions
Forward to target groups
Redirect to URL
Return fixed response
Forward to target group Info

certificate is linked to the external load balancer.

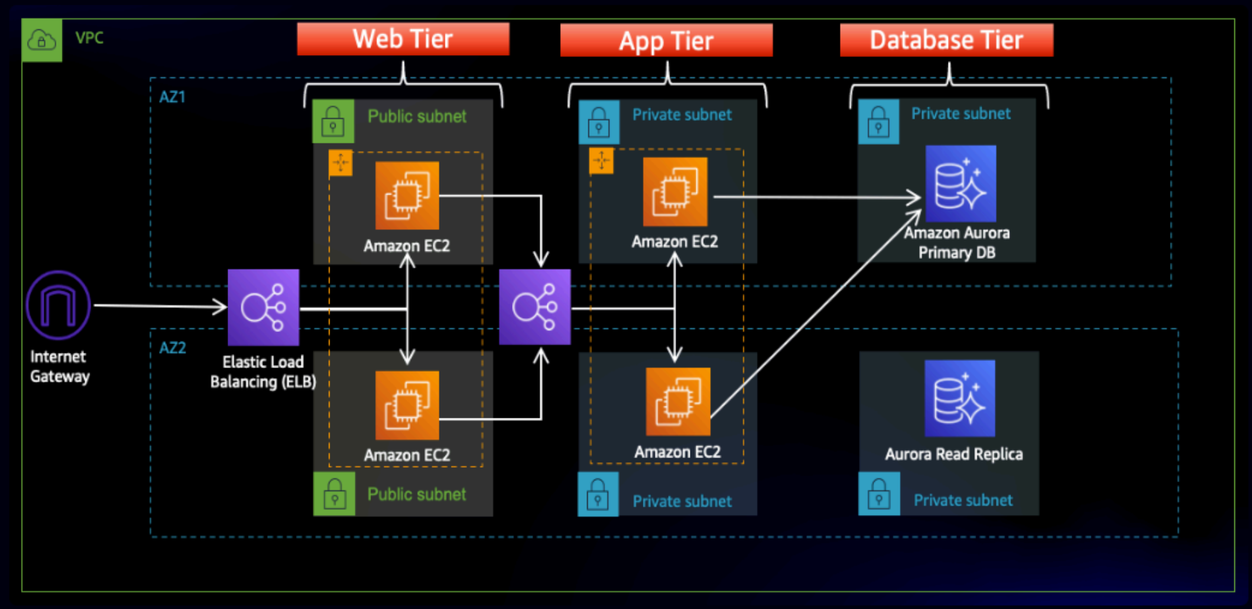


The project successfully implemented a robust and scalable 3-Tier Architecture using AWS services.

dhavanisha.site/#/

Incognito

'AWS 3-TIER WEB APP DEMO WITH DHAVANISHA'



HOME

DB DEMO

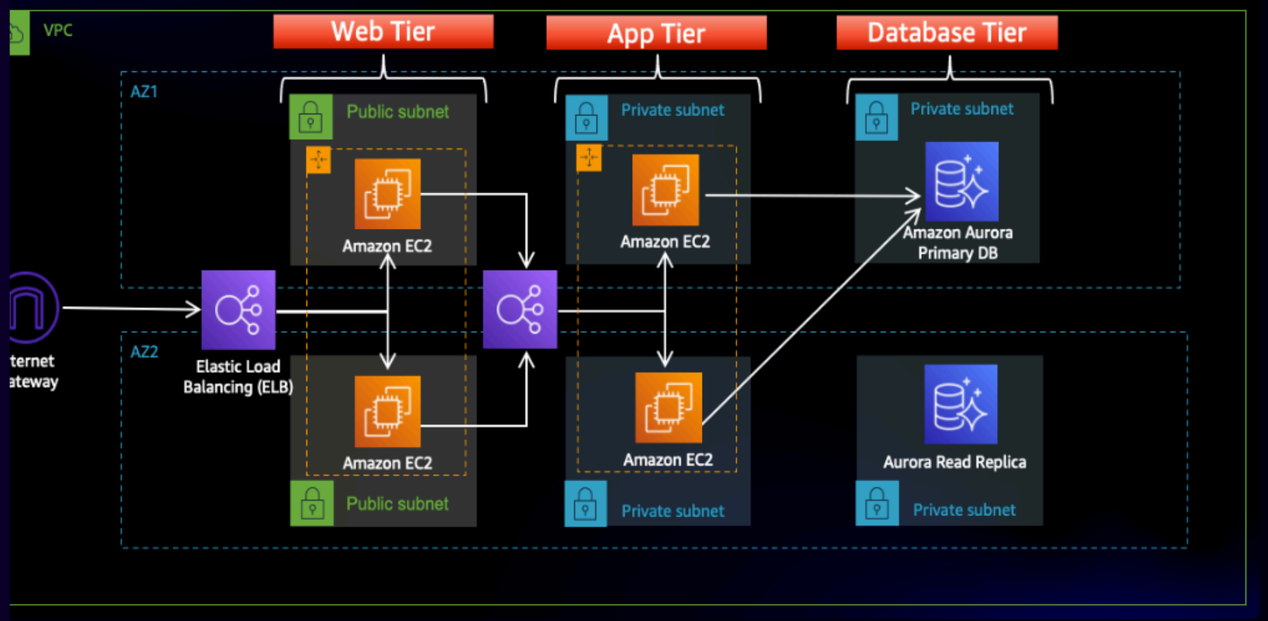
AURORA DATABASE DEMO PAGE

DEL

ID	AMOUNT	DESC
ADD	HI	WELCOME
8	1	DHAVANISHA
9	2	THREE
10	3	TIER
11	4	ARCHITECTURE

'AWS 3-TIER WEB APP DEMO WITH DHAVANISHA'

HOME
DB DEMO



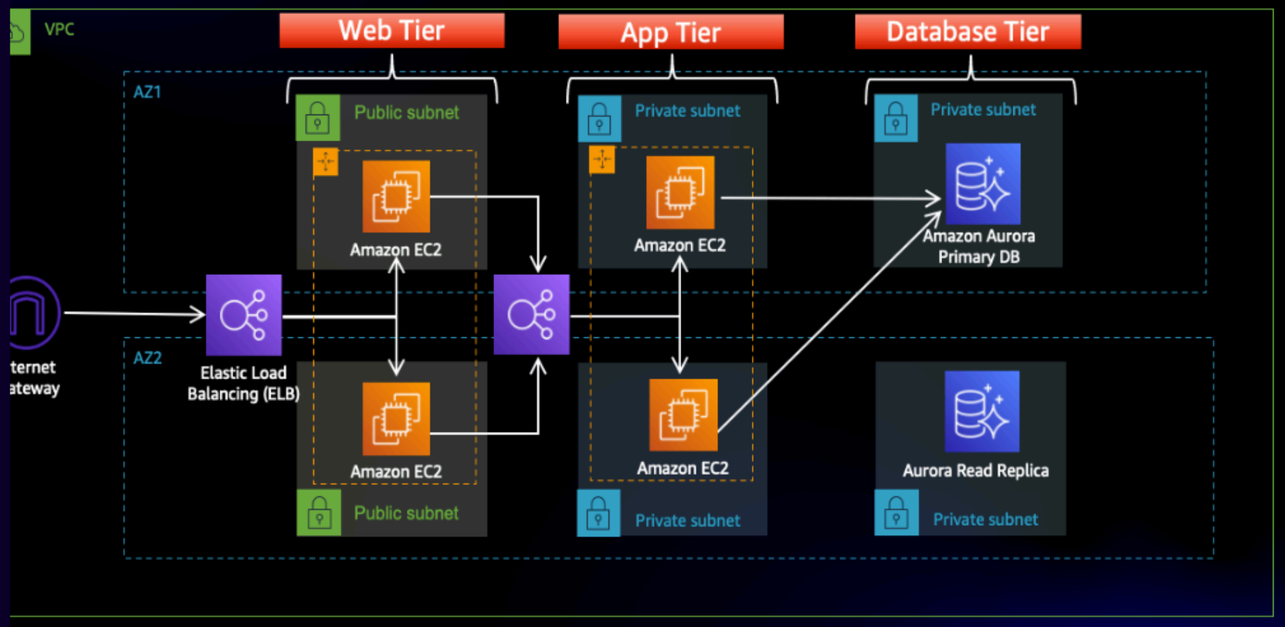
https://dhavanisha.site/#/

Incognito

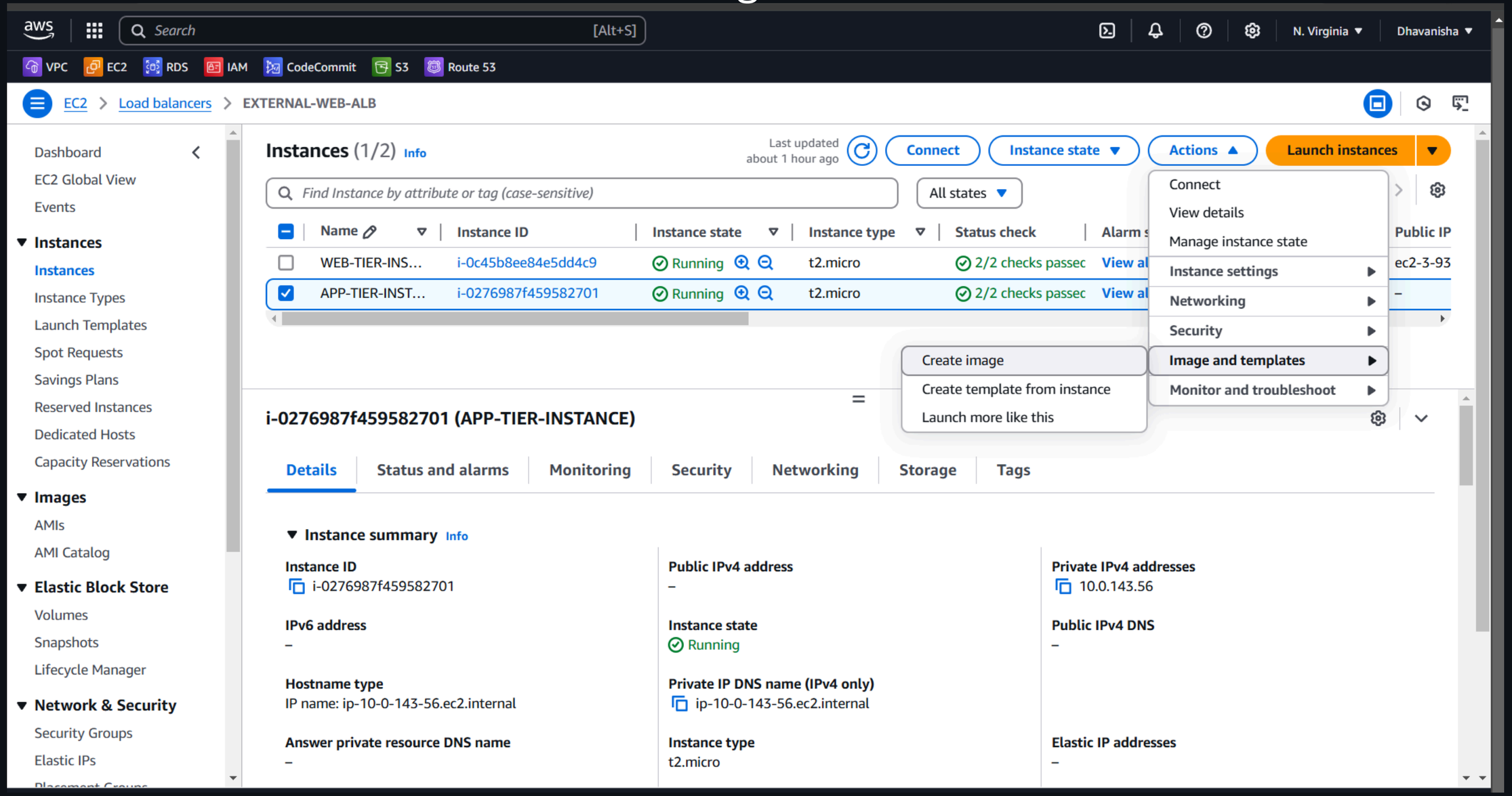
'AWS 3-TIER WEB APP DEMO WITH DHAVANISHA'

HOME

DB DEMO



Created a Autoscaling for Automatically adjusts resources to handle traffic surges.



Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID
i-0276987f459582701 (APP-TIER-INSTANCE)

Image name
APP-TIER-GOLDEN-AMI
Maximum 127 characters. Can't be modified after creation.

Image description - *optional*
APP-TIER-GOLDEN-AMI
Maximum 255 characters

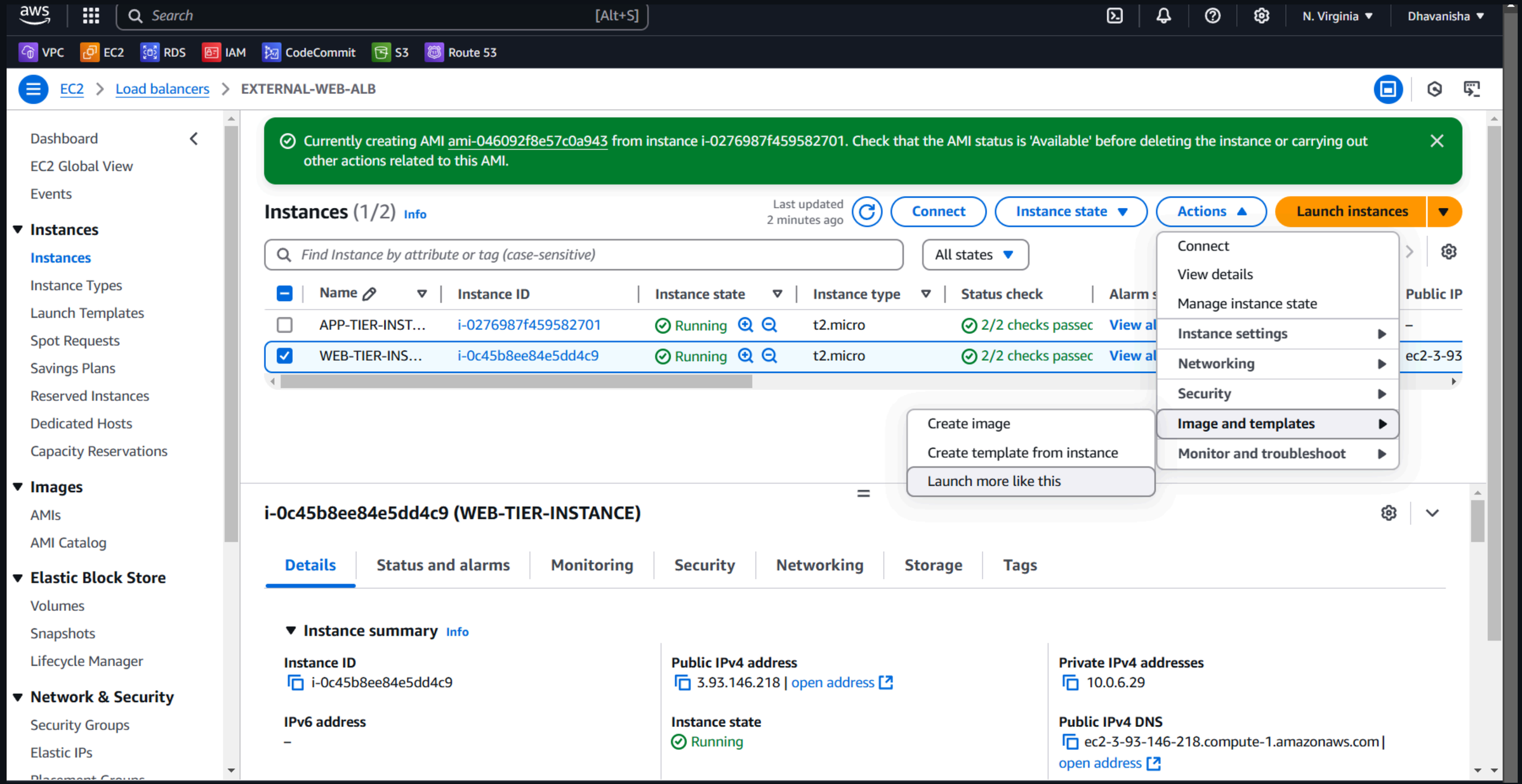
☑ Reboot instance
When selected, Amazon EC2 reboots the instance so that data is at rest when snapshots of the attached volumes are taken. This ensures data consistency.

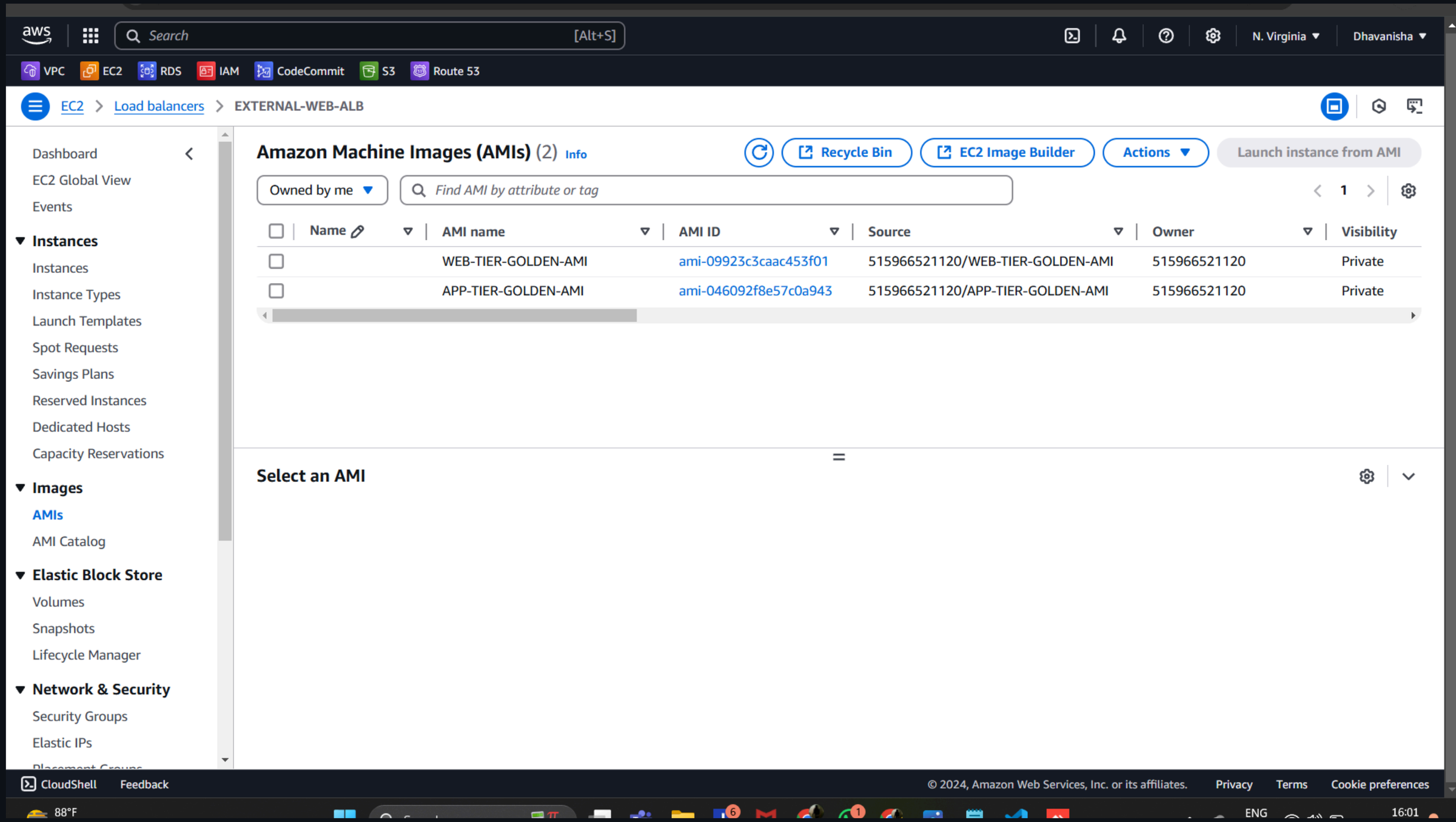
Instance volumes

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev...	Create new snapshot fr...	8	EBS General Purpose SS...	100		☑ Enable	☐ Enable

Add volume

ⓘ During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.


aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2 > Load balancers > EXTERNAL-WEB-ALB
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Currently creating AMI ami-046092f8e57c0a943 from instance i-0276987f459582701. Check that the AMI status is 'Available' before deleting the instance or carrying out other actions related to this AMI.
Instances (1/2) Info
Last updated 2 minutes ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm s
Public IP
APP-TIER-INST...
i-0276987f459582701
Running
t2.micro
2/2 checks passec
View al
WEB-TIER-INS...
i-0c45b8ee84e5dd4c9
Running
t2.micro
2/2 checks passec
View al
ec2-3-93
Connect
View details
Manage instance state
Instance settings
Networking
Security
Image and templates
Monitor and troubleshoot
Create image
Create template from instance
Launch more like this
i-0c45b8ee84e5dd4c9 (WEB-TIER-INSTANCE)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-0c45b8ee84e5dd4c9
IPv6 address
–
Public IPv4 address
3.93.146.218 | open address
Instance state
Running
Private IPv4 addresses
10.0.6.29
Public IPv4 DNS
ec2-3-93-146-218.compute-1.amazonaws.com | open address


aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2 > Load balancers > EXTERNAL-WEB-ALB
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Amazon Machine Images (AMIs) (2) Info
Recycle Bin
EC2 Image Builder
Actions
Launch instance from AMI
Owned by me
Find AMI by attribute or tag
1
Name
AMI name
AMI ID
Source
Owner
Visibility
WEB-TIER-GOLDEN-AMI
ami-09923c3caac453f01
515966521120/WEB-TIER-GOLDEN-AMI
515966521120
Private
APP-TIER-GOLDEN-AMI
ami-046092f8e57c0a943
515966521120/APP-TIER-GOLDEN-AMI
515966521120
Private
Select an AMI
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
88°F
ENG
16:01

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - *required*

APPP-TIER-LAUNCH-TEMPLATE

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

APPP-TIER-LAUNCH-TEMPLATE

Max 255 chars

Auto Scaling guidance | Info

Select this if you intend to use this template with EC2 Auto Scaling

☑ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

▶ **Template tags**

▶ **Source template**

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

▼ Application and OS Images (Amazon Machine Image) - required Info

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

▼ Summary

Software Image (AMI)
-

Virtual server type (instance type)
-

Firewall (security group)
-

Storage (volumes)
-

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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel | Create launch template

AWS, Marketplace and the Community

Amazon Machine Image (AMI)

APP-TIER-GOLDEN-AMI
ami-046092f8e57c0a943
2024-12-22T10:28:06.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

Description

APP-TIER-GOLDEN-AMI

Architecture

x86_64

AMI ID

ami-046092f8e57c0a943

Instance type

Info | Get advice

Advanced

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Key pair (login)

Info

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

Summary

Software Image (AMI)
APP-TIER-GOLDEN-AMI
ami-046092f8e57c0a943

Virtual server type (instance type)
t2.micro

Firewall (security group)
-

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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Create launch template

Add new volume

Resource tags Info

No resource tags are currently included in this template. Add a resource tag to include it in the launch template.

Add new tag

You can add up to 50 more tags.

Advanced details Info

IAM instance profile | Info

3-tier-role
arn:aws:iam::515966521120:instance-profile/3-tier-role

Create new IAM profile

Hostname type | Info

Don't include in launch template

DNS Hostname | Info

- [ ] Enable resource-based IPv4 (A record) DNS requests
- [ ] Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery | Info

Don't include in launch template

Shutdown behavior | Info

Summary

Software Image (AMI)
APP-TIER-GOLDEN-AMI
ami-046092f8e57c0a943

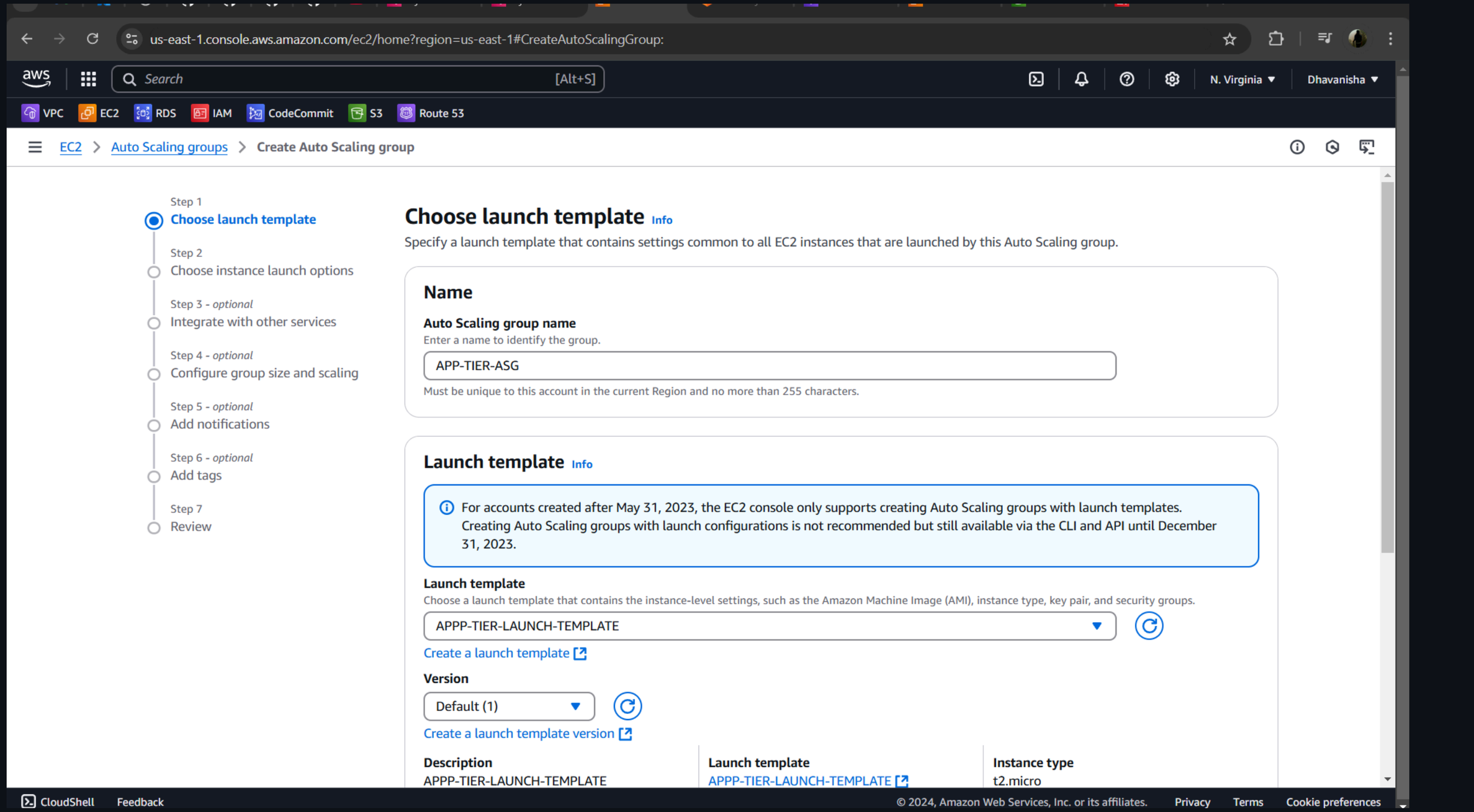
Virtual server type (instance type)
t2.micro

Firewall (security group)
APP-SG

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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel | Create launch template


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:
aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2
Auto Scaling groups
Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Integrate with other services
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Choose launch template Info
Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.
Name
Auto Scaling group name
Enter a name to identify the group.
APP-TIER-ASG
Must be unique to this account in the current Region and no more than 255 characters.
Launch template Info
For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.
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.
APPP-TIER-LAUNCH-TEMPLATE
Create a launch template
Version
Default (1)
Create a launch template version
Description
APPP-TIER-LAUNCH-TEMPLATE
Launch template
APPP-TIER-LAUNCH-TEMPLATE
Instance type
t2.micro
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 6 - *optional*
Add tags

Step 7
Review

t2.micro

Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-0b3c5059ed8826ce7 (3-tier-project-vpc)
10.0.0.0/16

Create a VPC

Availability Zones and subnets

Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

us-east-1b | subnet-0398b575e3d6a5f2d (3-tier-project-subnet-APP-2-us-east-1b)
10.0.144.0/20

us-east-1a | subnet-063705093f9bbf690 (3-tier-project-subnet-APP-1-us-east-1a)
10.0.128.0/20

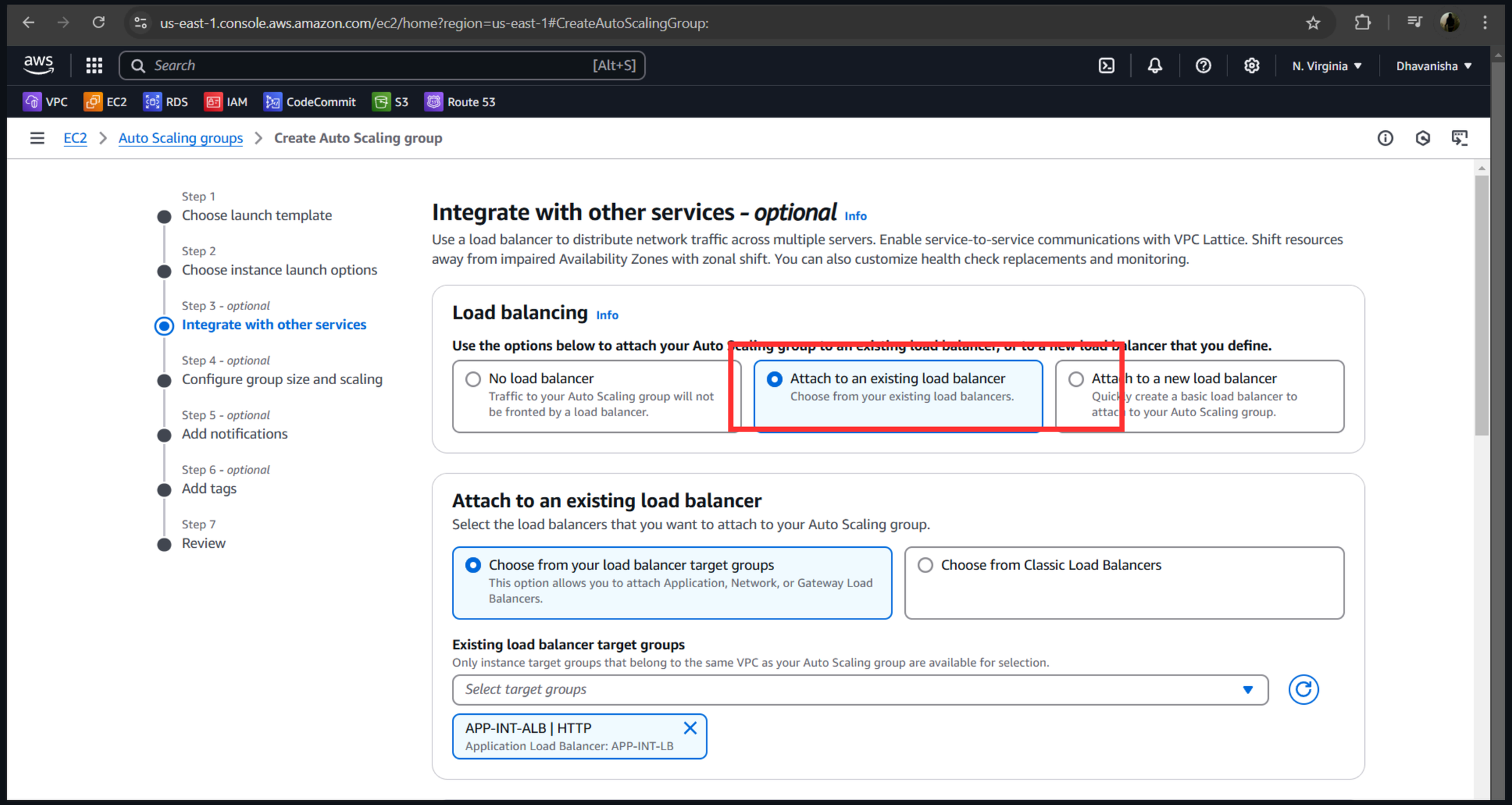
Create a subnet

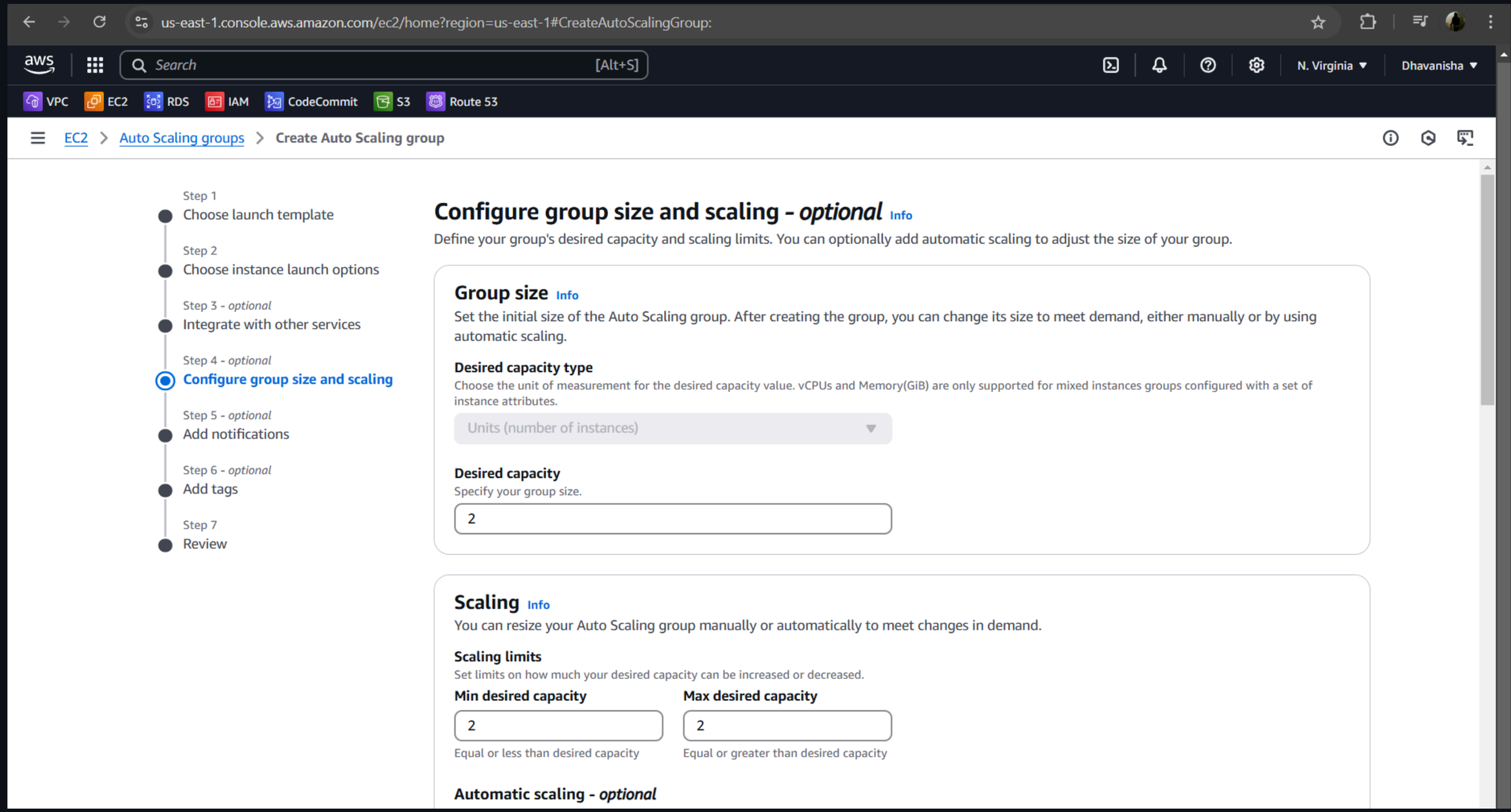
Availability Zone distribution - *new*

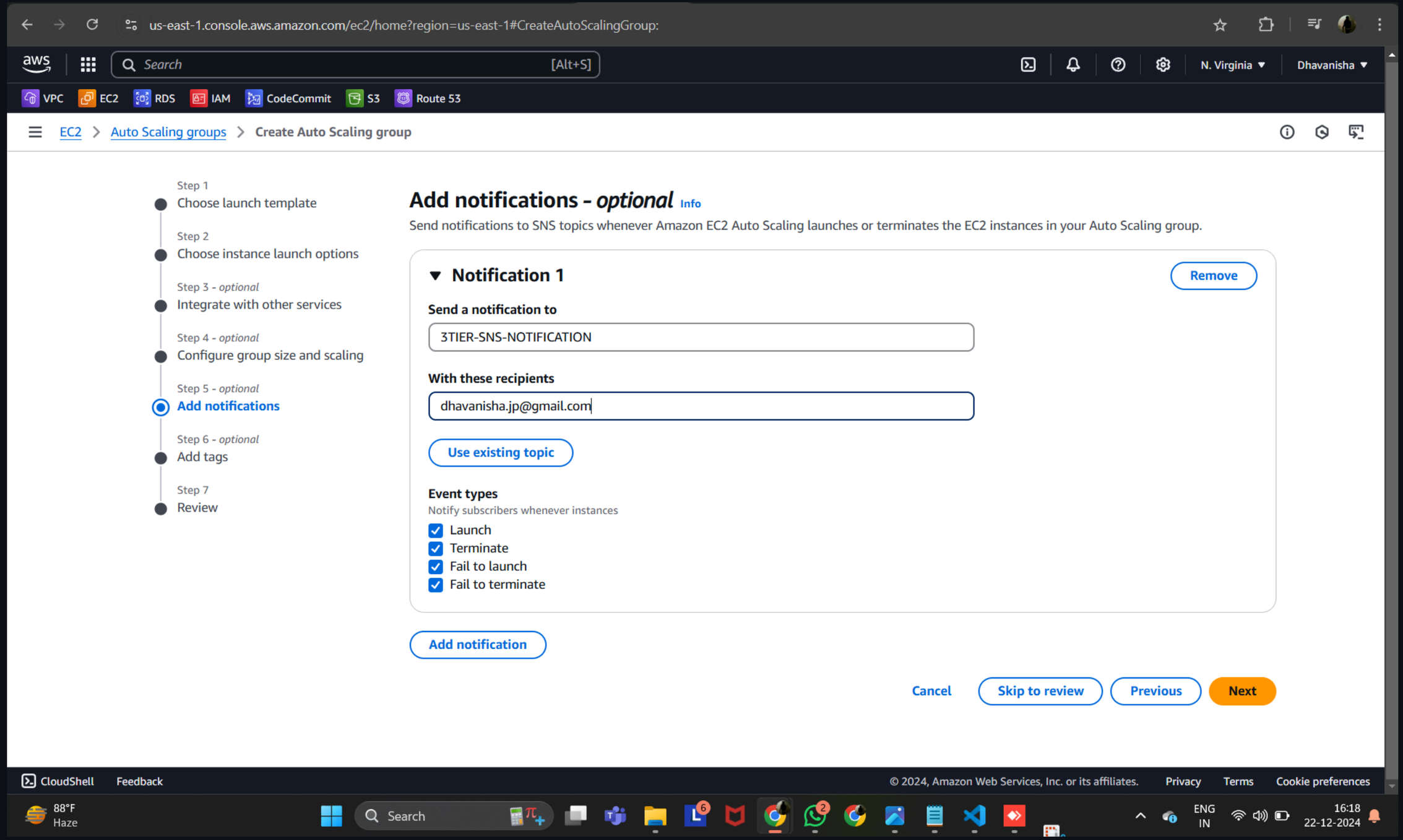
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

- (●) **Balanced best effort**
  If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.
- ( ) **Balanced only**
  If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve

Linked my existing load balancer to autoscaling




us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:
aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2
Auto Scaling groups
Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Integrate with other services
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Configure group size and scaling - optional
Info
Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.
Group size
Info
Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.
Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.
Units (number of instances)
Desired capacity
Specify your group size.
2
Scaling
Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.
Scaling limits
Set limits on how much your desired capacity can be increased or decreased.
Min desired capacity
2
Equal or less than desired capacity
Max desired capacity
2
Equal or greater than desired capacity
Automatic scaling - optional


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:
Search [Alt+S]
N. Virginia
Dhavanisha
VPC EC2 RDS IAM CodeCommit S3 Route 53
EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Integrate with other services
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Add notifications - optional Info
Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.
Notification 1
Remove
Send a notification to
3TIER-SNS-NOTIFICATION
With these recipients
dhavanisha.jp@gmail.com
Use existing topic
Event types
Notify subscribers whenever instances
Launch
Terminate
Fail to launch
Fail to terminate
Add notification
Cancel
Skip to review
Previous
Next
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
88°F
Haze
Search
ENG
IN
16:18
22-12-2024

Step 1
Choose launch template

Step 2
Choose instance launch options

Step 3 - *optional*
Integrate with other services

Step 4 - *optional*
Configure group size and scaling

Step 5 - *optional*
Add notifications

Step 6 - *optional*
Add tags

Step 7
Review

Add tags - *optional* Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.

You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group.

Tags (1)

Key	Value - optional	Tag new instances	
Name	APP-TIER-INSTANCE	☑	Remove

Add tag

49 remaining

Cancel Previous Next

Instance scale-in protection	Monitoring	Default instance warmup
Disabled	Disabled	Disabled

Capacity Reservation preference

Preference	Capacity Reservation IDs	Resource Groups
Default	-	-

Step 5: Add notifications

Edit

Notifications

Notification 1
SNS Topic
3TIER-SNS-NOTIFICATION (dhavanisha.jp@gmail.com)

Event types
- [x] Launch
- [x] Terminate
- [x] Fail to launch
- [x] Fail to terminate

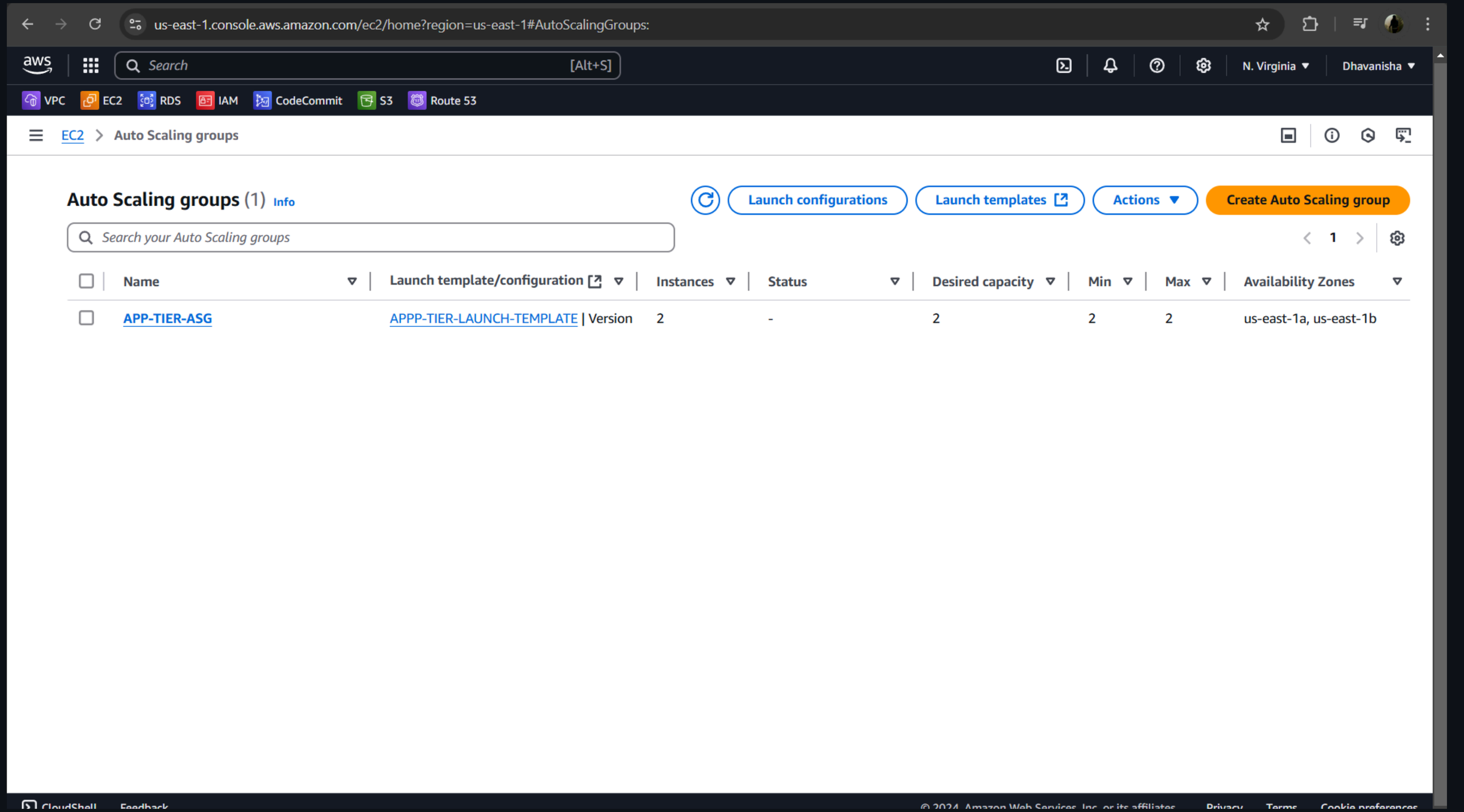
Step 6: Add tags

Edit

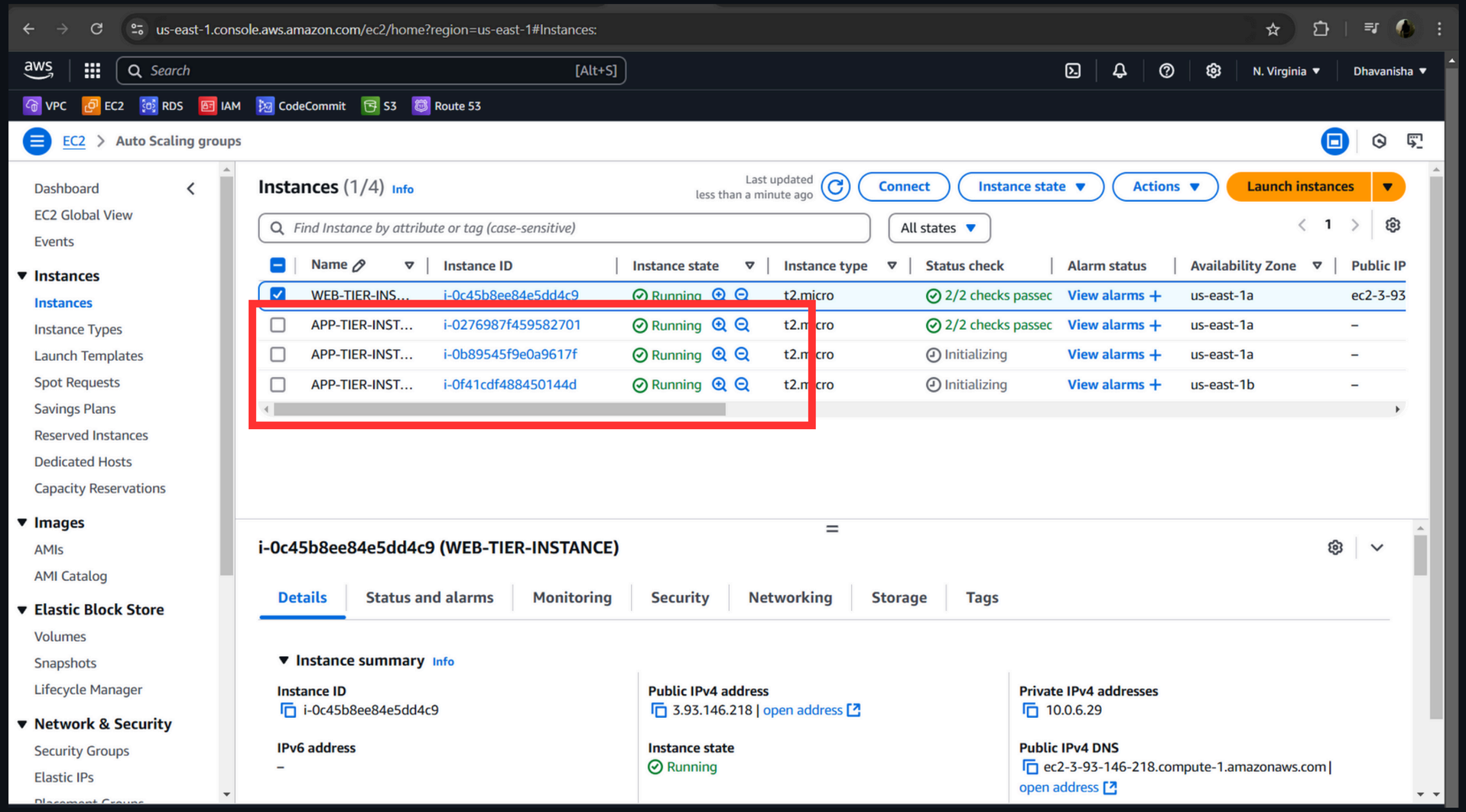
Tags (1)

Key	Value	Tag new instances
Name	APP-TIER-INSTANCE	Yes

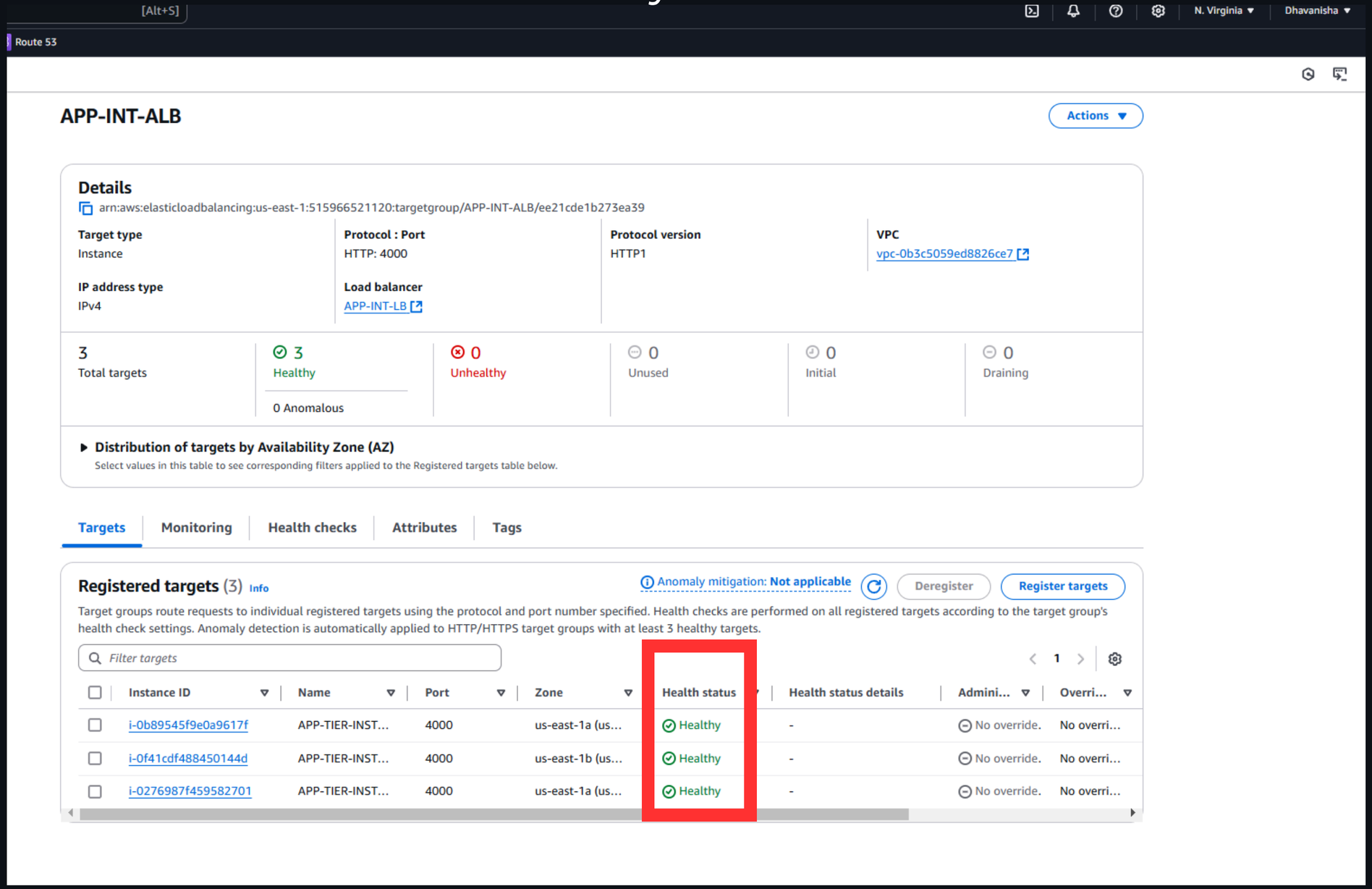
Preview code | Cancel | Previous | Create Auto Scaling group


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroups:
aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2
Auto Scaling groups
Auto Scaling groups (1) Info
Launch configurations
Launch templates
Actions
Create Auto Scaling group
Search your Auto Scaling groups
1
Name
Launch template/configuration
Instances
Status
Desired capacity
Min
Max
Availability Zones
APP-TIER-ASG
APPP-TIER-LAUNCH-TEMPLATE | Version
2
-
2
2
2
us-east-1a, us-east-1b
CloudShell
Feedback
Privacy
Terms
Cookie preferences

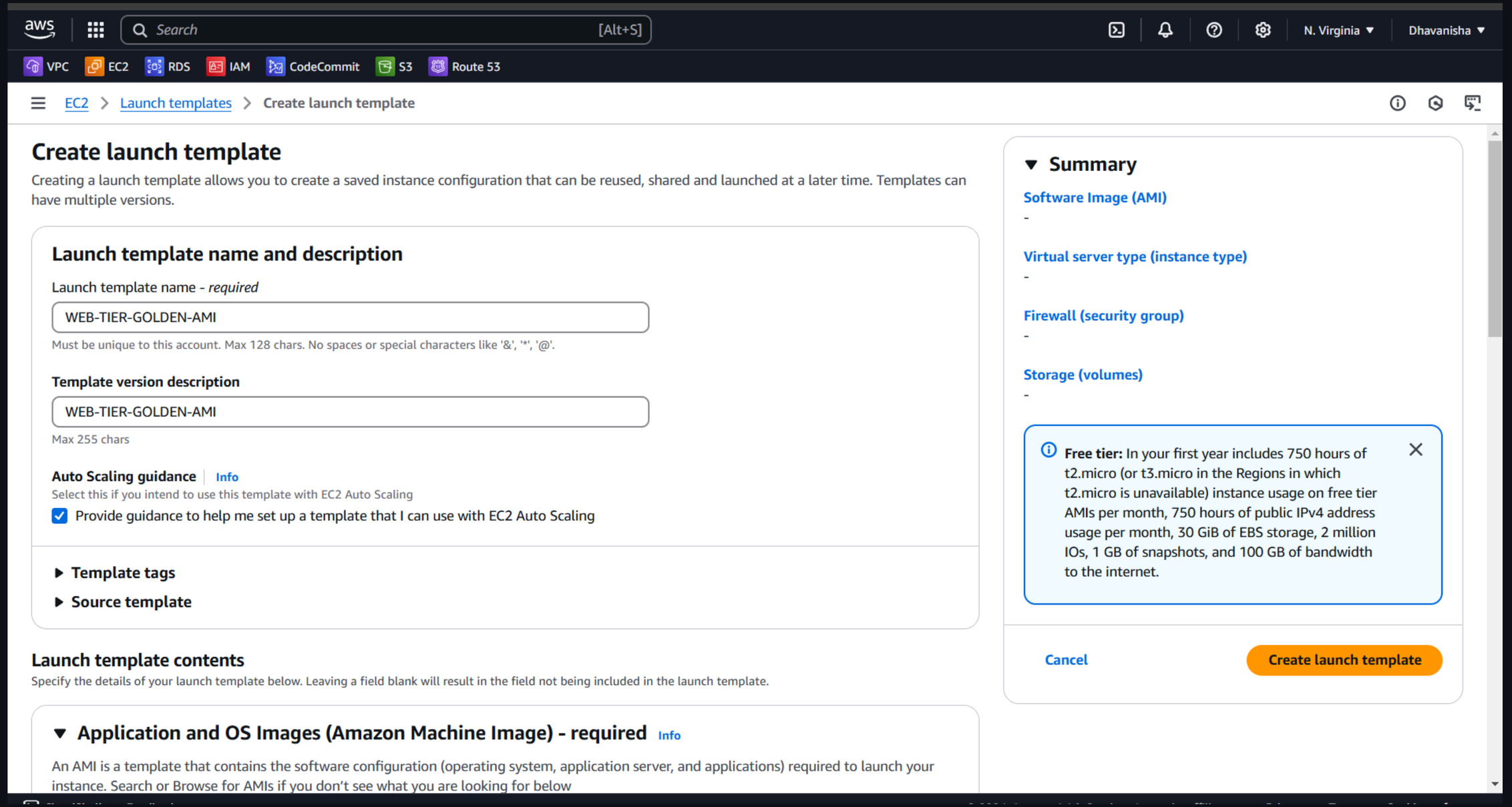
Autoscaling created for App-tier-instance

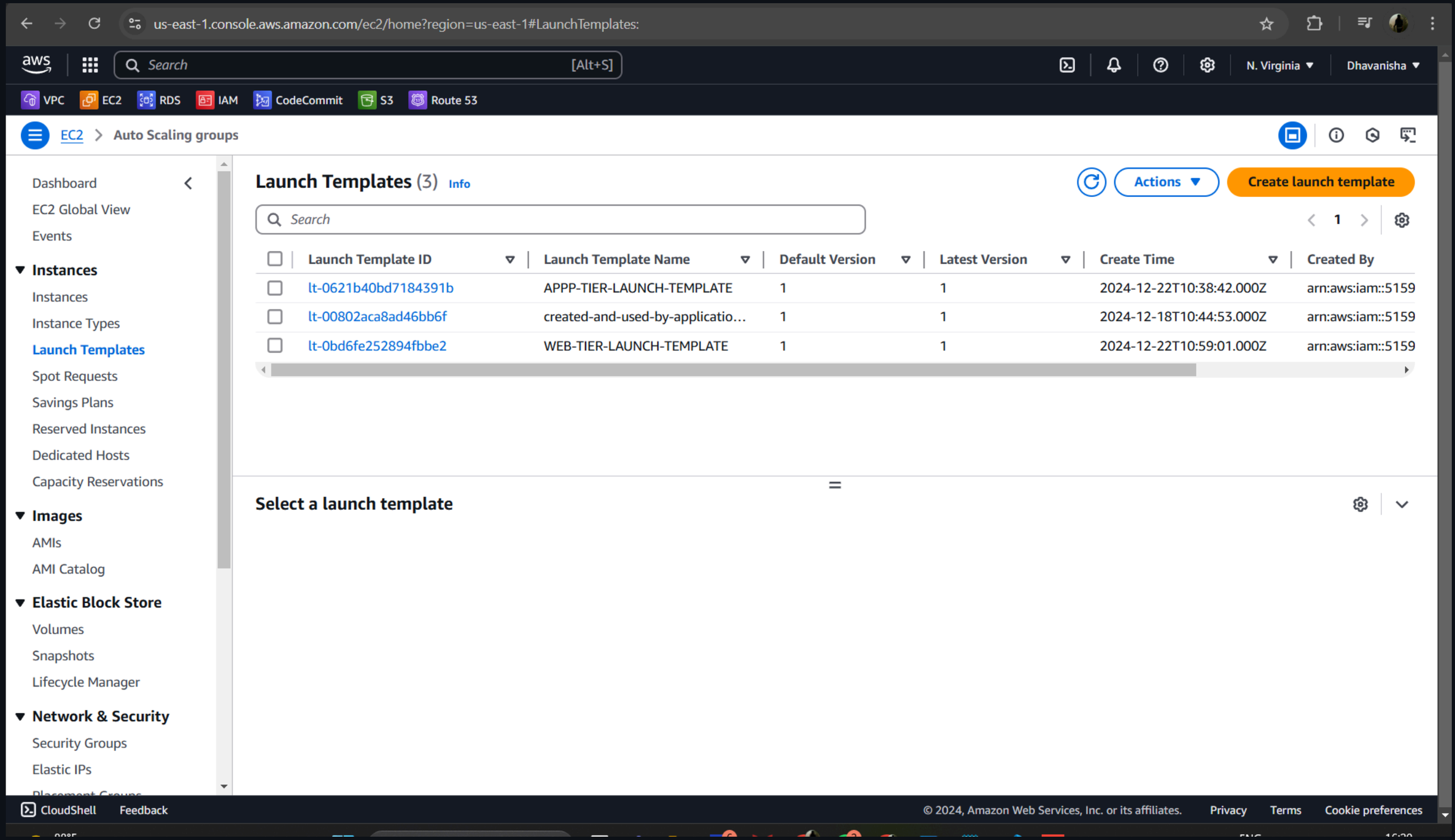


navigate to the target group. Check the status, all instances are in healthy condition.



The web tier follows similar techniques to the app tier.




us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchTemplates:
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2
Auto Scaling groups
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Launch Templates (3) Info
Actions
Create launch template
Search
1
Launch Template ID
Launch Template Name
Default Version
Latest Version
Create Time
Created By
lt-0621b40bd7184391b
APPP-TIER-LAUNCH-TEMPLATE
1
1
2024-12-22T10:38:42.000Z
arn:aws:iam::5159
lt-00802aca8ad46bb6f
created-and-used-by-applicatio...
1
1
2024-12-18T10:44:53.000Z
arn:aws:iam::5159
lt-0bd6fe252894fbbe2
WEB-TIER-LAUNCH-TEMPLATE
1
1
2024-12-22T10:59:01.000Z
arn:aws:iam::5159
Select a launch template
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - *optional*
Integrate with other services
Step 4 - *optional*
Configure group size and scaling
Step 5 - *optional*
Add notifications
Step 6 - *optional*
Add tags
Step 7
Review

Choose launch template Info

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name

Enter a name to identify the group.

WEB-TIER-ASG

Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

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.

WEB-TIER-LAUNCH-TEMPLATE

Create a launch template

Version

Default (1)

Create a launch template version

Description	Launch template	Instance type
WEB-TIER-LAUNCH-TEMPALTE	WEB-TIER-LAUNCH-TEMPLATE	t2.micro

Step 1
Choose launch template

Step 2
Choose instance launch options

Step 3 - *optional*
Integrate with other services

Step 4 - *optional*
Configure group size and scaling

Step 5 - *optional*
Add notifications

Step 6 - *optional*
Add tags

Step 7
Review

Configure group size and scaling - *optional* Info

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

Group size Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity

2

Equal or less than desired capacity

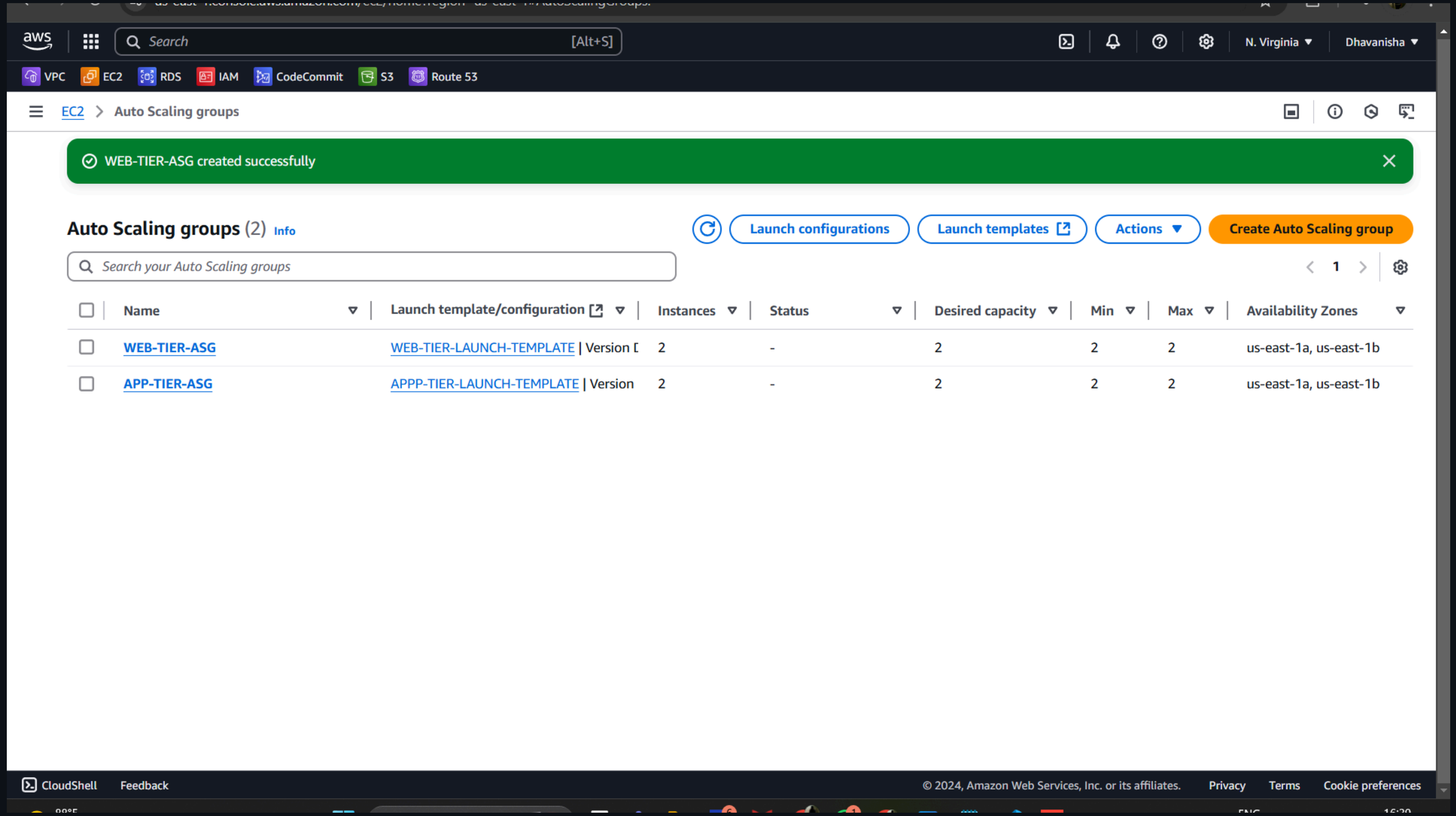
Max desired capacity

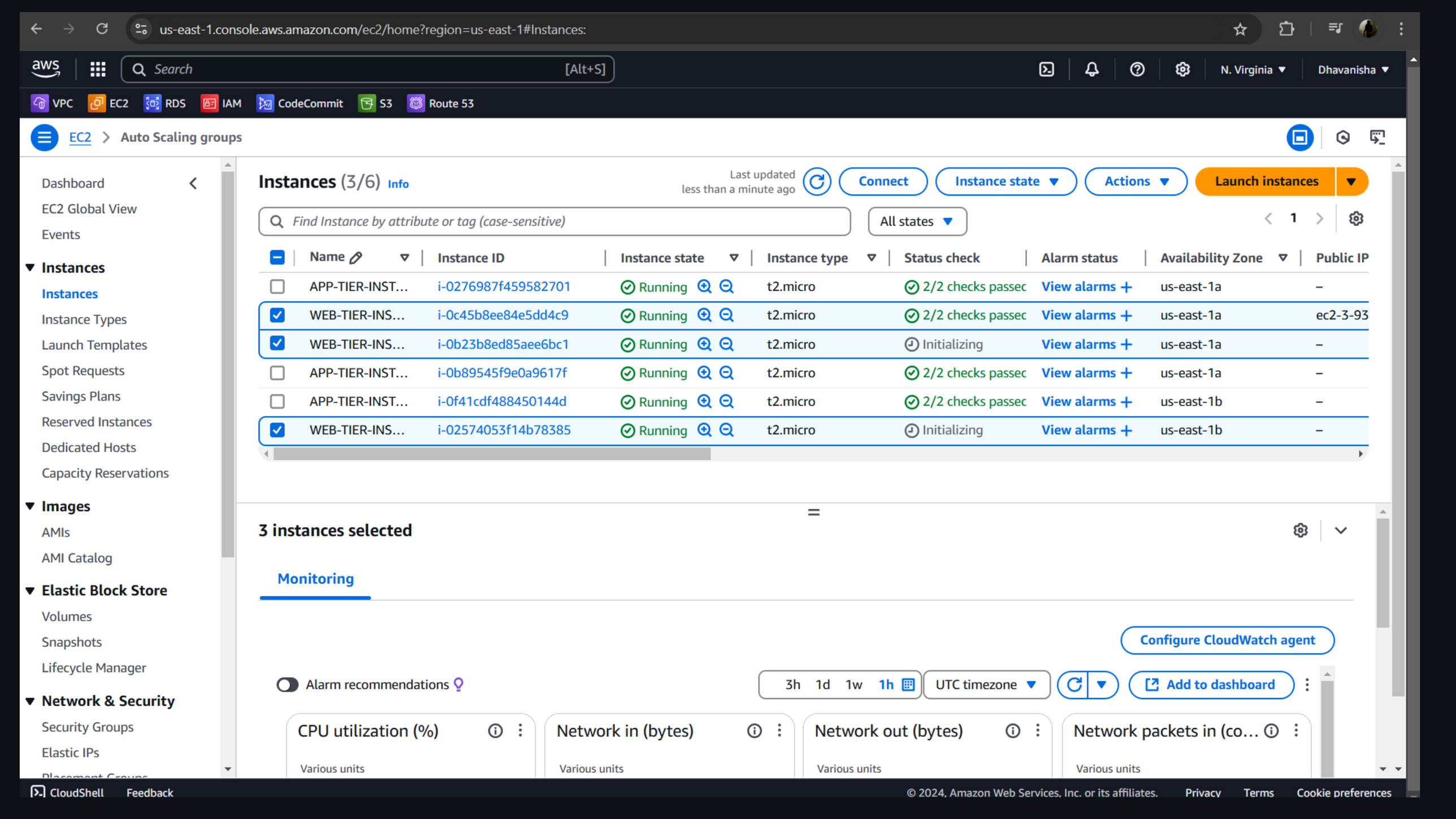
2

Equal or greater than desired capacity

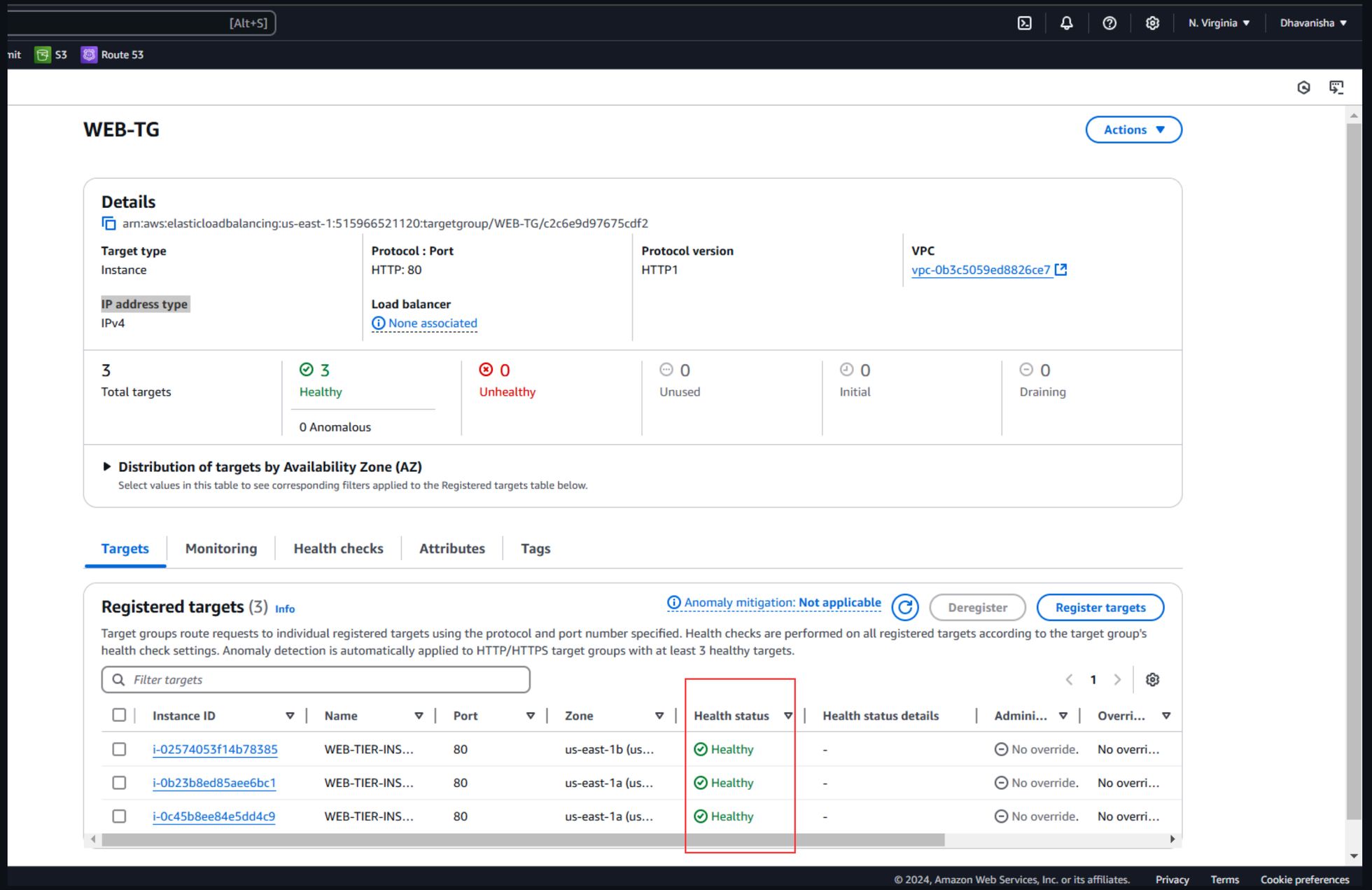
Automatic scaling - *optional*

Choose whether to use a target tracking policy Info

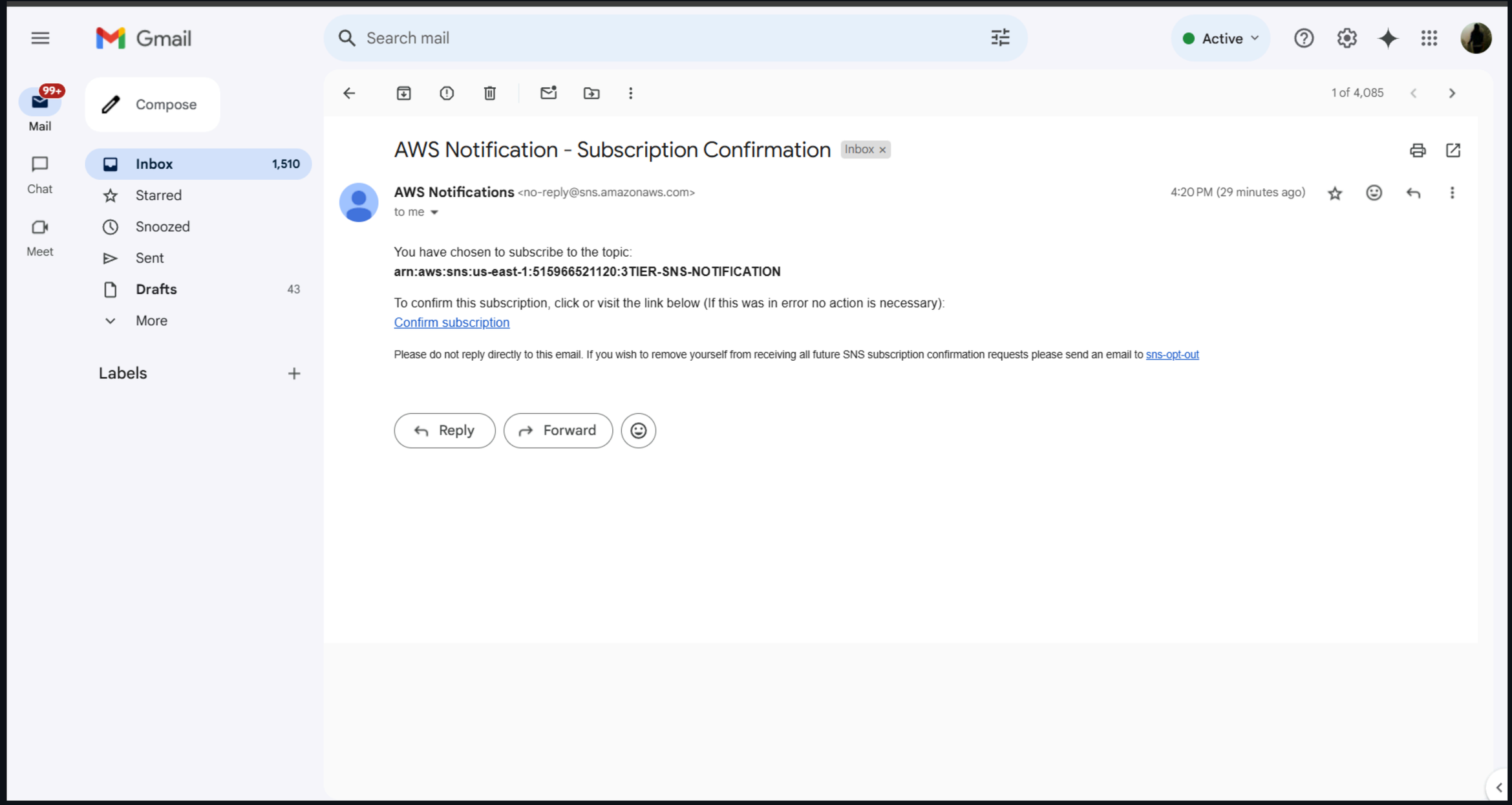

aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
EC2
Auto Scaling groups
WEB-TIER-ASG created successfully
Auto Scaling groups (2) Info
Launch configurations
Launch templates
Actions
Create Auto Scaling group
Search your Auto Scaling groups
1
Name
Launch template/configuration
Instances
Status
Desired capacity
Min
Max
Availability Zones
WEB-TIER-ASG
WEB-TIER-LAUNCH-TEMPLATE | Version [
2
-
2
2
2
us-east-1a, us-east-1b
APP-TIER-ASG
APPP-TIER-LAUNCH-TEMPLATE | Version
2
-
2
2
2
us-east-1a, us-east-1b
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

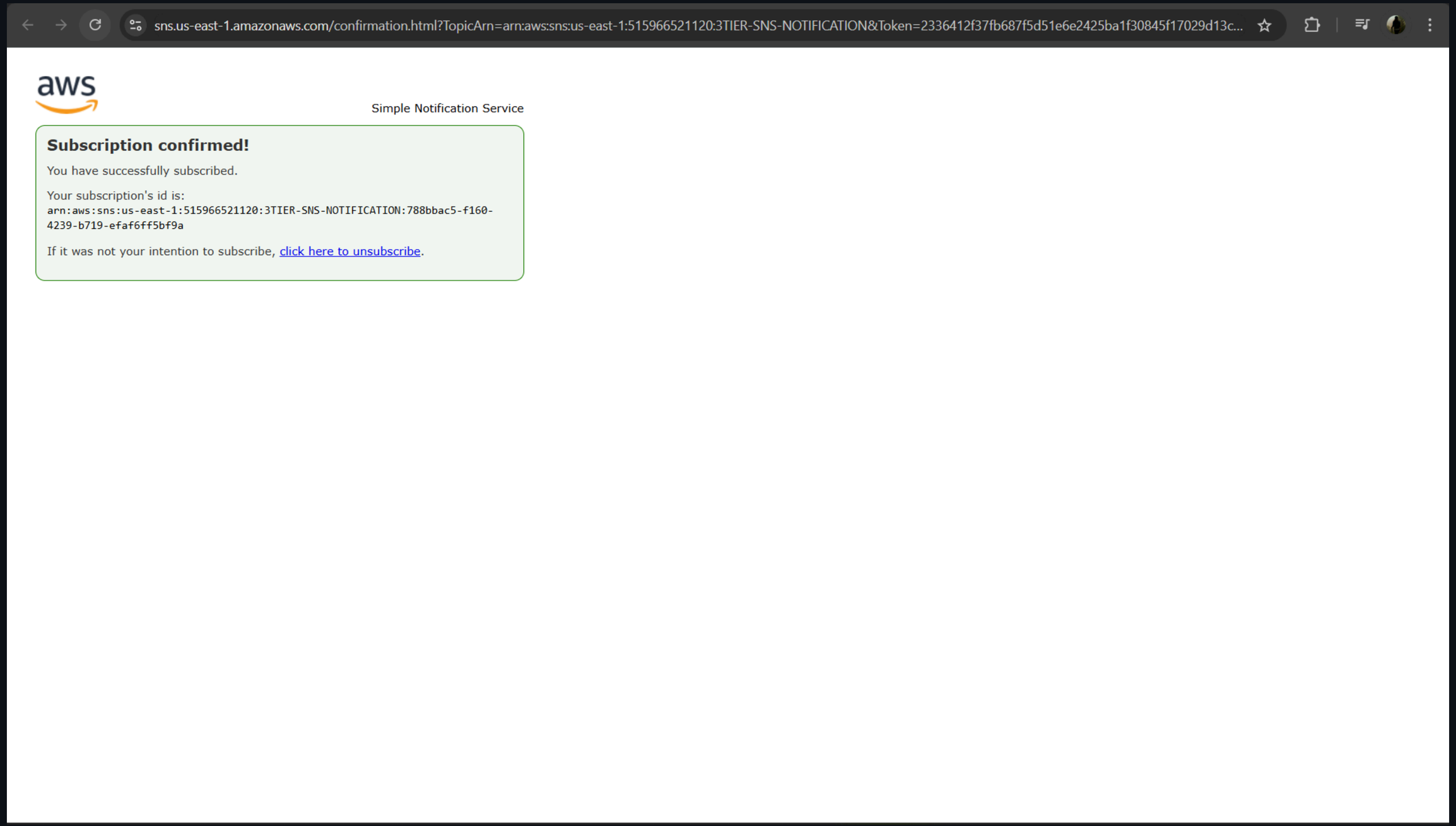

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:
Search [Alt+S]
N. Virginia
Dhavanisha
VPC EC2 RDS IAM CodeCommit S3 Route 53
EC2 > Auto Scaling groups
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Instances (3/6) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IP
APP-TIER-INST... i-0276987f459582701 Running t2.micro 2/2 checks passec View alarms + us-east-1a –
WEB-TIER-INS... i-0c45b8ee84e5dd4c9 Running t2.micro 2/2 checks passec View alarms + us-east-1a ec2-3-93
WEB-TIER-INS... i-0b23b8ed85aee6bc1 Running t2.micro Initializing View alarms + us-east-1a –
APP-TIER-INST... i-0b89545f9e0a9617f Running t2.micro 2/2 checks passec View alarms + us-east-1a –
APP-TIER-INST... i-0f41cdf488450144d Running t2.micro 2/2 checks passec View alarms + us-east-1b –
WEB-TIER-INS... i-02574053f14b78385 Running t2.micro Initializing View alarms + us-east-1b –
3 instances selected
Monitoring
Configure CloudWatch agent
Alarm recommendations
3h 1d 1w 1h
UTC timezone
Add to dashboard
CPU utilization (%)
Network in (bytes)
Network out (bytes)
Network packets in (co...
Various units
CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This is also in healthy condition.

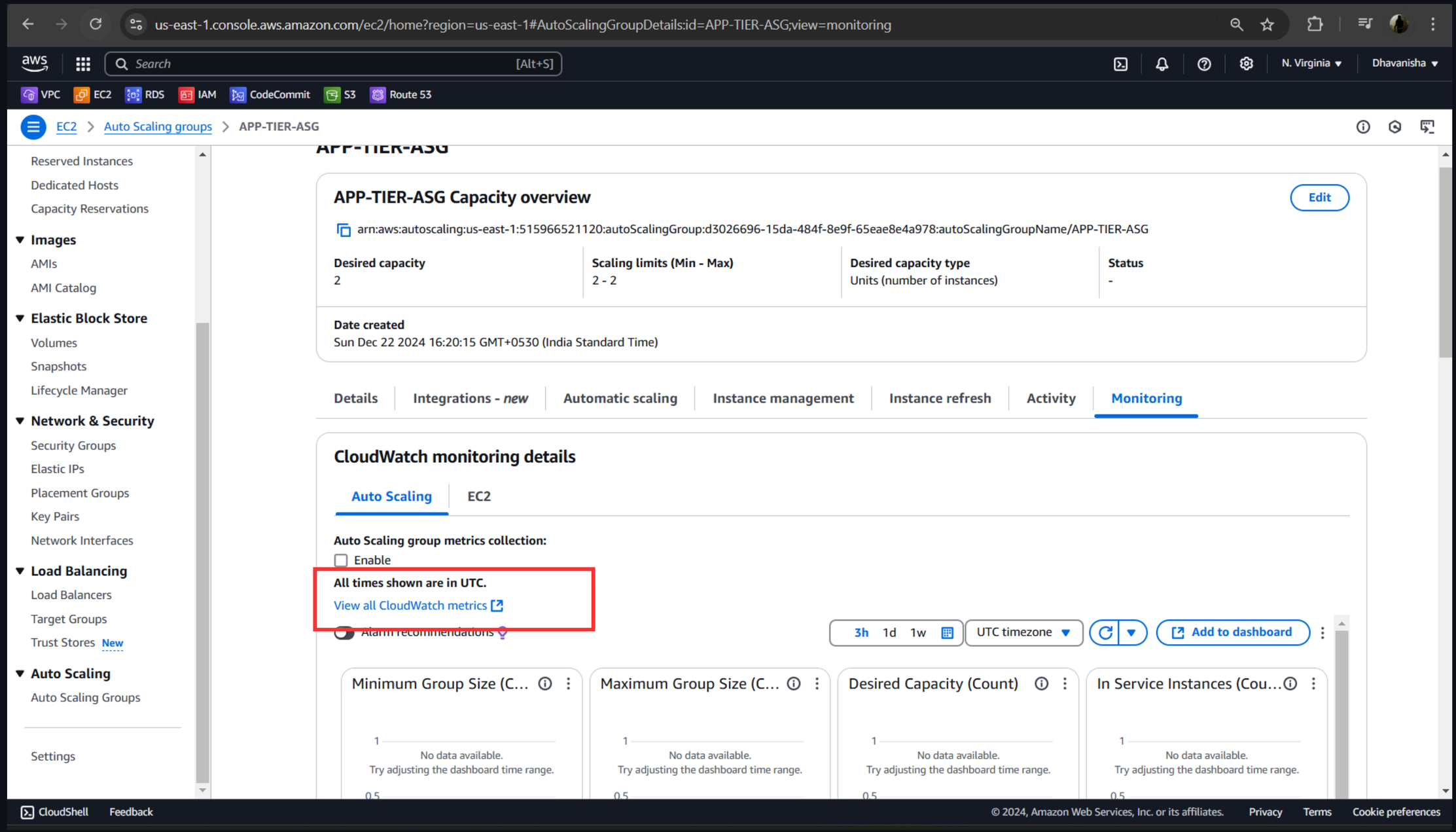


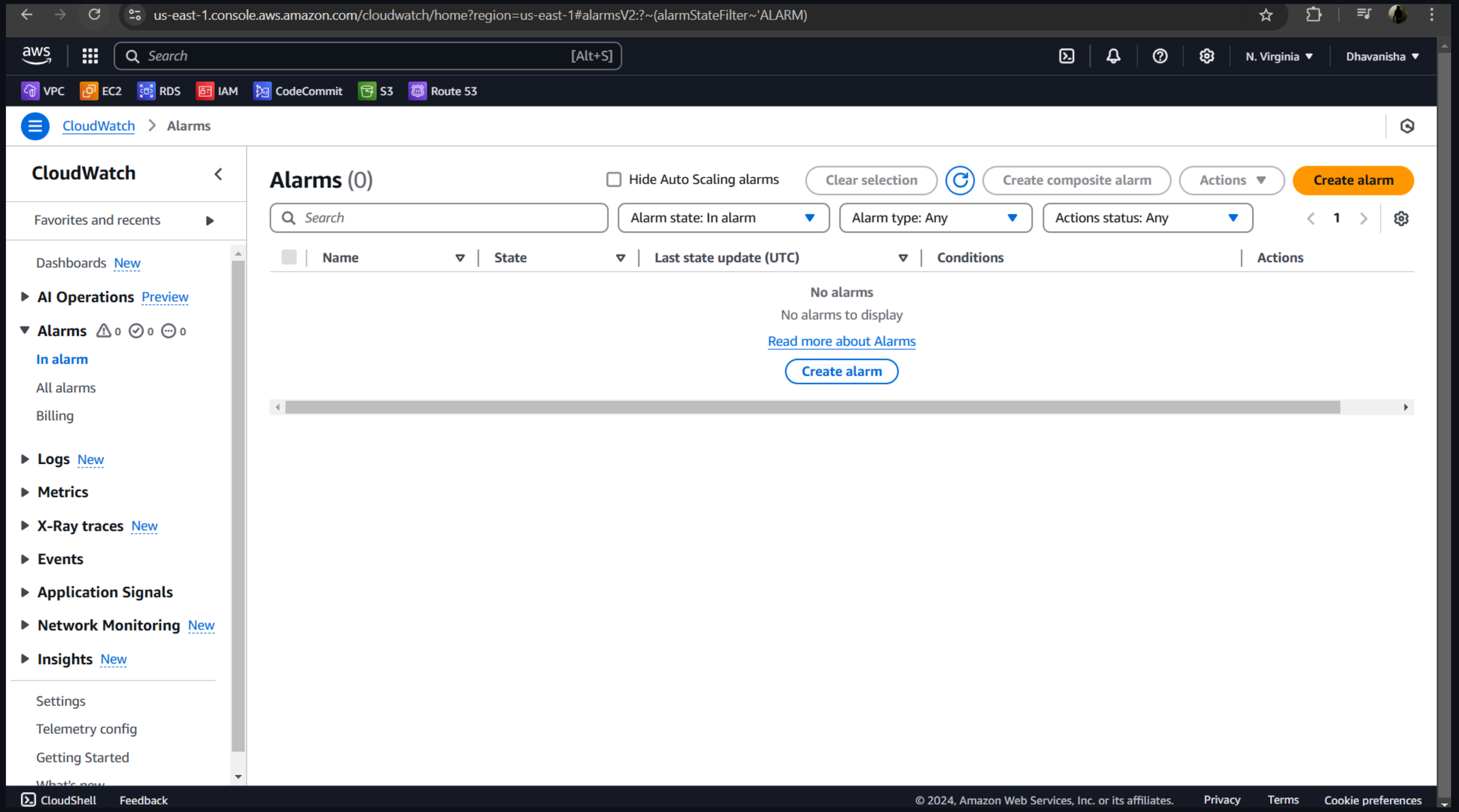
got a sns notification from aws

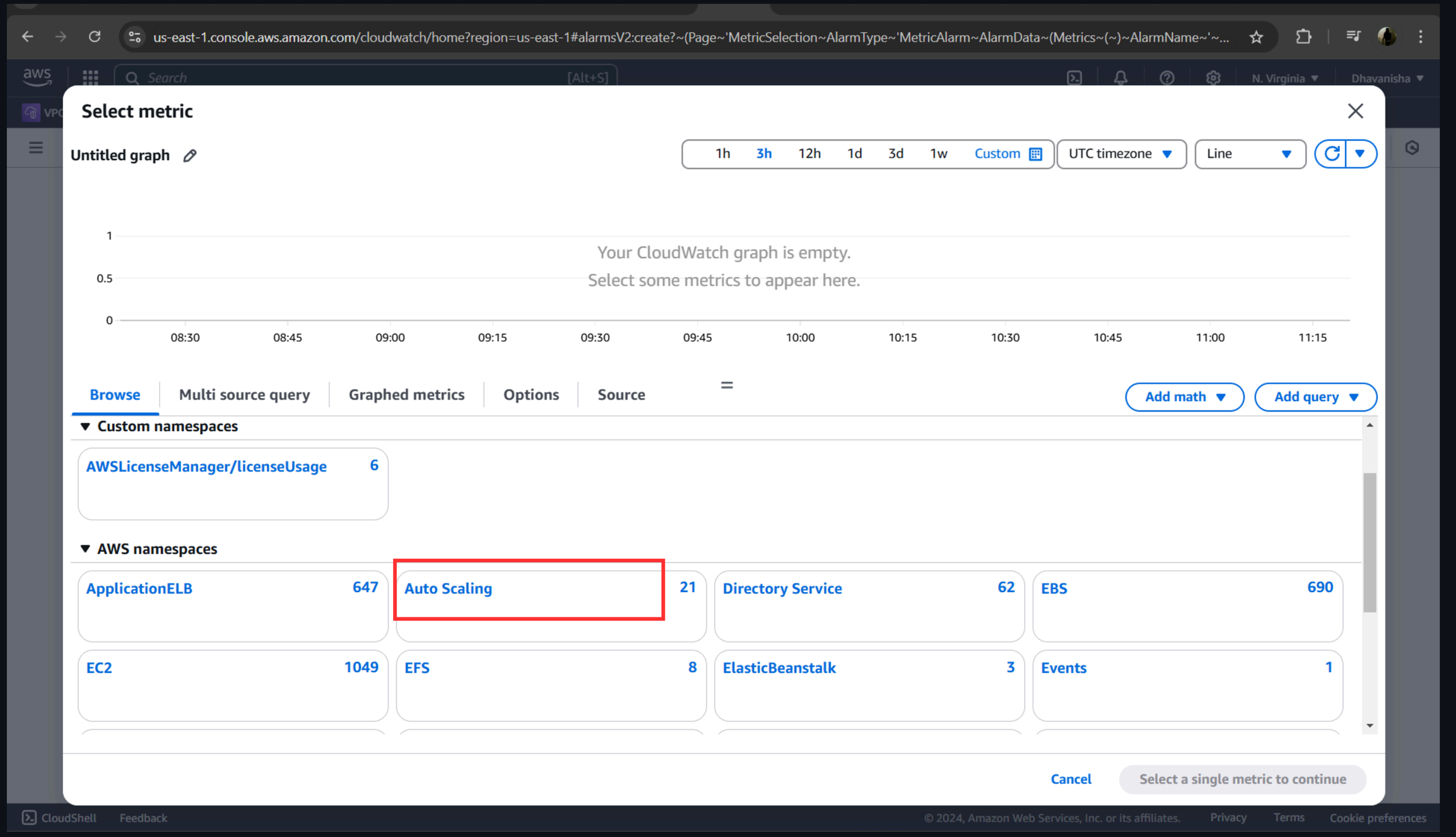


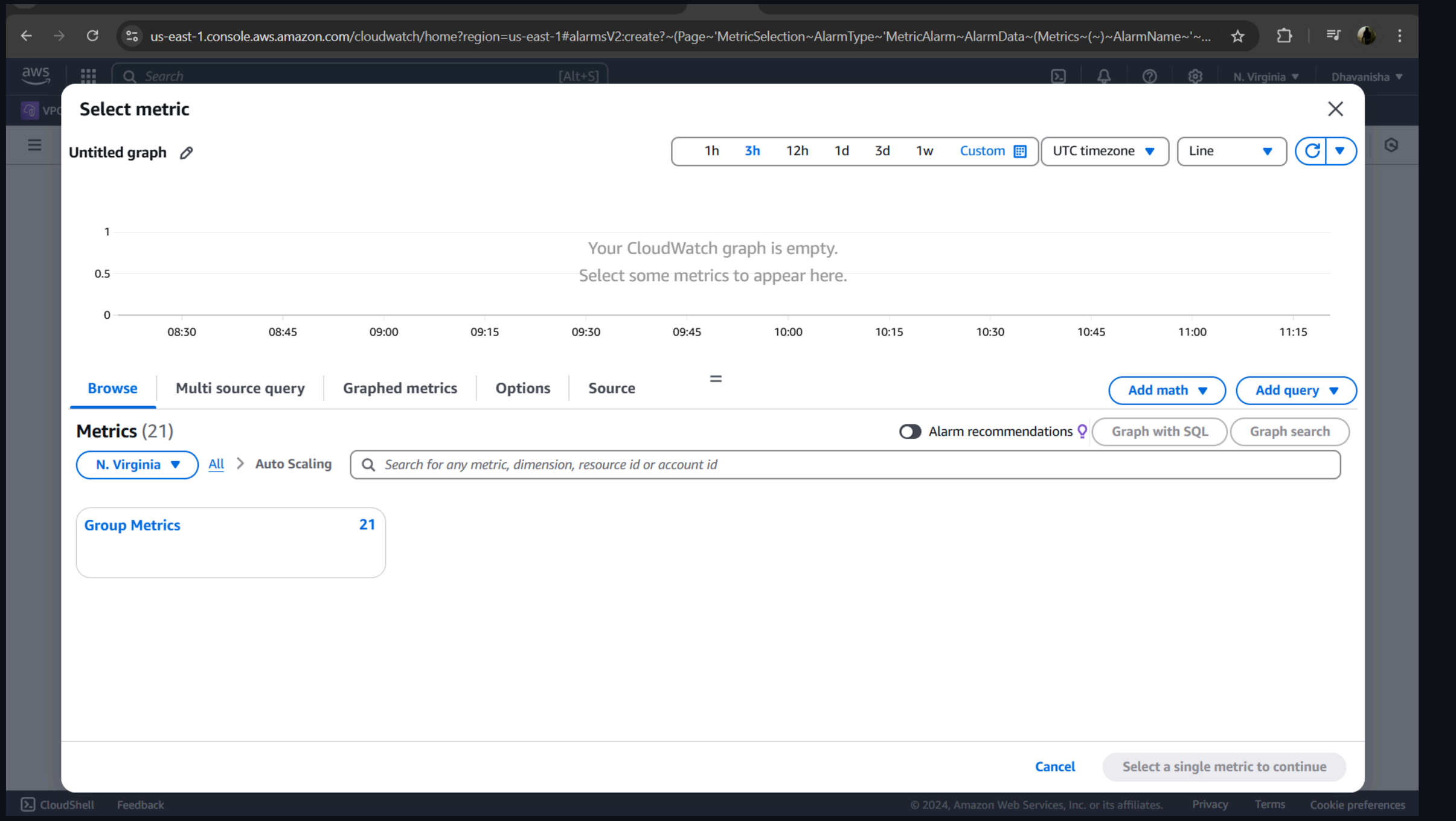

sns.us-east-1.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:us-east-1:515966521120:3TIER-SNS-NOTIFICATION&Token=2336412f37fb687f5d51e6e2425ba1f30845f17029d13c...
aws
Simple Notification Service
Subscription confirmed!
You have successfully subscribed.
Your subscription's id is:
arn:aws:sns:us-east-1:515966521120:3TIER-SNS-NOTIFICATION:788bbac5-f160-4239-b719-efaf6ff5bf9a
If it was not your intention to subscribe, click here to unsubscribe.

Created a cloud watch for Monitoring a application performance




us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#alarmsV2:?~(alarmStateFilter~'ALARM)
aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
CloudWatch
Alarms
CloudWatch
Favorites and recents
Dashboards New
AI Operations Preview
Alarms 0 0 0
In alarm
All alarms
Billing
Logs New
Metrics
X-Ray traces New
Events
Application Signals
Network Monitoring New
Insights New
Settings
Telemetry config
Getting Started
Alarms (0)
Hide Auto Scaling alarms
Clear selection
Create composite alarm
Actions
Create alarm
Search
Alarm state: In alarm
Alarm type: Any
Actions status: Any
1
Name
State
Last state update (UTC)
Conditions
Actions
No alarms
No alarms to display
Read more about Alarms
Create alarm
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#alarmsV2:create?~(Page~'MetricSelection~AlarmType~'MetricAlarm~AlarmData~(Metrics~(~)~AlarmName~'~...
Search
[Alt+S]
N. Virginia
Dhavanisha
Select metric
Untitled graph
1h
3h
12h
1d
3d
1w
Custom
UTC timezone
Line
1
0.5
0
Your CloudWatch graph is empty.
Select some metrics to appear here.
08:30
08:45
09:00
09:15
09:30
09:45
10:00
10:15
10:30
10:45
11:00
11:15
Browse
Multi source query
Graphed metrics
Options
Source
Add math
Add query
Custom namespaces
AWSLicenseManager/licenseUsage 6
AWS namespaces
ApplicationELB 647
Auto Scaling 21
Directory Service 62
EBS 690
EC2 1049
EFS 8
ElasticBeanstalk 3
Events 1
Cancel
Select a single metric to continue
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#alarmsV2:create?~(Page~'MetricSelection~AlarmType~'MetricAlarm~AlarmData~(Metrics~(~)~AlarmName~'~...
Search
[Alt+S]
N. Virginia
Dhavanisha
Select metric
Untitled graph
1h
3h
12h
1d
3d
1w
Custom
UTC timezone
Line
1
0.5
0
Your CloudWatch graph is empty.
Select some metrics to appear here.
08:30
08:45
09:00
09:15
09:30
09:45
10:00
10:15
10:30
10:45
11:00
11:15
Browse
Multi source query
Graphed metrics
Options
Source
Add math
Add query
Metrics (21)
Alarm recommendations
Graph with SQL
Graph search
N. Virginia
All
Auto Scaling
Search for any metric, dimension, resource id or account id
Group Metrics
21
Cancel
Select a single metric to continue
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Select metric

Untitled graph

1h | 3h | 12h | 1d | 3d | 1w | Custom | UTC timezone | Line

Browse | Multi source query | Graphed metrics (1) | Options | Source

Add math | Add query

☐	APP-TIER-ASG	WarmPoolWarmedCapacity	No alarms
☐	APP-TIER-ASG	GroupAndWarmPoolDesiredCapacity	No alarms
☐	APP-TIER-ASG	GroupMaxSize	No alarms
☐	APP-TIER-ASG	GroupInServiceCapacity	No alarms
☑	APP-TIER-ASG	GroupTotalInstances	No alarms
☐	APP-TIER-ASG	GroupTerminatingCapacity	No alarms
☐	APP-TIER-ASG	WarmPoolTotalCapacity	No alarms

Cancel | Select metric

Statistic

Average

Period

5 minutes

Conditions

Threshold type

- (●) **Static** — Use a value as a threshold
- ( ) **Anomaly detection** — Use a band as a threshold

Whenever GroupTotalInstances is...

Define the alarm condition.

- ( ) Greater — > threshold
- ( ) Greater/Equal — >= threshold
- ( ) Lower/Equal — <= threshold
- (●) Lower — < threshold

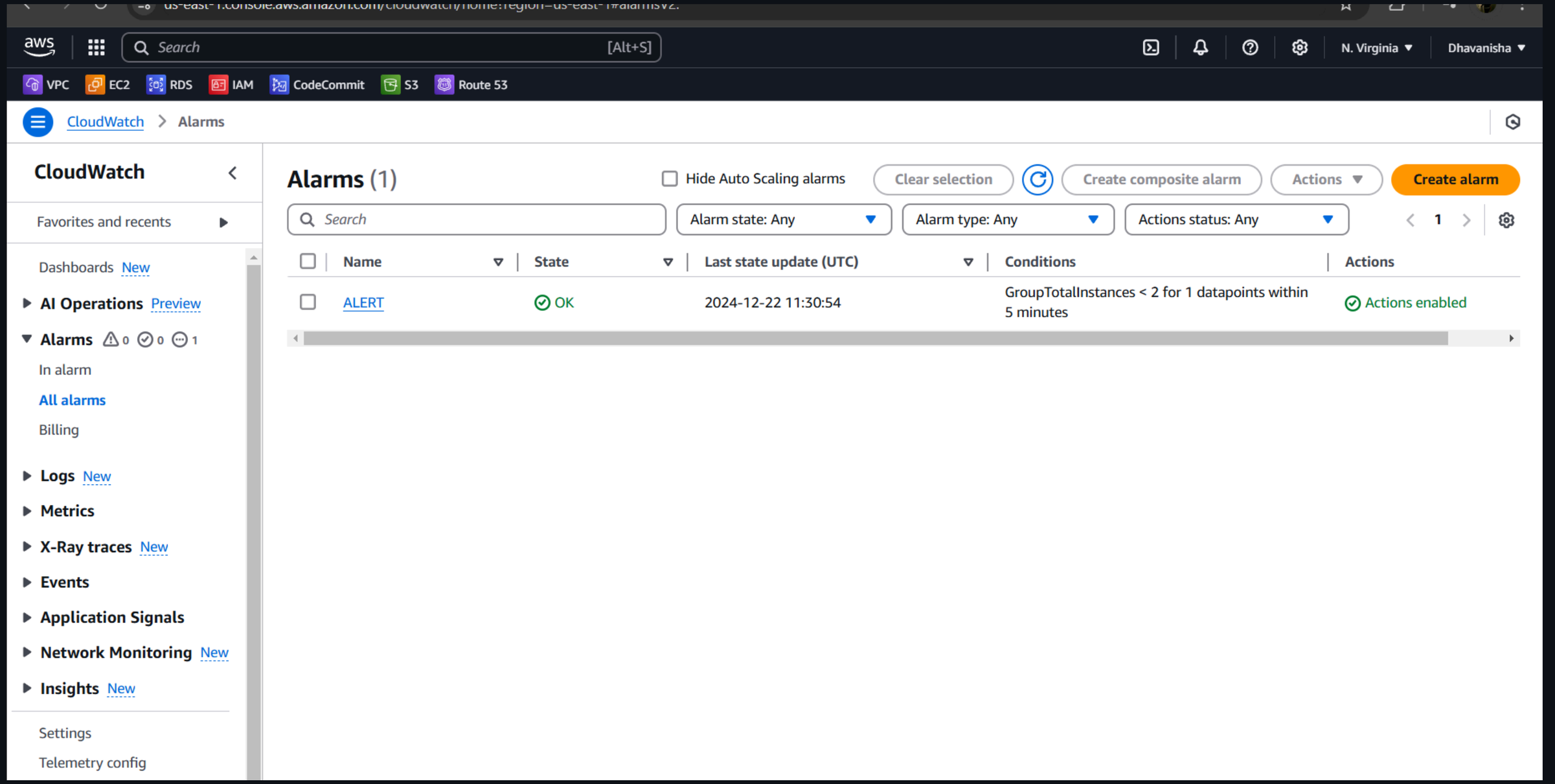
than...

Define the threshold value.

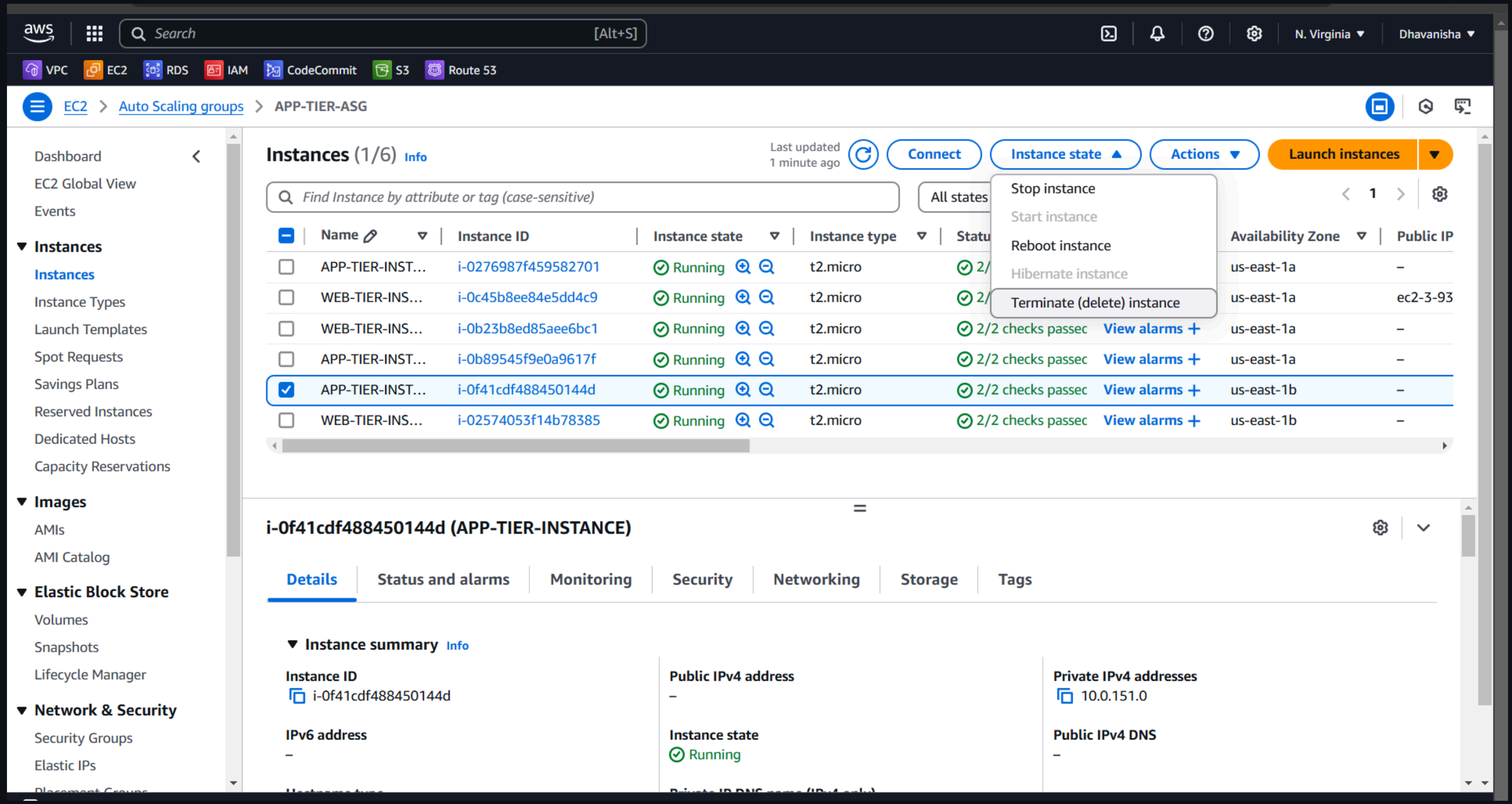
2

Must be a number

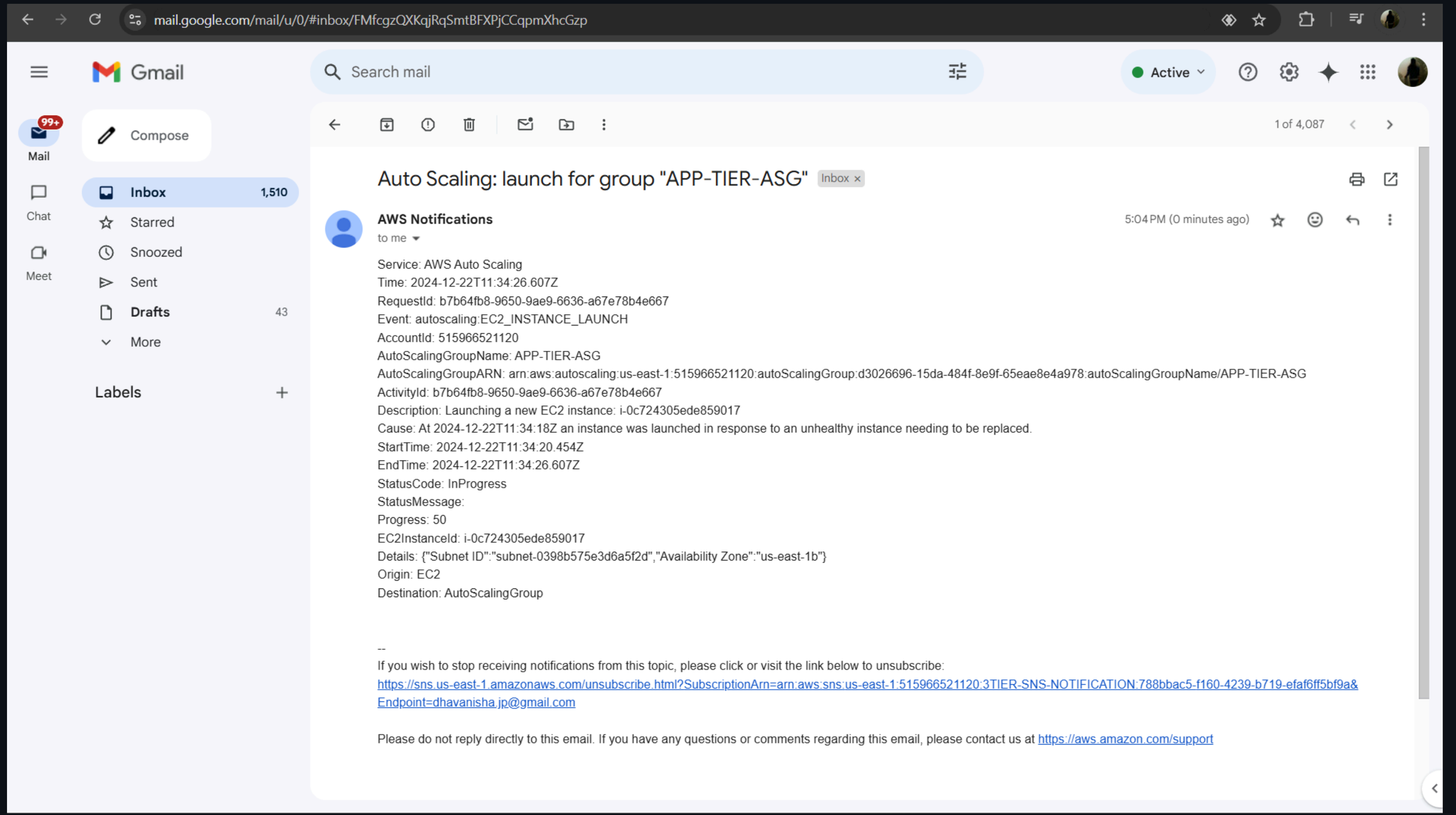
▶ **Additional configuration**


us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#alarmsV2:
aws
Search
[Alt+S]
N. Virginia
Dhavanisha
VPC
EC2
RDS
IAM
CodeCommit
S3
Route 53
CloudWatch
Alarms
CloudWatch
Favorites and recents
Dashboards New
AI Operations Preview
Alarms 0 0 1
In alarm
All alarms
Billing
Logs New
Metrics
X-Ray traces New
Events
Application Signals
Network Monitoring New
Insights New
Settings
Telemetry config
Alarms (1)
Hide Auto Scaling alarms
Clear selection
Create composite alarm
Actions
Create alarm
Search
Alarm state: Any
Alarm type: Any
Actions status: Any
1
Name
State
Last state update (UTC)
Conditions
Actions
ALERT
OK
2024-12-22 11:30:54
GroupTotalInstances < 2 for 1 datapoints within 5 minutes
Actions enabled

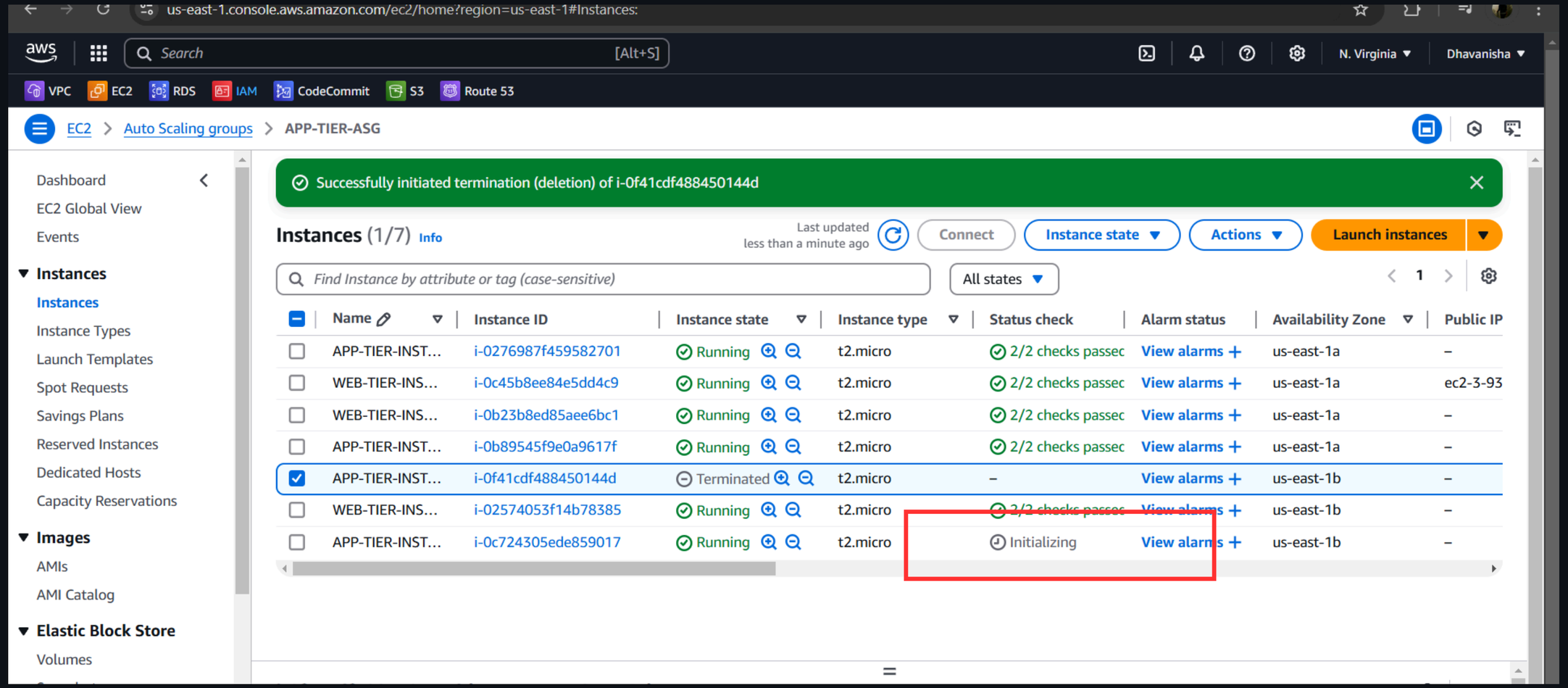
To check outcomes, I am terminating one instance.



I received a sns notification after quitting the instance.



Instances are also created automatically because of the implemented autoscaling.



CONCLUSION

In conclusion, this project successfully implemented a robust and scalable 3-Tier Architecture using AWS services, showcasing the potential of cloud technologies in building modern, efficient, and reliable applications. By utilizing key AWS tools such as VPC, EC2, S3, RDS, IAM, Route 53, Load Balancer, Auto Scaling, Certificate Manager, and CloudWatch, the architecture achieved the following:

- High Scalability: Auto Scaling dynamically adjusted resources to handle varying traffic loads.
- Enhanced Security: IAM and SSL certificates from Certificate Manager ensured secure access and communication.
- Reliability and Fault Tolerance: Load Balancer and Route 53 distributed traffic efficiently and provided redundancy.
- Performance Optimization: CloudWatch enabled real-time monitoring and insights for system improvements.

This architecture demonstrates the power of AWS in providing a flexible and modular framework for deploying multi-tier applications. The project not only addressed key challenges such as integration and scaling but also delivered measurable results, including improved uptime, reduced latency, and cost efficiency.

THANK YOU

dhavanisha.jp@gmail.com