AWS Feedback Management System

Work Integrated Project

Karthikeyan Jeyabalasuntharam

Program: Cloud Computing Technologies (T465)

Group Name: CloudZen

Contents

Project Overview

The AWS Feedback Management System is a multi-tier cloud application designed to collect, process, analyze, and visualize feedback from users in real-time. It is deployed entirely on Amazon Web Services (AWS) using secure, scalable, and highly available architecture.

The system follows a three-tier architecture:

- Frontend (Presentation Layer) – Provides a user interface where users submit feedback through a form.
- Backend (Application Layer) – Processes incoming feedback, stores it in the database, and integrates with AWS Comprehend for sentiment analysis.
- Database (Data Layer) – Stores persistent feedback data, including sentiment analysis.

A key feature of this system is Auto Scaling, which automatically adjusts the number of application and web and app server instances based on incoming traffic. This ensures optimal performance during peak loads while reducing costs during low traffic periods.

In addition to collecting feedback, the system uses AWS Comprehend to detect sentiment (positive, negative, neutral). Monitoring, alerting, and logging are handled by Amazon CloudWatch, ensuring operational visibility and rapid troubleshooting.

Website Link - https://karthikcode.com/

Objectives

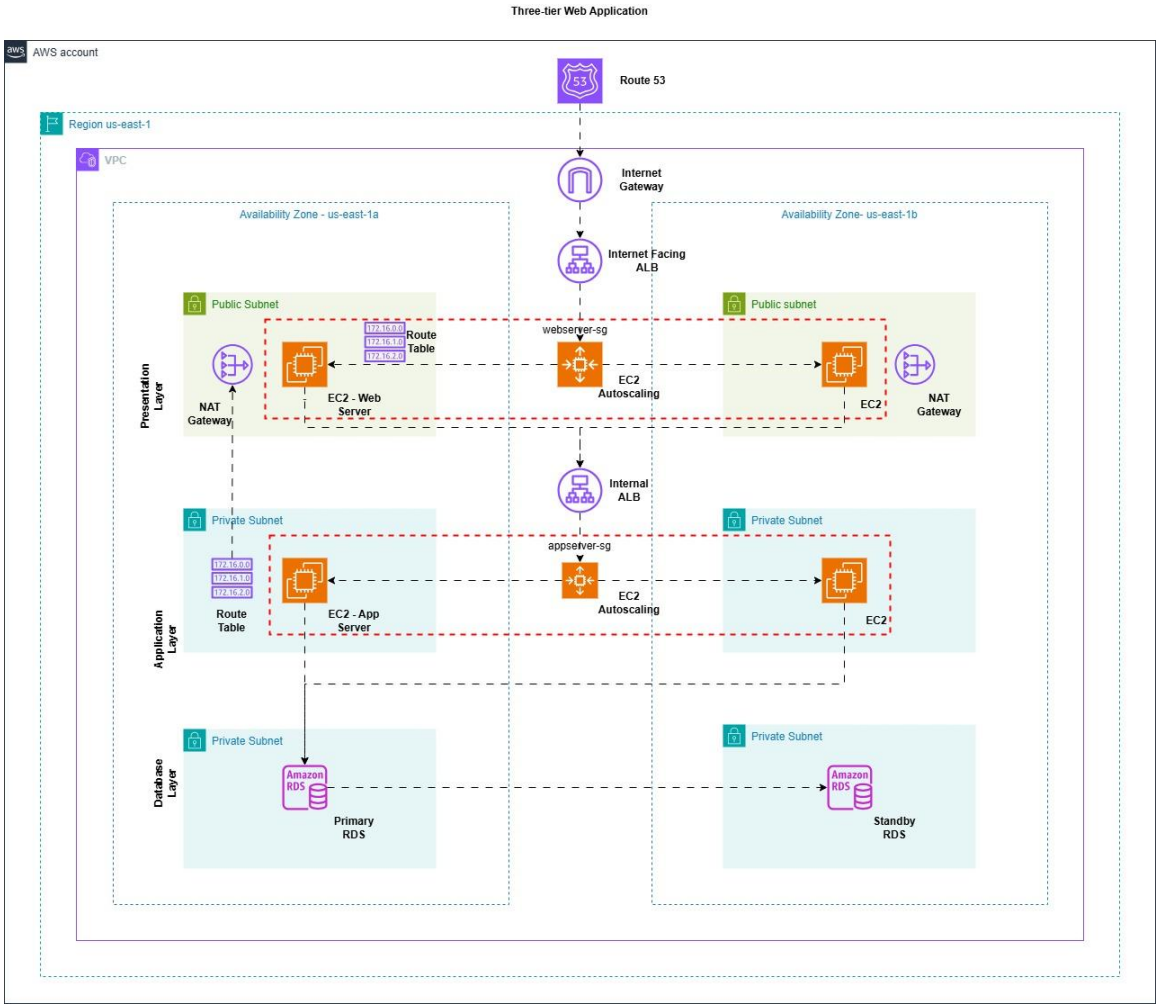
The objectives of the AWS Feedback Management System are:

- Collect User Feedback: Provide a simple and accessible web interface for feedback submission.
- Process and Store Feedback Securely: Ensure data integrity and confidentiality using Amazon RDS.
- Analyze Feedback Sentiment: Integrate AWS Comprehend to automatically classify feedback sentiment.
- Implement Auto Scaling: Dynamically scale EC2 instances for the web and app tiers to match workload demands.
- Maintain High Availability: Deploy resources across multiple Availability Zones with load balancing for fault tolerance.

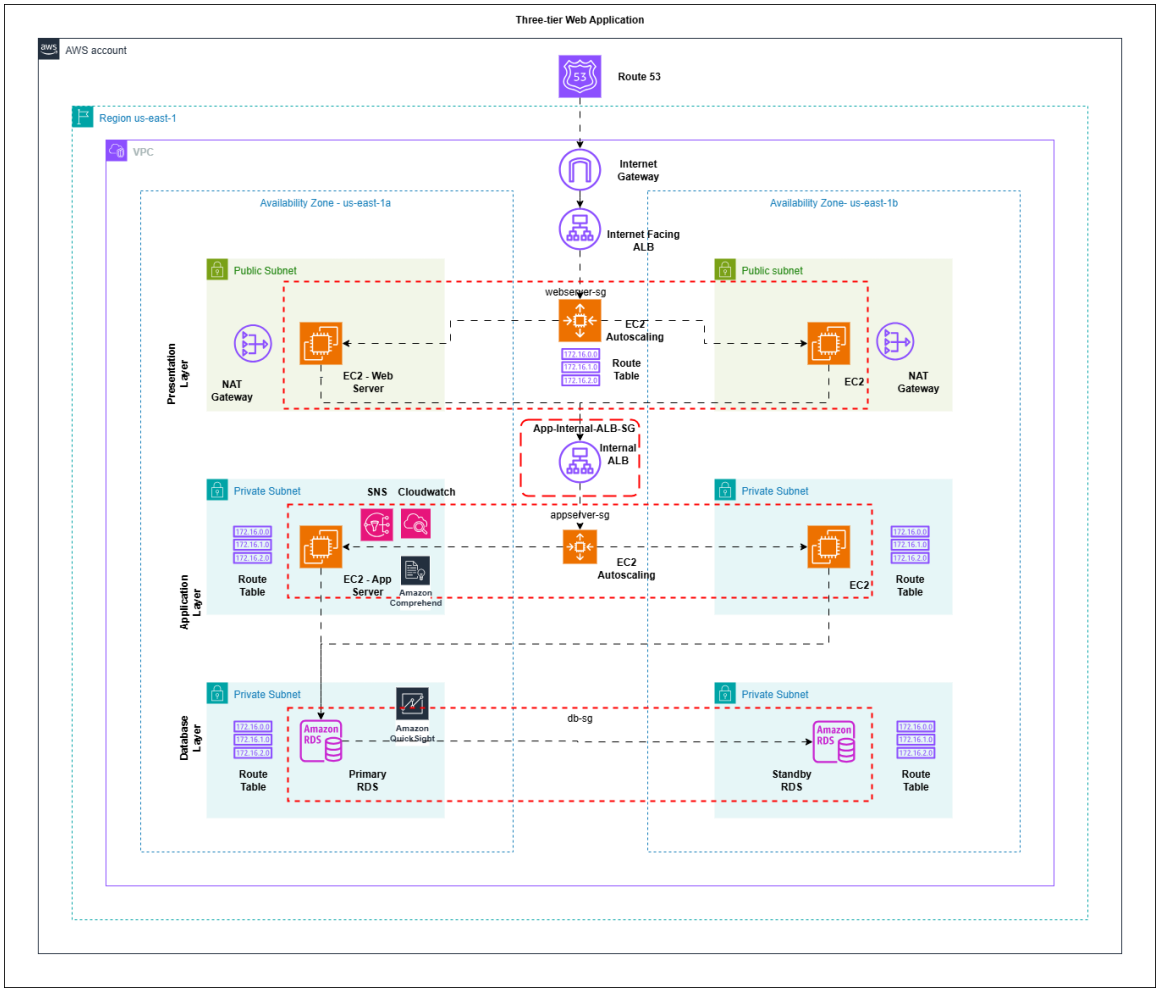
- Enable Monitoring & Alerts**:** Use Amazon CloudWatch to track key performance metrics and trigger alerts on anomalies.
- Demonstrate Scalability: Perform load testing to validate the Auto Scaling setup in real-world-like conditions.

AWS Architecture Diagram

Group Diagram



Individual Diagram



The architecture of the AWS Feedback Management System follows a secure, scalable, and highly available three-tier design:

- Amazon Route 53 – Handles domain name resolution for the feedback system's public endpoint and directs traffic to the Public ALB.
- Internet Gateway (IGW) – Provides internet connectivity to public subnets for the web tier and ALB.
- Public Application Load Balancer (ALB) – Distributes incoming traffic across web-tier EC2 instances in multiple Availability Zones.
- Web Tier (Public Subnets) –
    1. Auto Scaling Group (WebServer-ASG) runs EC2 instances hosting the feedback form frontend.
    2. Security Group (webserver-sg) allows HTTP/HTTPS traffic from anywhere and SSH access only from admin IPs.

- Internal Application Load Balancer (Internal ALB) –
    1. Routes requests from the web tier to the app tier.
    2. Security Group (App-Internal-ALB-SG) allows inbound HTTP only from webserver-sg.
- App Tier (Private Subnets) –
    1. Auto Scaling Group (AppServerASG) runs EC2 instances hosting the backend logic that processes feedback, calls AWS Comprehend, and writes to RDS.
    2. Security Group (appserversg) allows HTTP only from the Internal ALB SG.
- Database Tier (Private Subnets) –
    1. Amazon RDS MySQL for storing data and analysis data.
    2. Security Group (dbsg) allows MySQL access only from appserversg.
- NAT Gateways – Allow private-tier EC2 instances (app and DB layers) to reach the internet for OS updates and package installations without public exposure.
- Route Tables – Separate public and private routes, directing internet traffic via the IGW and private subnet traffic via NAT Gateways.

AWS Resource

The AWS Feedback Management System is deployed using the following AWS services and components:

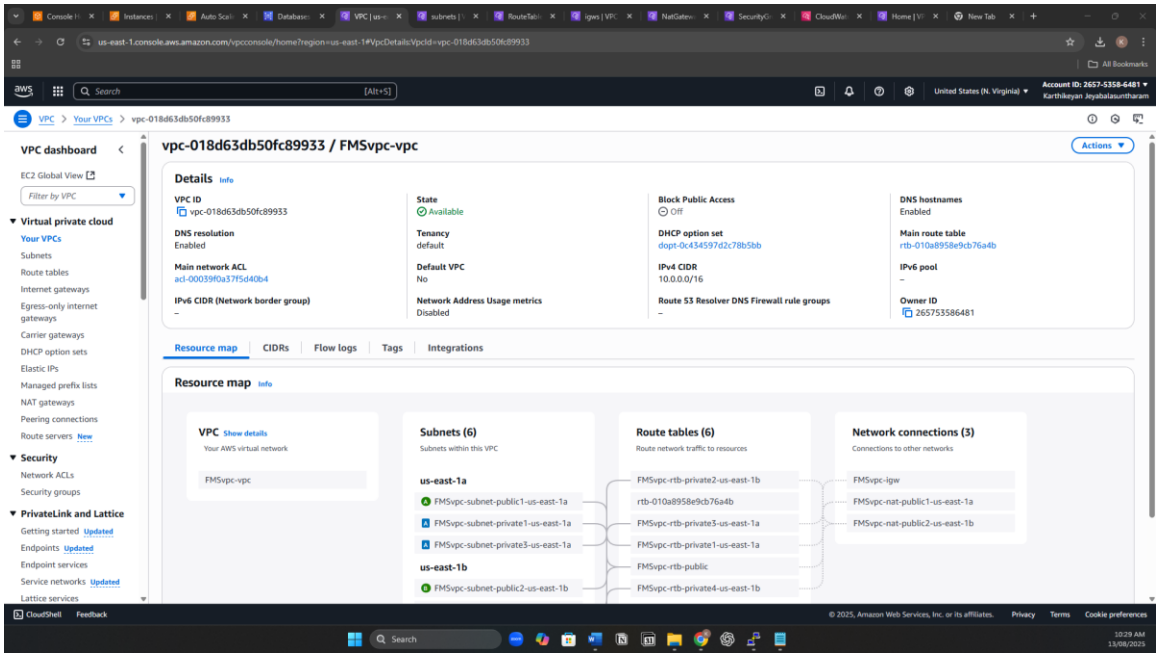
- Amazon VPC – Provides an isolated virtual network for hosting all application components securely.
- Public and Private Subnets – Public subnets host web-tier EC2 instances and load balancers, while private subnets host the application-tier EC2 and RDS.
- Internet Gateway (IGW) – Enables public internet access for frontend servers and public ALB.
- NAT Gateways – Allow private subnets to access the internet for system updates without exposing them.
- Route Tables – Manage traffic flow between subnets, IGW, and NAT Gateways.
- Security Groups – Restrict inbound and outbound traffic for each component (web, app, DB, ALBs) using least-privilege principles.
- Public Application Load Balancer (ALB) – Routes external HTTP traffic to healthy web-tier EC2 instances.

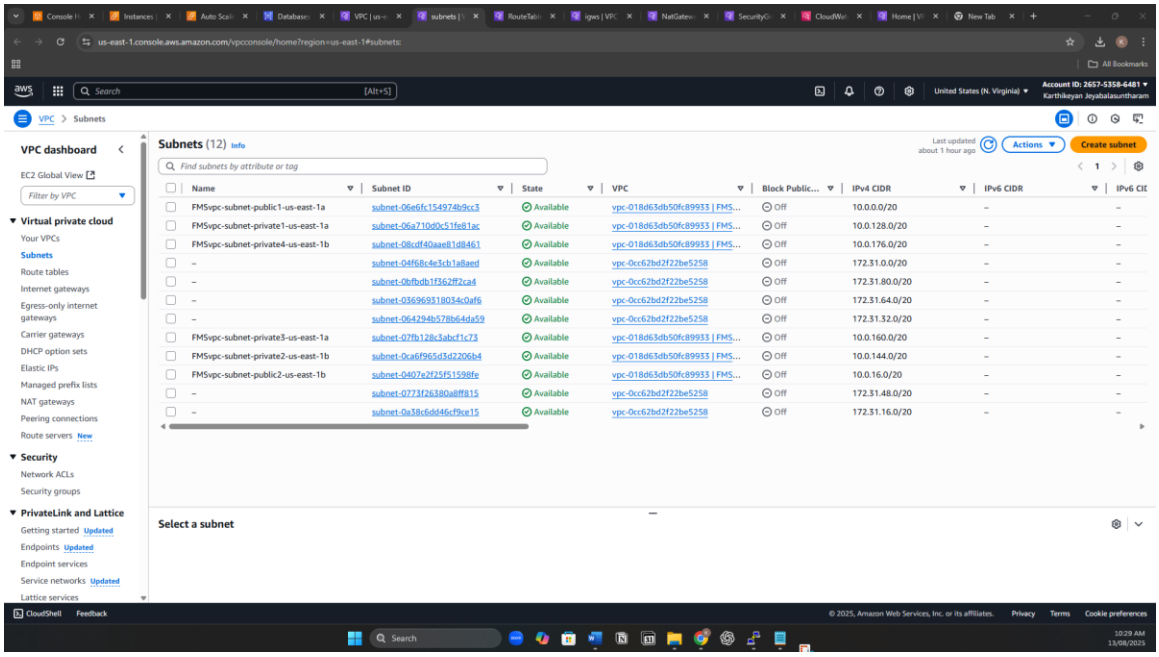
- Internal Application Load Balancer (Internal ALB) – Routes traffic from the web tier to the app tier securely.
- Target Groups – Manage EC2 instance registration for load balancers and perform health checks.
- EC2 Auto Scaling Groups – Dynamically scale EC2 instances for web and app tiers based on CPU utilization and request load.
- Amazon EC2 – Web tier serves the frontend feedback form, and the app tier processes submissions, calls AWS Comprehend, and stores data in RDS.
- Amazon RDS (MySQL) – Managed database storing feedback submissions and sentiment results.
- AWS Comprehend – Analyzes feedback text to determine sentiment (Positive, Negative, Neutral, Mixed).
- Amazon QuickSight – Visualizes sentiment analysis and feedback trends through interactive dashboards.
- Amazon CloudWatch – Monitors application performance, collects logs, and sends alerts based on configured alarms.
- Amazon SNS – Sends notifications when Auto Scaling events occur, such as launching or terminating EC2 instances.

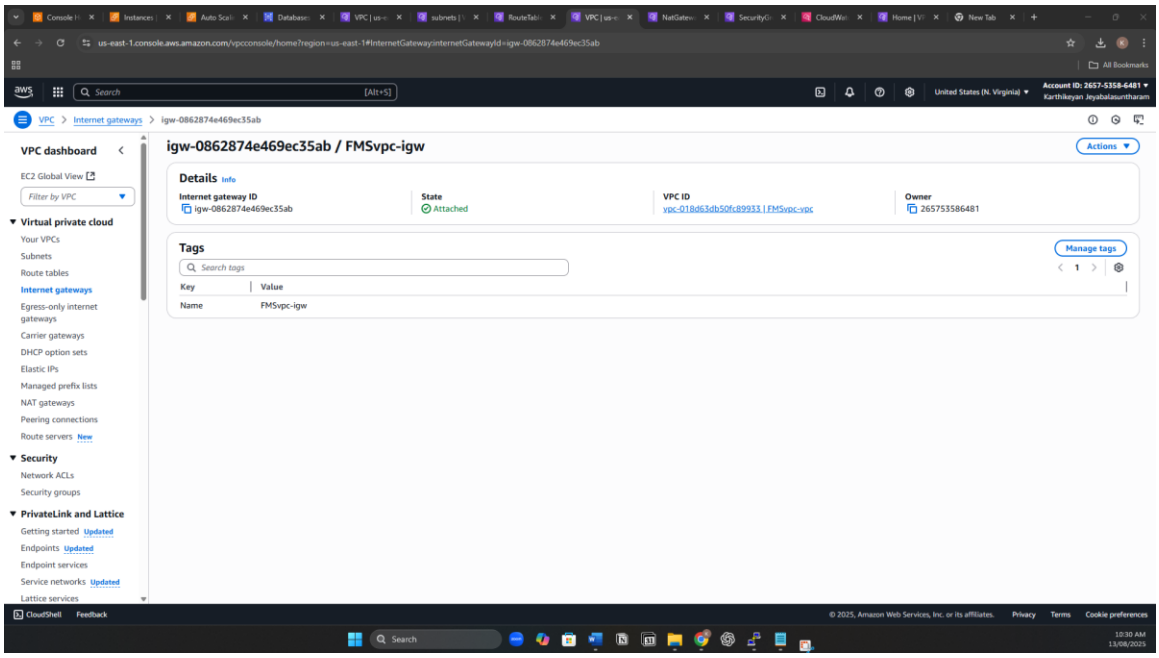
Implementation Steps

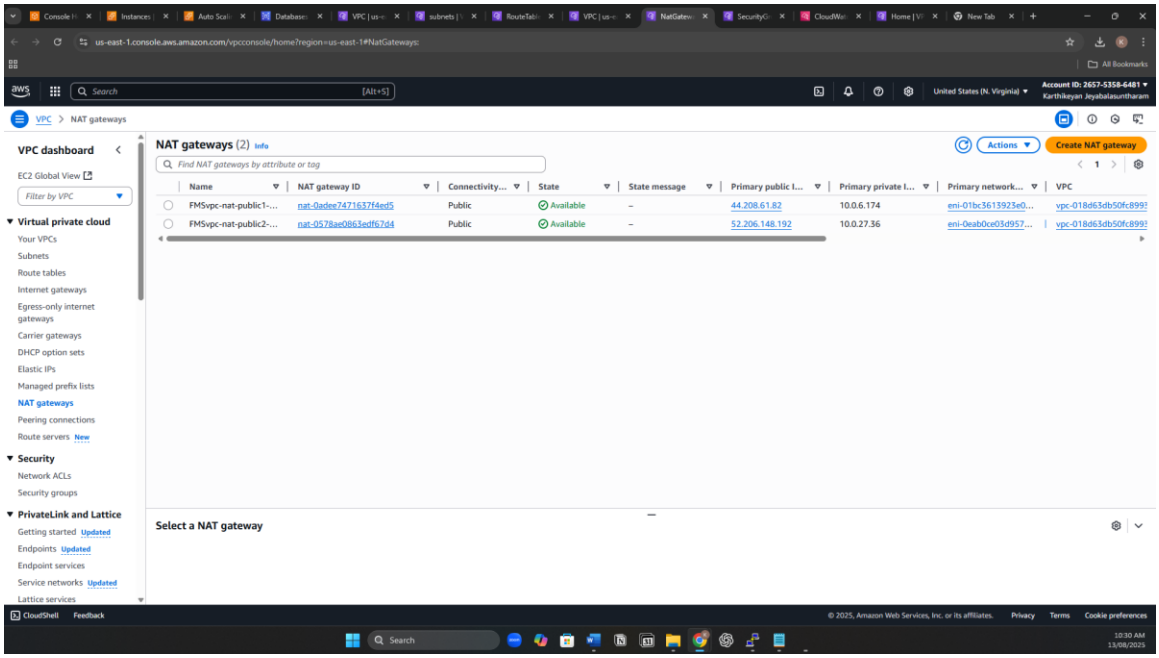
Networking Setup

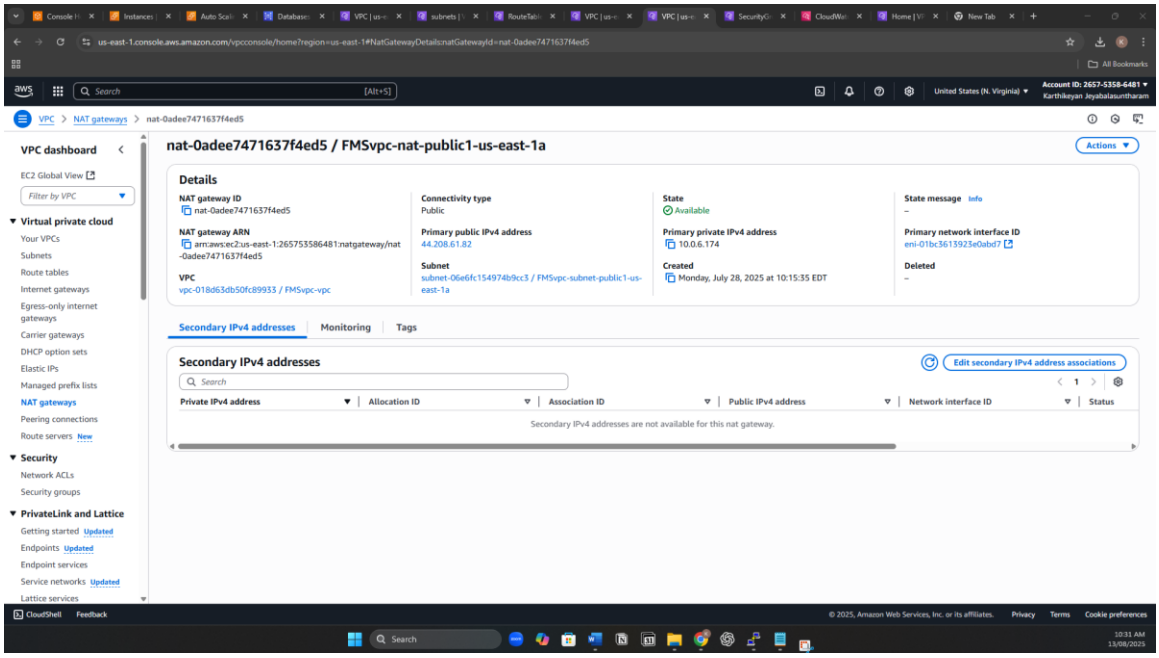
- In the AWS Management Console, open the VPC service and create a VPC with CIDR block 10.0.0.0/16. Enable DNS hostnames and DNS resolution.
- Create 6 subnets:
- Create and attach an Internet Gateway (IGW) to the VPC to allow public internet access for public subnets.
- Create NAT Gateways in each public subnet to provide outbound internet access for private subnets without exposing them publicly.
- Create Route Tables

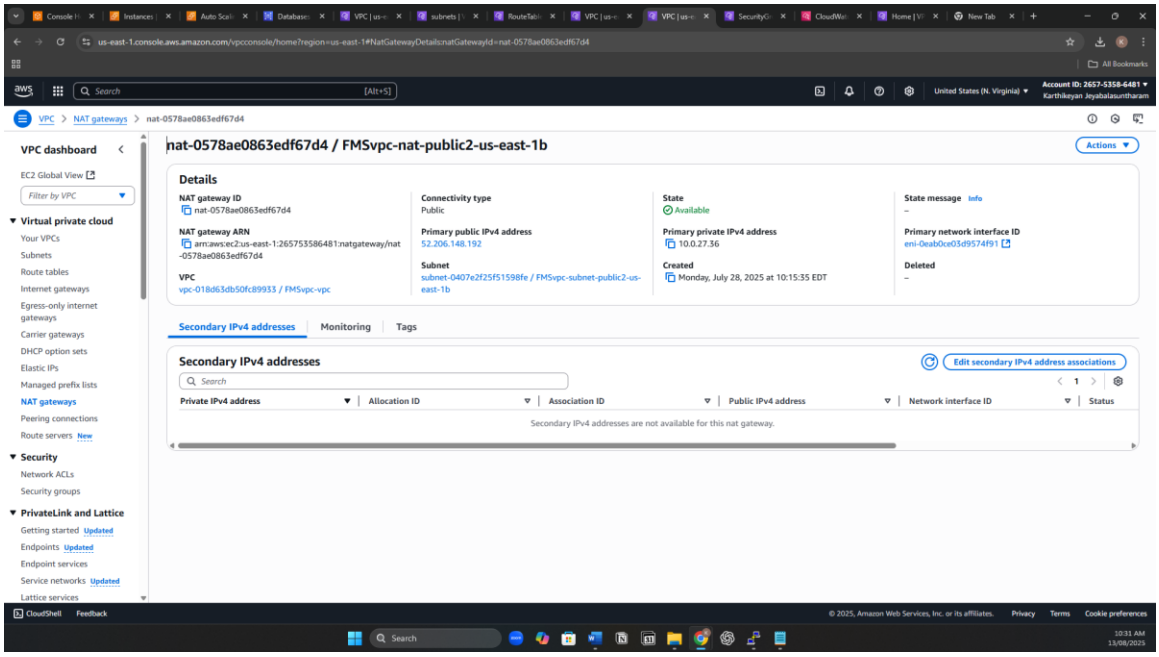

vpc-018d63db50fc89933 / FMSvpc-vpc
Actions
Details Info
VPC ID
vpc-018d63db50fc89933
State
Available
Block Public Access
Off
DNS hostnames
Enabled
DNS resolution
Enabled
Tenancy
default
DHCP option set
dopt-0c434597d2c78b5bb
Main route table
rtb-010a8958e9cb76a4b
Main network ACL
acl-00039f0a37f5d40b4
Default VPC
No
IPv4 CIDR
10.0.0.0/16
IPv6 pool
Network Address Usage metrics
Disabled
Route 53 Resolver DNS Firewall rule groups
Owner ID
265753586481
Resource map
CIDRs
Flow logs
Tags
Integrations
Resource map Info
VPC Show details
Your AWS virtual network
FMSvpc-vpc
Subnets (6)
Subnets within this VPC
us-east-1a
FMSvpc-subnet-public1-us-east-1a
FMSvpc-subnet-private1-us-east-1a
FMSvpc-subnet-private3-us-east-1a
us-east-1b
FMSvpc-subnet-public2-us-east-1b
Route tables (6)
Route network traffic to resources
FMSvpc-rtb-private2-us-east-1b
rtb-010a8958e9cb76a4b
FMSvpc-rtb-private3-us-east-1a
FMSvpc-rtb-private1-us-east-1a
FMSvpc-rtb-public
FMSvpc-rtb-private4-us-east-1b
Network connections (3)
Connections to other networks
FMSvpc-igw
FMSvpc-nat-public1-us-east-1a
FMSvpc-nat-public2-us-east-1b

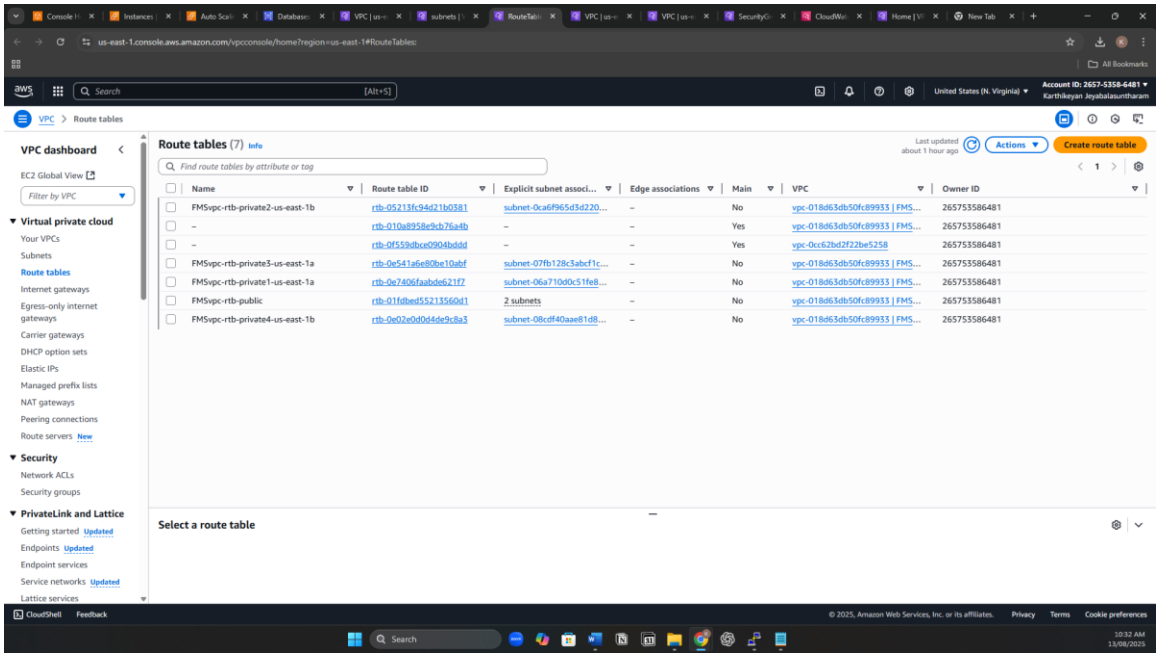

Subnets (12) Info
Actions
Create subnet
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
IPv6 CIDR
FMSvpc-subnet-public1-us-east-1a
subnet-06e6fc154974b9cc3
Available
vpc-018d63db50fc89933 | FMS...
Off
10.0.0.0/20
FMSvpc-subnet-private1-us-east-1a
subnet-06a710d0c51fe81ac
10.0.128.0/20
FMSvpc-subnet-private4-us-east-1b
subnet-08cdf40aae81d8461
10.0.176.0/20
subnet-04f68c4e3cb1a8aed
vpc-0cc62bd2f22be5258
172.31.0.0/20
subnet-0bfbdb1f362ff2ca4
172.31.80.0/20
subnet-036969318034c0af6
172.31.64.0/20
subnet-064294b578b64da59
172.31.32.0/20
FMSvpc-subnet-private3-us-east-1a
subnet-07fb128c3abcf1c73
10.0.160.0/20
FMSvpc-subnet-private2-us-east-1b
subnet-0ca6f965d3d2206b4
10.0.144.0/20
FMSvpc-subnet-public2-us-east-1b
subnet-0407e2f25f51598fe
10.0.16.0/20
subnet-0773f26380a8fff815
172.31.48.0/20
subnet-0a38c6dd46cf9ce15
172.31.16.0/20
Select a subnet


us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#InternetGateway:internetGatewayId=igw-0862874e469ec35ab
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
VPC > Internet gateways > igw-0862874e469ec35ab
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
Route servers New
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Endpoint services
Service networks Updated
Lattice services
igw-0862874e469ec35ab / FMSvpc-igw
Actions
Details Info
Internet gateway ID
igw-0862874e469ec35ab
State
Attached
VPC ID
vpc-018d63db50fc89933 | FMSvpc-vpc
Owner
265753586481
Tags
Search tags
Manage tags
Key
Value
Name
FMSvpc-igw
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
10:30 AM
13/08/2025


us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#NatGateways:
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
VPC > NAT gateways
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
Route servers New
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started Updated
Endpoints Updated
Endpoint services
Service networks Updated
Lattice services
NAT gateways (2) Info
Find NAT gateways by attribute or tag
Actions
Create NAT gateway
Name
NAT gateway ID
Connectivity...
State
State message
Primary public I...
Primary private I...
Primary network...
VPC
FMSvpc-nat-public1-...
nat-0adee7471637f4ed5
Public
Available
–
44.208.61.82
10.0.6.174
eni-01bc3613923e0...
vpc-018d63db50fc899
FMSvpc-nat-public2-...
nat-0578ae0863edf67d4
Public
Available
–
52.206.148.192
10.0.27.36
eni-0eab0ce03d957...
vpc-018d63db50fc899
Select a NAT gateway
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
10:30 AM
13/08/2025


us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#NatGatewayDetails:natGatewayId=nat-0adee7471637f4ed5
VPC > NAT gateways > nat-0adee7471637f4ed5
nat-0adee7471637f4ed5 / FMSvpc-nat-public1-us-east-1a
Actions
Details
NAT gateway ID
nat-0adee7471637f4ed5
NAT gateway ARN
arn:aws:ec2:us-east-1:265753586481:natgateway/nat-0adee7471637f4ed5
VPC
vpc-018d63db50fc89933 / FMSvpc-vpc
Connectivity type
Public
Primary public IPv4 address
44.208.61.82
Subnet
subnet-06e6fc154974b9cc3 / FMSvpc-subnet-public1-us-east-1a
State
Available
Primary private IPv4 address
10.0.6.174
Created
Monday, July 28, 2025 at 10:15:35 EDT
State message
Primary network interface ID
eni-01bc3613923e0abd7
Deleted
Secondary IPv4 addresses
Monitoring
Tags
Secondary IPv4 addresses
Edit secondary IPv4 address associations
Private IPv4 address
Allocation ID
Association ID
Public IPv4 address
Network interface ID
Status
Secondary IPv4 addresses are not available for this nat gateway.
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
Route servers
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started
Endpoints
Endpoint services
Service networks
Lattice services
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

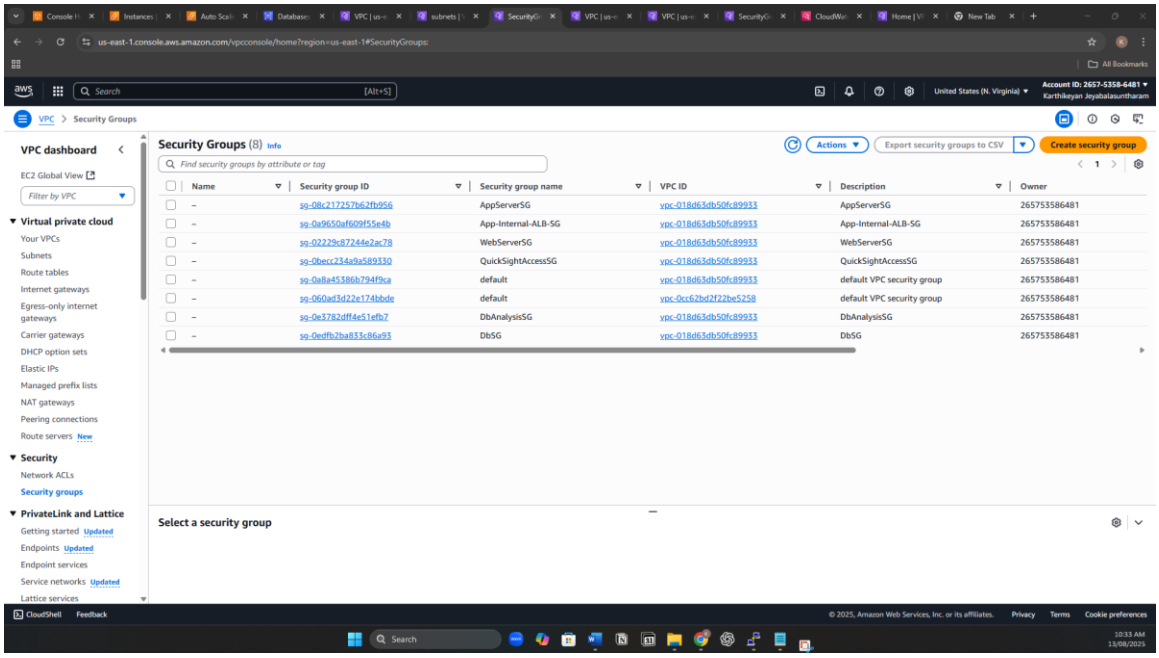

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#NatGatewayDetails:natGatewayId=nat-0578ae0863edf67d4
VPC > NAT gateways > nat-0578ae0863edf67d4
nat-0578ae0863edf67d4 / FMSvpc-nat-public2-us-east-1b
Actions
Details
NAT gateway ID
nat-0578ae0863edf67d4
NAT gateway ARN
arn:aws:ec2:us-east-1:265753586481:natgateway/nat-0578ae0863edf67d4
VPC
vpc-018d63db50fc89933 / FMSvpc-vpc
Connectivity type
Public
Primary public IPv4 address
52.206.148.192
Subnet
subnet-0407e2f25f51598fe / FMSvpc-subnet-public2-us-east-1b
State
Available
Primary private IPv4 address
10.0.27.36
Created
Monday, July 28, 2025 at 10:15:35 EDT
State message
Primary network interface ID
eni-0eab0ce03d9574f91
Deleted
Secondary IPv4 addresses
Monitoring
Tags
Secondary IPv4 addresses
Edit secondary IPv4 address associations
Private IPv4 address
Allocation ID
Association ID
Public IPv4 address
Network interface ID
Status
Secondary IPv4 addresses are not available for this nat gateway.
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
Route servers
Security
Network ACLs
Security groups
PrivateLink and Lattice
Getting started
Endpoints
Endpoint services
Service networks
Lattice services
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

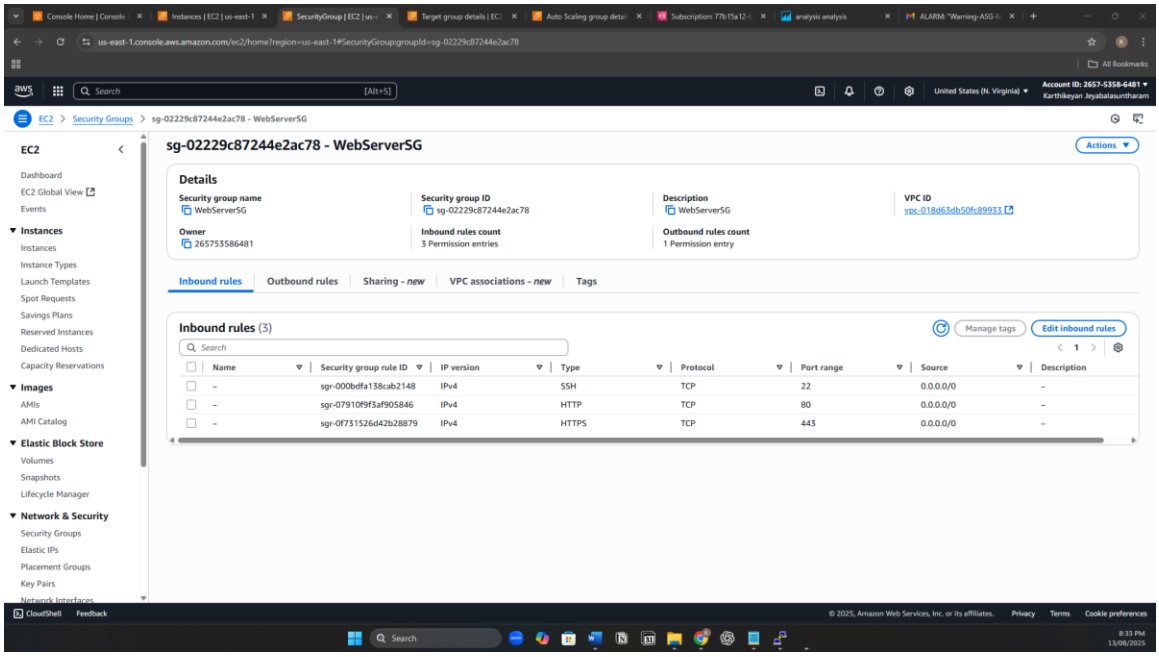


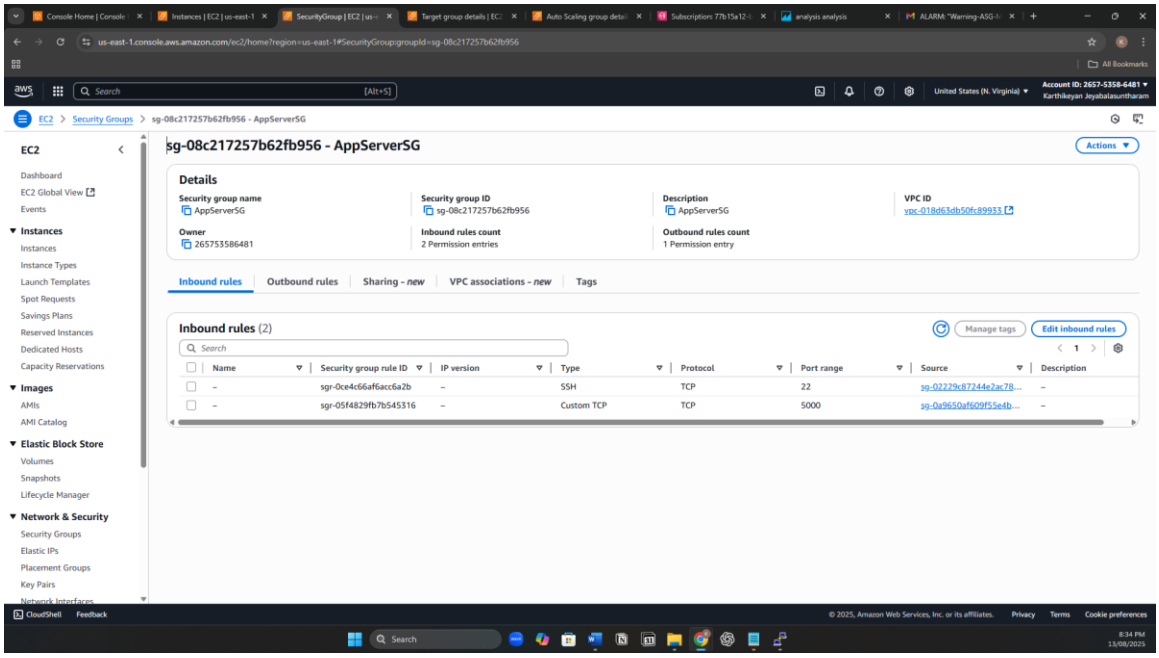
Security Configuration

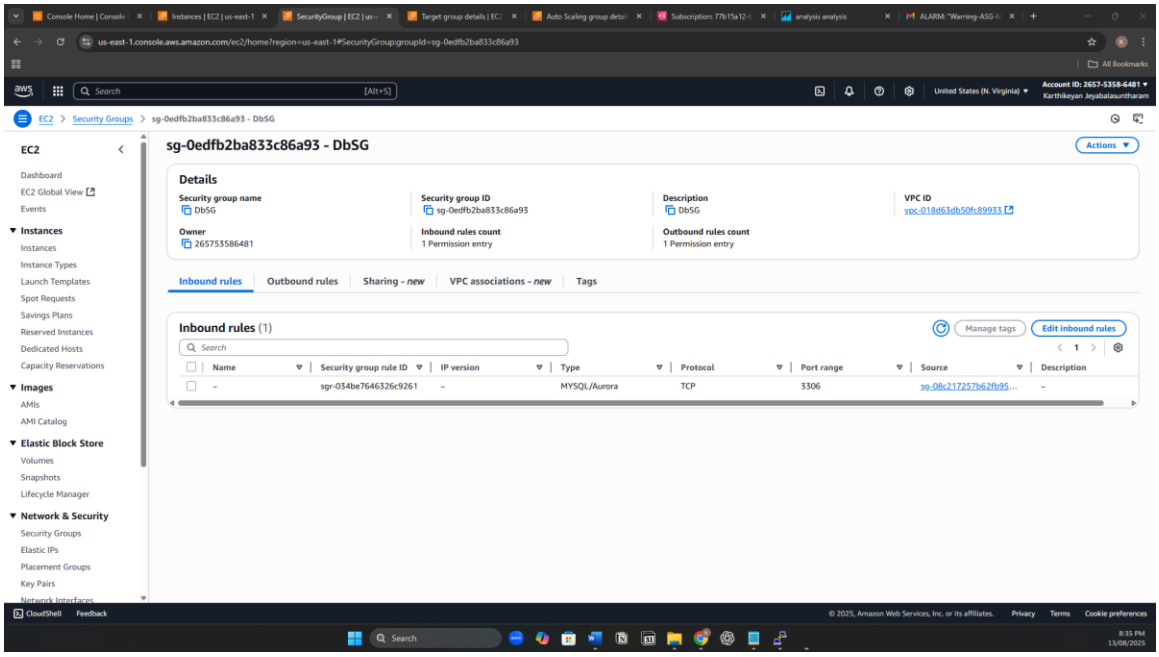
Security Groups

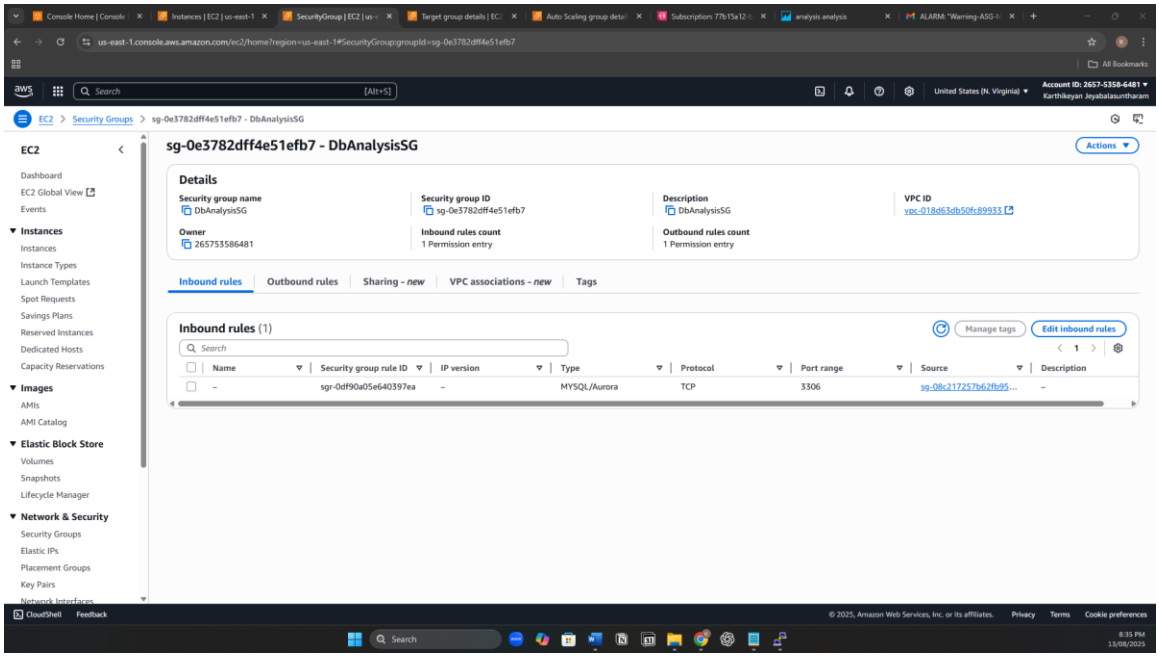
- Create WebServerSG:
  - Inbound: Allow HTTP (80), SSH (22) and HTTPS (443) from 0.0.0.0/0.
  - Outbound: Allow all traffic (default).
- Create AppServerSG:
  - Inbound: Allow SSH (22) and Custom TCP(5000) from WebServerSG.
  - Outbound: Allow all traffic (default).
- Create DBSG, DbAnalysisSG:
  - Inbound: Allow MySQL (3306) from AppServerSG.
  - Outbound: Allow all traffic (default).
- Create ALB security group:
  - Internal ALB-SG: Allow HTTP(80) from WebServerSG only.

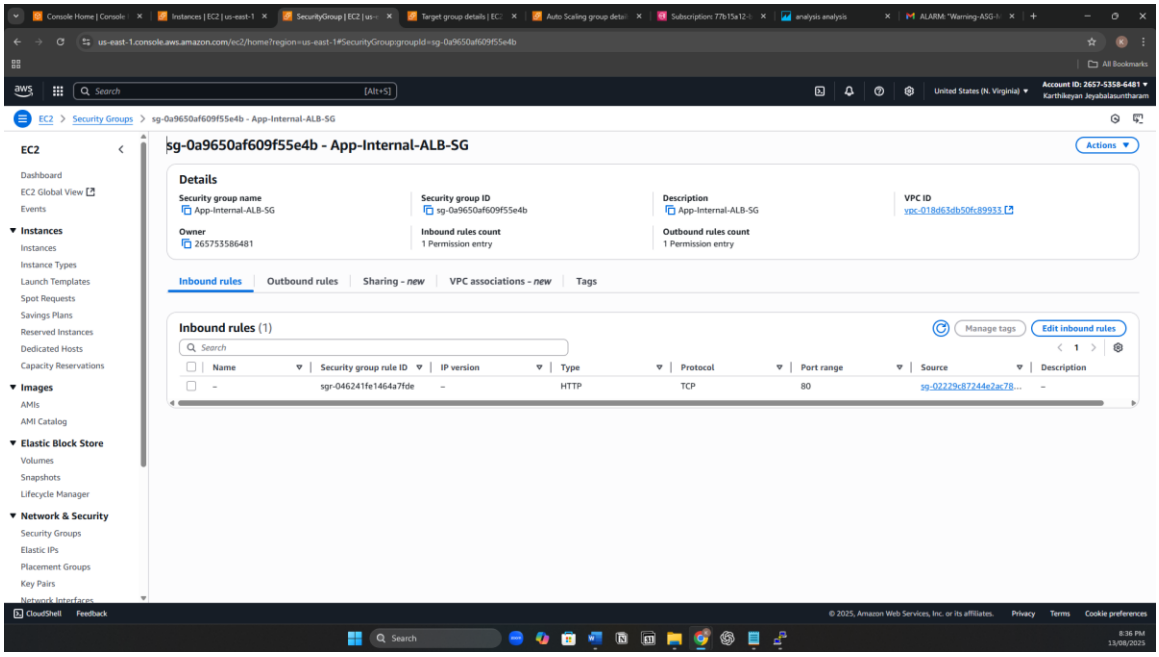

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#SecurityGroups:
VPC > Security Groups
VPC dashboard
Security Groups (8) Info
Actions
Export security groups to CSV
Create security group
Find security groups by attribute or tag
Name
Security group ID
Security group name
VPC ID
Description
Owner
sg-08c217257b62fb956 AppServerSG vpc-018d63db50fc89933 AppServerSG 265753586481
sg-0a9650af609f55e4b App-Internal-ALB-SG vpc-018d63db50fc89933 App-Internal-ALB-SG 265753586481
sg-02229c87244e2ac78 WebServerSG vpc-018d63db50fc89933 WebServerSG 265753586481
sg-0becc234a9a589330 QuickSightAccessSG vpc-018d63db50fc89933 QuickSightAccessSG 265753586481
sg-0a8a45386b794f9ca default vpc-018d63db50fc89933 default VPC security group 265753586481
sg-060ad3d22e174bbde default vpc-0cc62bd2f22be5258 default VPC security group 265753586481
sg-0e3782dff4e51efb7 DbAnalysisSG vpc-018d63db50fc89933 DbAnalysisSG 265753586481
sg-0edfb2ba833c86a93 DbSG vpc-018d63db50fc89933 DbSG 265753586481
Select a security group
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroup:groupId=sg-02229c87244e2ac78
EC2 > Security Groups > sg-02229c87244e2ac78 - WebServerSG
sg-02229c87244e2ac78 - WebServerSG
Actions
Details
Security group name
WebServerSG
Security group ID
sg-02229c87244e2ac78
Description
WebServerSG
VPC ID
vpc-018d63db50fc89933
Owner
265753586481
Inbound rules count
3 Permission entries
Outbound rules count
1 Permission entry
Inbound rules
Outbound rules
Sharing - new
VPC associations - new
Tags
Inbound rules (3)
Manage tags
Edit inbound rules
Name
Security group rule ID
IP version
Type
Protocol
Port range
Source
Description
sgr-000bdfa138cab2148 IPv4 SSH TCP 22 0.0.0.0/0
sgr-07910f9f3af905846 IPv4 HTTP TCP 80 0.0.0.0/0
sgr-0f731526d42b28879 IPv4 HTTPS TCP 443 0.0.0.0/0
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


sg-08c217257b62fb956 - AppServerSG
Actions
Details
Security group name
AppServerSG
Security group ID
sg-08c217257b62fb956
Description
AppServerSG
VPC ID
vpc-018d63db50fc89933
Owner
265753586481
Inbound rules count
2 Permission entries
Outbound rules count
1 Permission entry
Inbound rules
Outbound rules
Sharing - new
VPC associations - new
Tags
Inbound rules (2)
Manage tags
Edit inbound rules
Name
Security group rule ID
IP version
Type
Protocol
Port range
Source
Description
sgr-0ce4c66af6acc6a2b
SSH
TCP
22
sg-02229c8724e2ac78...
sgr-05f4829fb7b545316
Custom TCP
TCP
5000
sg-0a9650af609f55e4b...


sg-0edfb2ba833c86a93 - DbSG
Actions
Details
Security group name
DbSG
Security group ID
sg-0edfb2ba833c86a93
Description
DbSG
VPC ID
vpc-018d63db50fc89933
Owner
265753586481
Inbound rules count
1 Permission entry
Outbound rules count
1 Permission entry
Inbound rules
Outbound rules
Sharing - new
VPC associations - new
Tags
Inbound rules (1)
Manage tags
Edit inbound rules
Name
Security group rule ID
IP version
Type
Protocol
Port range
Source
Description
sgr-034be7646326c9261
MYSQL/Aurora
TCP
3306
sg-08c217257b62fb95...

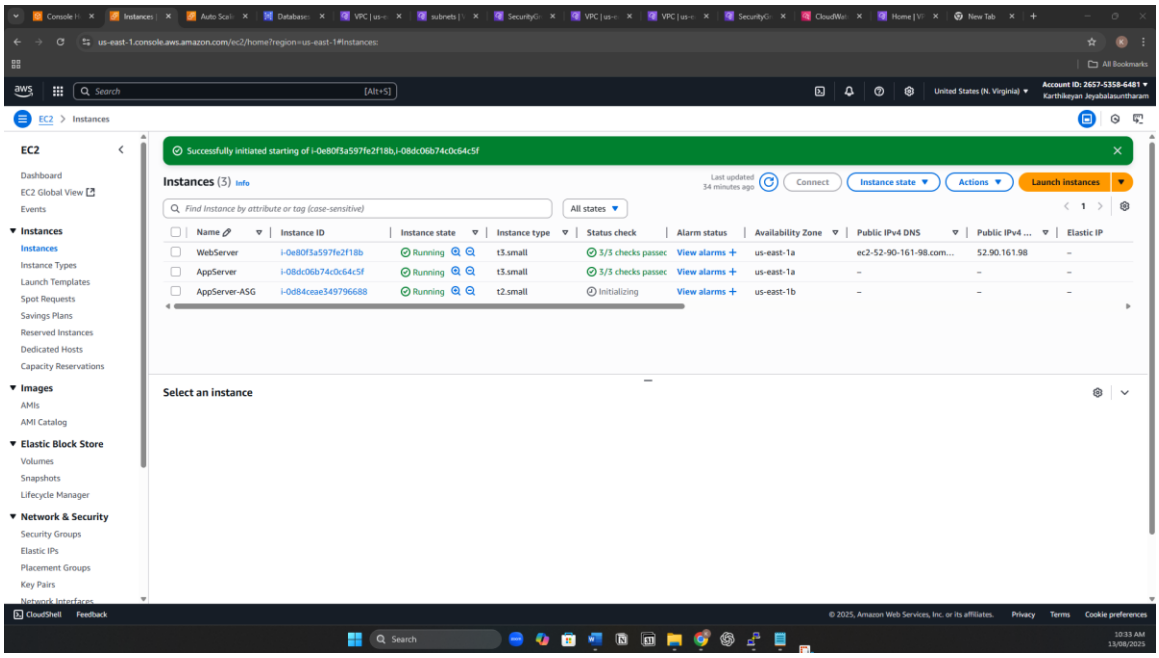


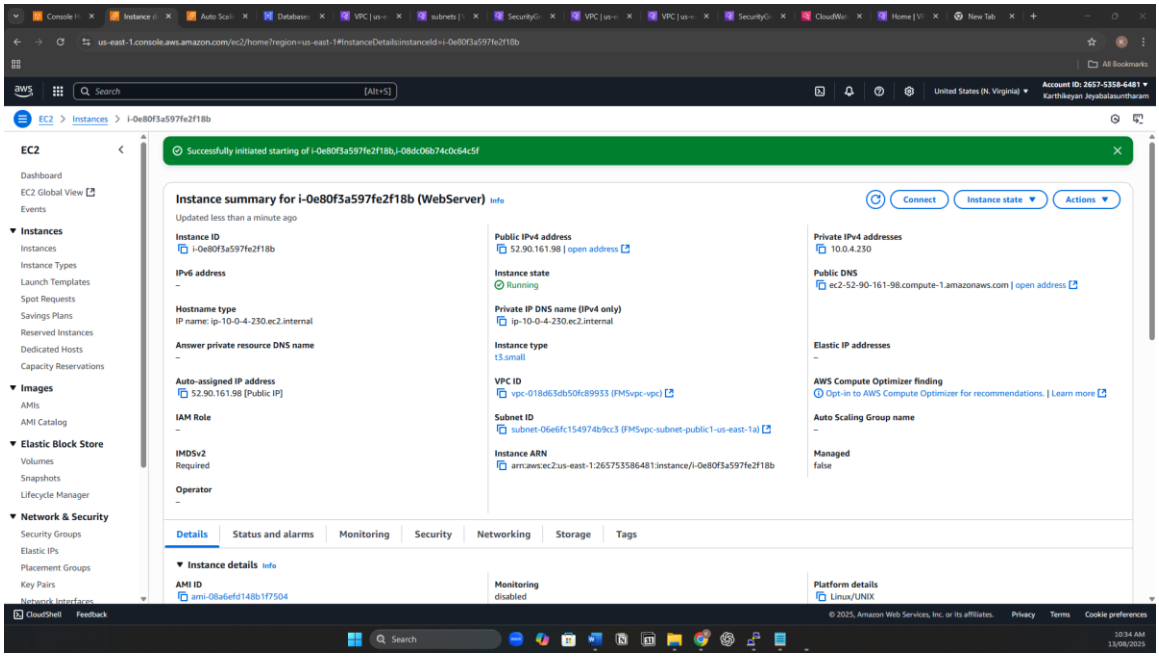


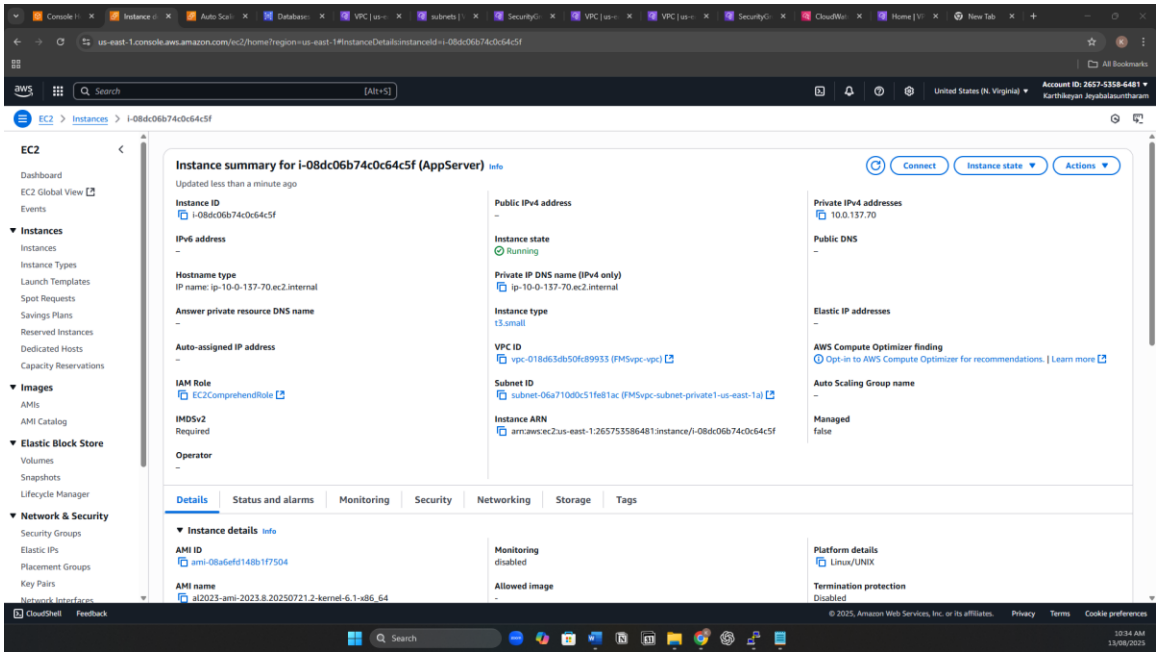
Web & App Deployment

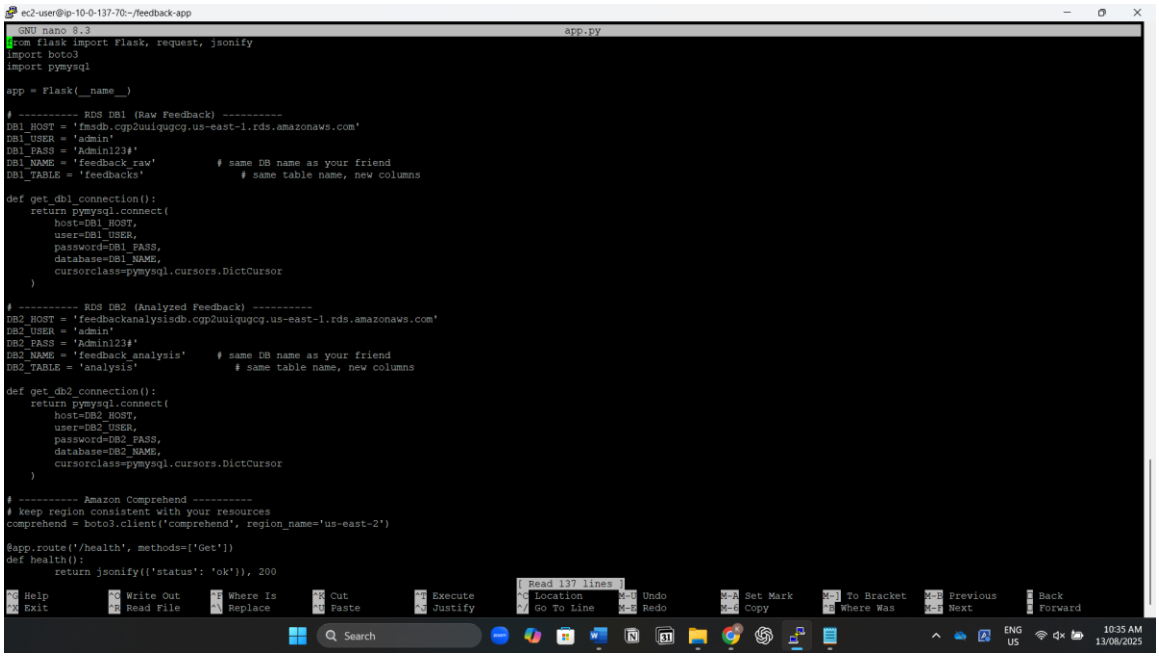
- Launch EC2 instances in the Web Tier (public subnets) using Amazon Linux 2023. Assign WebServer-SG.
- Install Apache web server and configure it to serve the frontend feedback form.

- Launch EC2 instances in the App Tier (private subnets) using Amazon Linux 2023. Assign AppServer-SG.
- Install Python3, Flask, MySQL-Connector, and Boto3 for AWS Comprehend integration.
- Deploy the feedback application code to /home/ec2-user/feedback-app.
- Create feedback-app.service systemd unit file to start the app automatically on boot
- Enable and start the service:



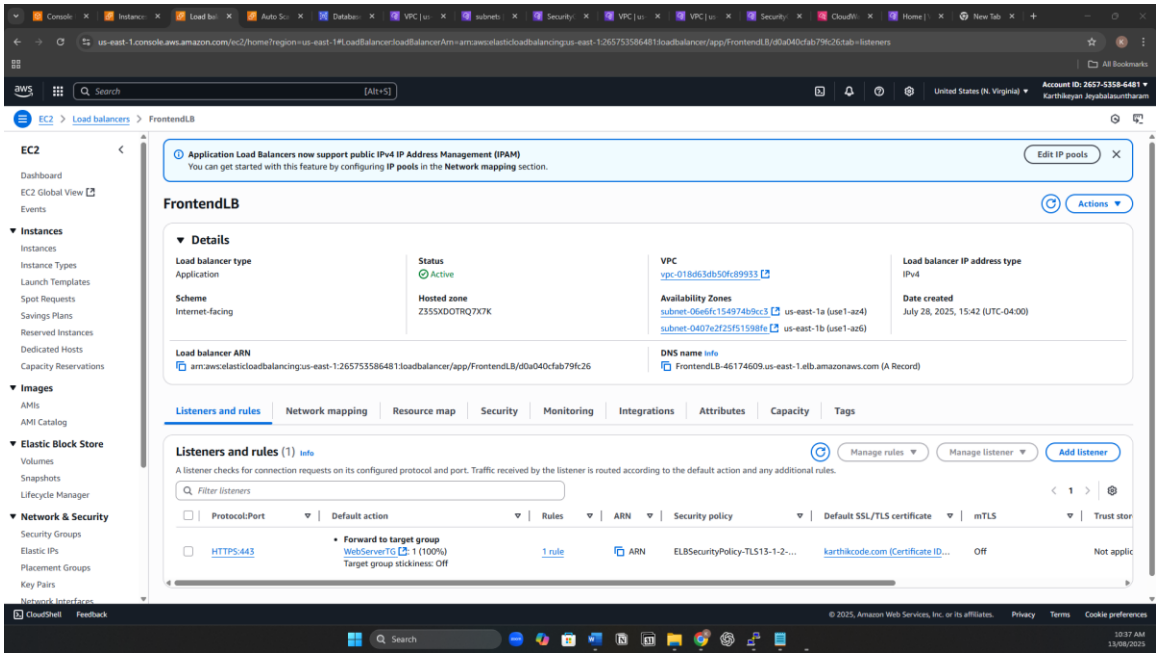

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-0e80f3a597fe2f18b
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
EC2 > Instances > i-0e80f3a597fe2f18b
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
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
Placement Groups
Key Pairs
Network Interfaces
Successfully initiated starting of i-0e80f3a597fe2f18b,i-08dc06b74c0c64c5f
Instance summary for i-0e80f3a597fe2f18b (WebServer) Info
Updated less than a minute ago
Connect
Instance state
Actions
Instance ID
i-0e80f3a597fe2f18b
IPv6 address
–
Hostname type
IP name: ip-10-0-4-230.ec2.internal
Answer private resource DNS name
–
Auto-assigned IP address
52.90.161.98 [Public IP]
IAM Role
–
IMDSv2
Required
Operator
–
Public IPv4 address
52.90.161.98 | open address
Instance state
Running
Private IP DNS name (IPv4 only)
ip-10-0-4-230.ec2.internal
Instance type
t3.small
VPC ID
vpc-018d63db50fc89933 (FMSvpc-vpc)
Subnet ID
subnet-06e6fc154974b9cc3 (FMSvpc-subnet-public1-us-east-1a)
Instance ARN
arn:aws:ec2:us-east-1:265753586481:instance/i-0e80f3a597fe2f18b
Private IPv4 addresses
10.0.4.230
Public DNS
ec2-52-90-161-98.compute-1.amazonaws.com | open address
Elastic IP addresses
–
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more
Auto Scaling Group name
–
Managed
false
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance details Info
AMI ID
ami-08a6efd148b1f7504
Monitoring
disabled
Platform details
Linux/UNIX
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
10:34 AM
13/08/2025

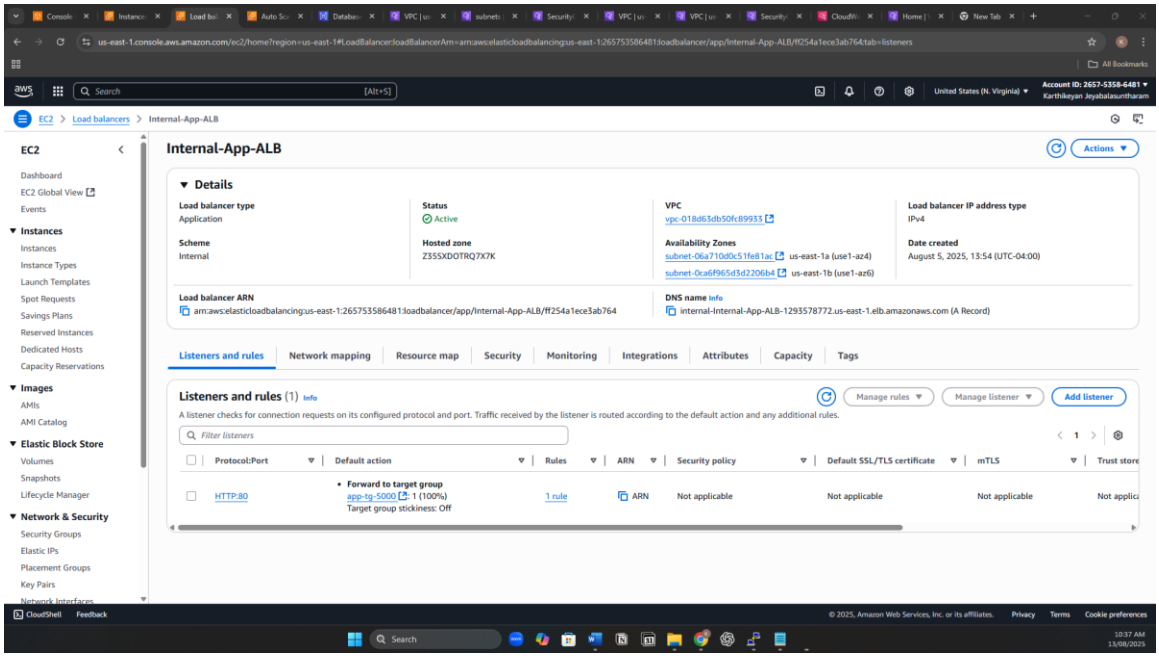

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-08dc06b74c0c64c5f
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
EC2 > Instances > i-08dc06b74c0c64c5f
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
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
Placement Groups
Key Pairs
Network Interfaces
Instance summary for i-08dc06b74c0c64c5f (AppServer) Info
Updated less than a minute ago
Connect
Instance state
Actions
Instance ID
i-08dc06b74c0c64c5f
IPv6 address
–
Hostname type
IP name: ip-10-0-137-70.ec2.internal
Answer private resource DNS name
–
Auto-assigned IP address
–
IAM Role
EC2ComprehendRole
IMDSv2
Required
Operator
–
Public IPv4 address
–
Instance state
Running
Private IP DNS name (IPv4 only)
ip-10-0-137-70.ec2.internal
Instance type
t3.small
VPC ID
vpc-018d63db50fc89933 (FMSvpc-vpc)
Subnet ID
subnet-06a710d0c51fe81ac (FMSvpc-subnet-private1-us-east-1a)
Instance ARN
arn:aws:ec2:us-east-1:265753586481:instance/i-08dc06b74c0c64c5f
Private IPv4 addresses
10.0.137.70
Public DNS
–
Elastic IP addresses
–
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more
Auto Scaling Group name
–
Managed
false
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance details Info
AMI ID
ami-08a6efd148b1f7504
Monitoring
disabled
Platform details
Linux/UNIX
AMI name
al2023-ami-2023.8.20250721.2-kernel-6.1-x86_64
Allowed image
-
Termination protection
Disabled
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
10:34 AM
13/08/2025

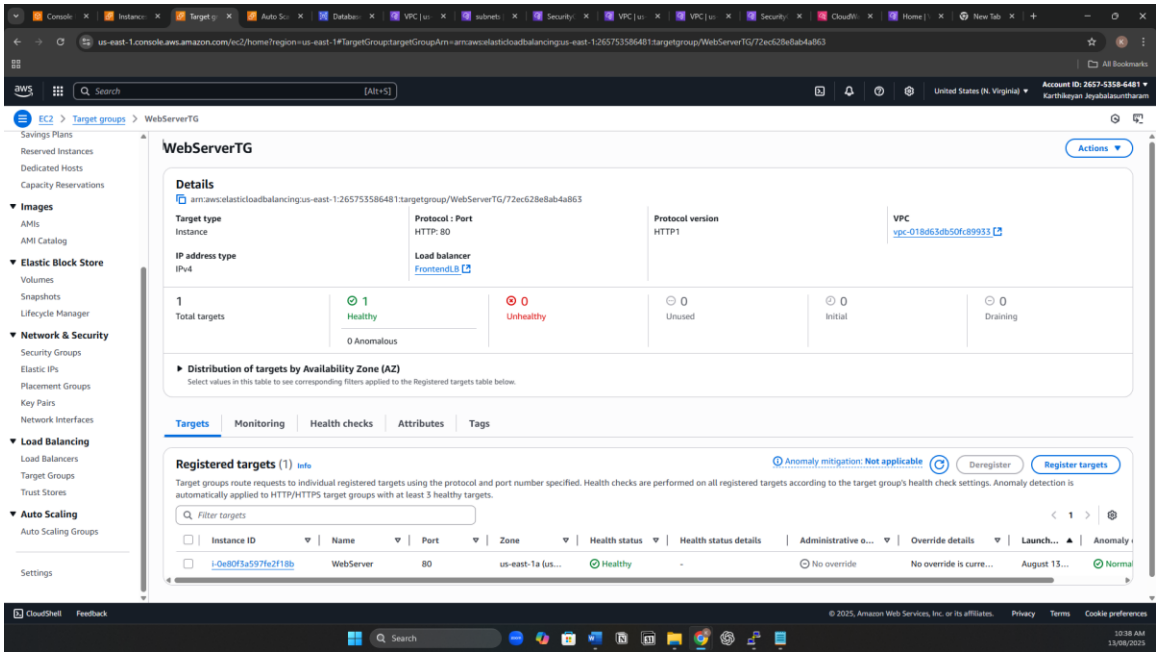


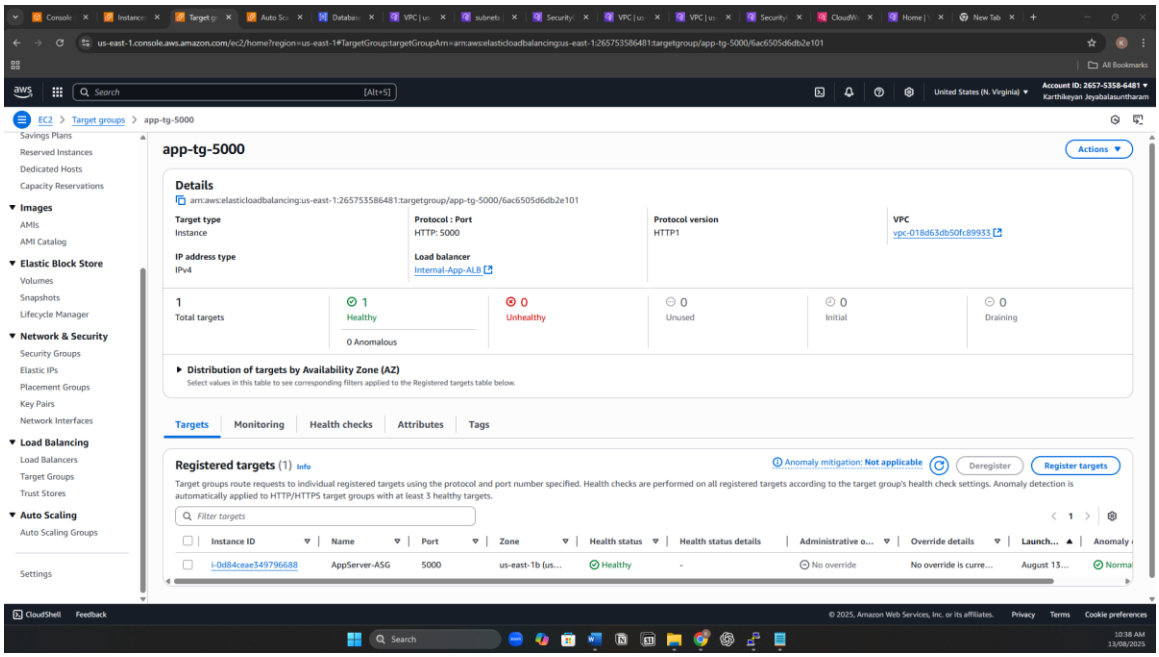
Load Balancing

- Create a Public Application Load Balancer for Web Tier.
- Create an Internal Application Load Balancer App Tier.
- Register target groups.



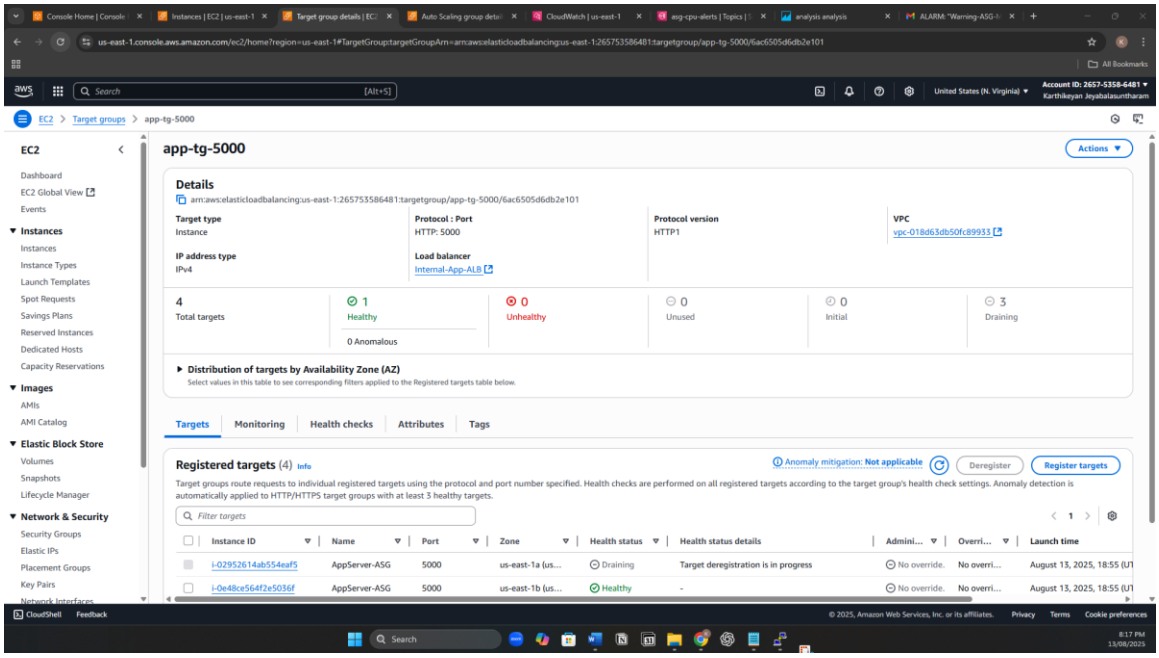

Internal-App-ALB
Actions
Details
Load balancer type
Application
Status
Active
VPC
vpc-018d63db50fc89933
Load balancer IP address type
IPv4
Scheme
Internal
Hosted zone
Z35SXDOTRQ7X7K
Availability Zones
subnet-06a710d0c51fe81ac us-east-1a (use1-az4)
subnet-0ca6f965d3d2206b4 us-east-1b (use1-az6)
Date created
August 5, 2025, 13:54 (UTC-04:00)
Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:265753586481:loadbalancer/app/Internal-App-ALB/ff254a1ece3ab764
DNS name Info
internal-Internal-App-ALB-1293578772.us-east-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map
Security
Monitoring
Integrations
Attributes
Capacity
Tags
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
Protocol:Port
Default action
Rules
ARN
Security policy
Default SSL/TLS certificate
mTLS
Trust store
HTTP:80
Forward to target group
app-tg-5000: 1 (100%)
Target group stickiness: Off
1 rule
ARN
Not applicable
Not applicable
Not applicable
Not applic

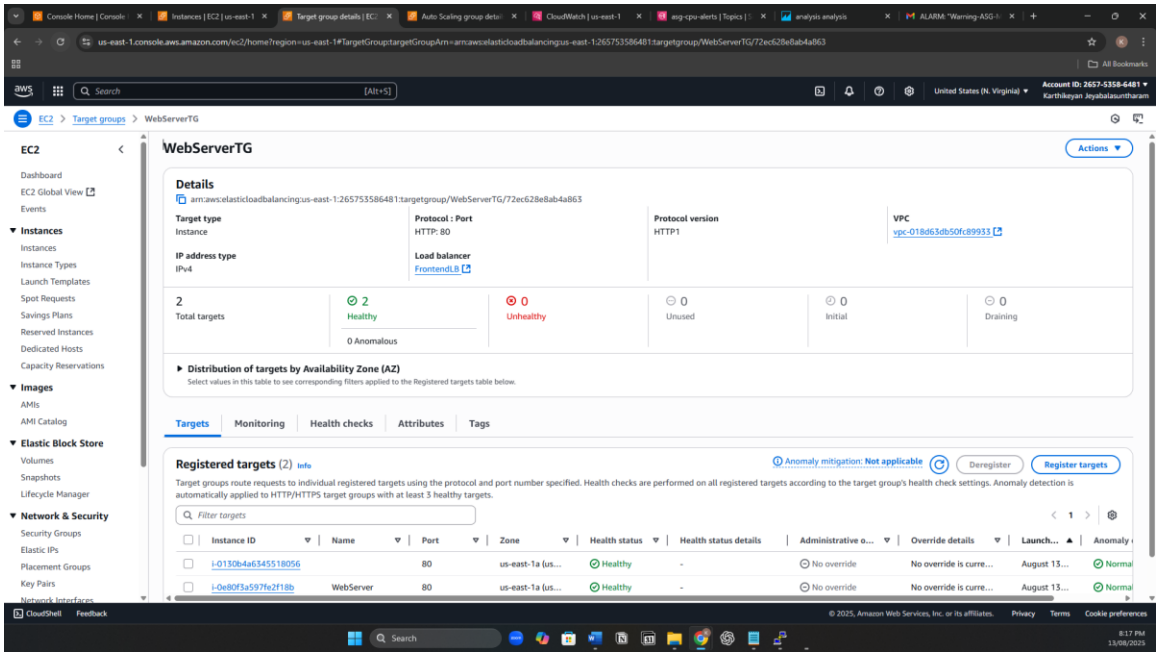

WebServerTG
Actions
Details
arn:aws:elasticloadbalancing:us-east-1:265753586481:targetgroup/WebServerTG/72ec628e8ab4a863
Target type
Instance
Protocol : Port
HTTP: 80
Protocol version
HTTP1
VPC
vpc-018d63db50fc89933
IP address type
IPv4
Load balancer
FrontendLB
1
Total targets
1
Healthy
0 Anomalous
0
Unhealthy
0
Unused
0
Initial
0
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags
Registered targets (1) Info
Anomaly mitigation: Not applicable
Deregister
Register targets
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.
Filter targets
Instance ID
Name
Port
Zone
Health status
Health status details
Administrative o...
Override details
Launch...
Anomaly
i-0e80f3a597fe2f18b
WebServer
80
us-east-1a (us...
Healthy
-
No override
No override is curre...
August 13...
Normal

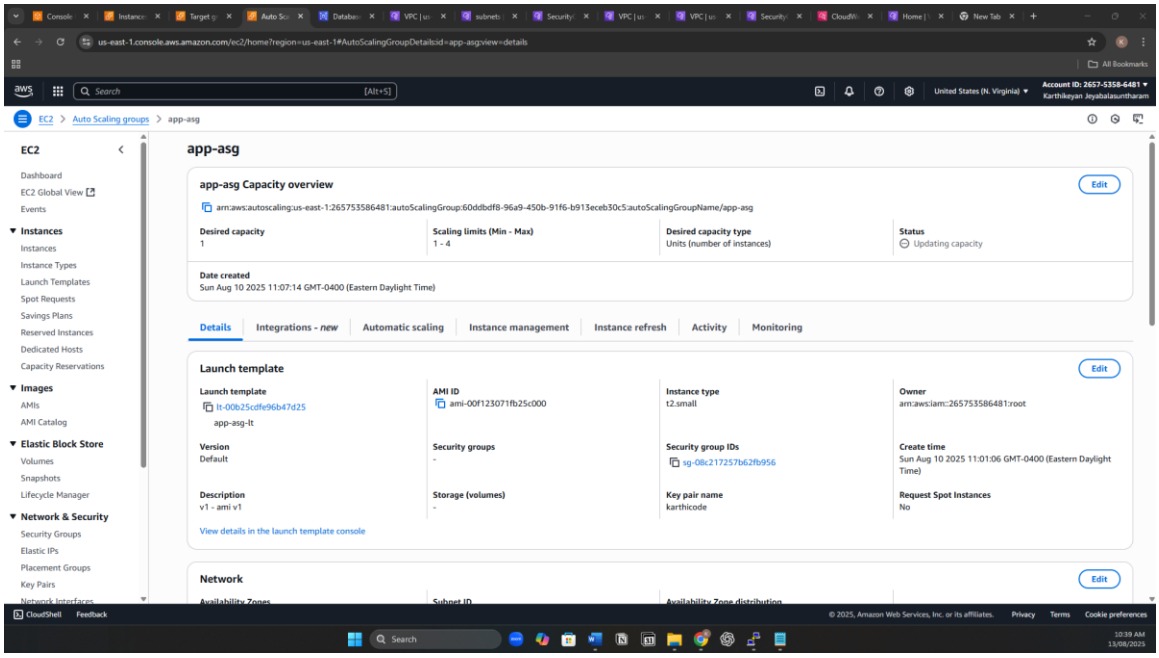


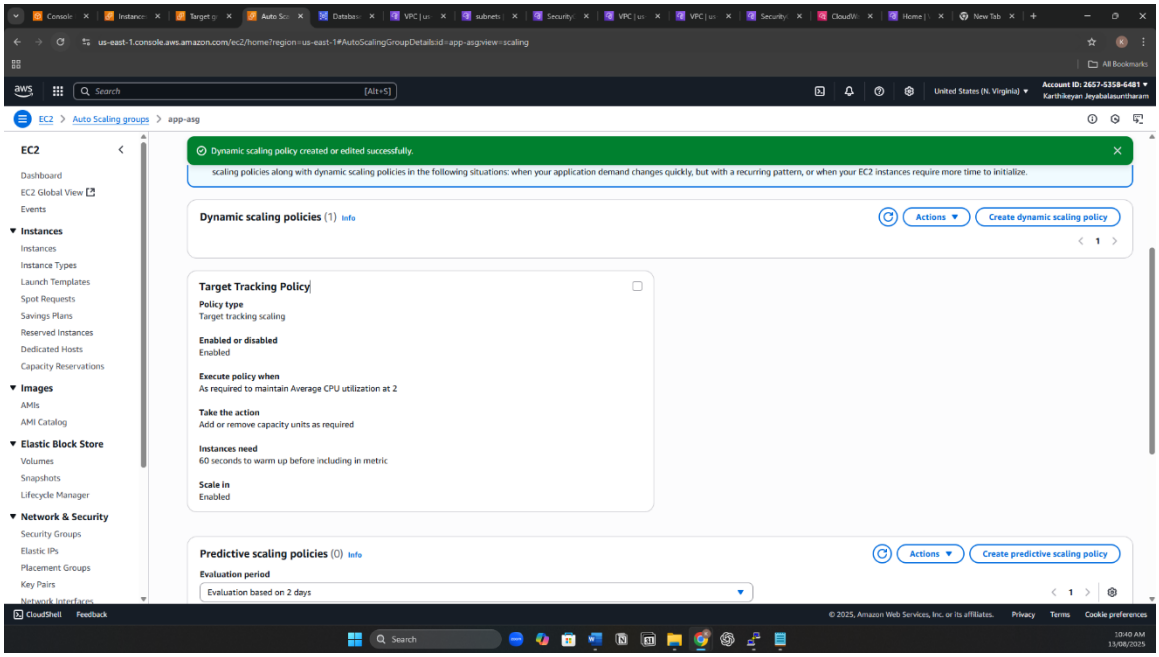
Auto Scaling

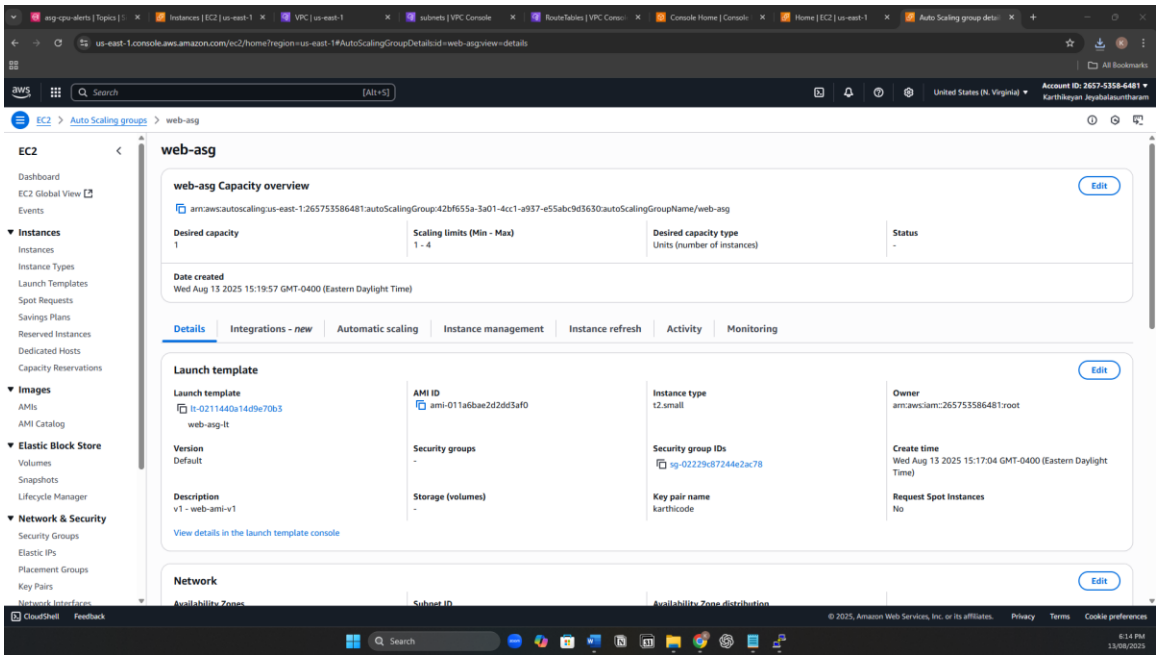
- Create Launch Templates for both Web Tier and App Tier:
    - Specify AMI, instance type, key pair, security group, and user data to install packages and deploy app.
- Create Auto Scaling Groups:
    - WebServer-ASG: Public subnets, attach Web Tier target group.
    - AppServer-ASG: Private app subnets, attach App Tier target group.
- Configure scaling policies:
    - Target tracking based on average CPU utilization.
- Enable notifications:
    - Create an Amazon SNS topic for Auto Scaling events.
    - Subscribe with email.
    - Attach SNS topic to the ASG notifications for instance launch/terminate events.

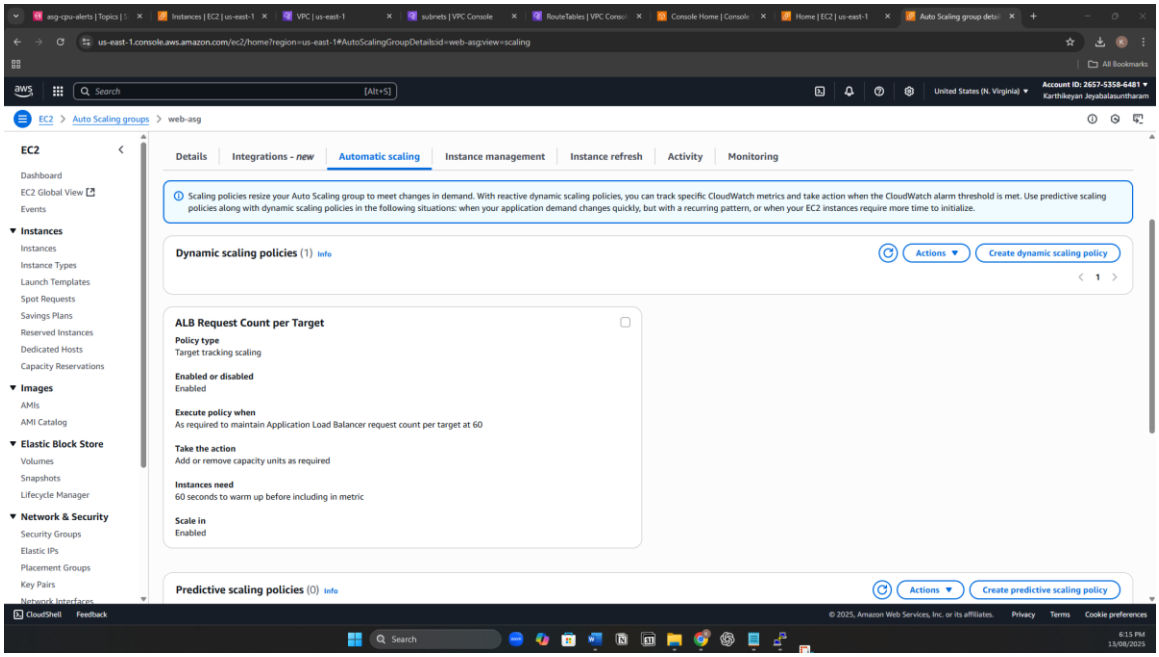

app-tg-5000
Actions
Details
arn:aws:elasticloadbalancing:us-east-1:265753586481:targetgroup/app-tg-5000/6ac6505d6db2e101
Target type
Instance
Protocol : Port
HTTP: 5000
Protocol version
HTTP1
VPC
vpc-018d63db50fc89933
IP address type
IPv4
Load balancer
Internal-App-ALB
4
Total targets
1
Healthy
0 Anomalous
0
Unhealthy
0
Unused
0
Initial
3
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags
Registered targets (4) Info
Anomaly mitigation: Not applicable
Deregister
Register targets
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.
Filter targets
Instance ID
Name
Port
Zone
Health status
Health status details
Admini...
Overri...
Launch time
i-02952614ab554eaf5
AppServer-ASG
5000
us-east-1a (us...
Draining
Target deregistration is in progress
No override.
No overri...
August 13, 2025, 18:55 (UT
i-0e48ce564f2e5036f
AppServer-ASG
5000
us-east-1b (us...
Healthy
-
No override.
No overri...
August 13, 2025, 18:55 (UT

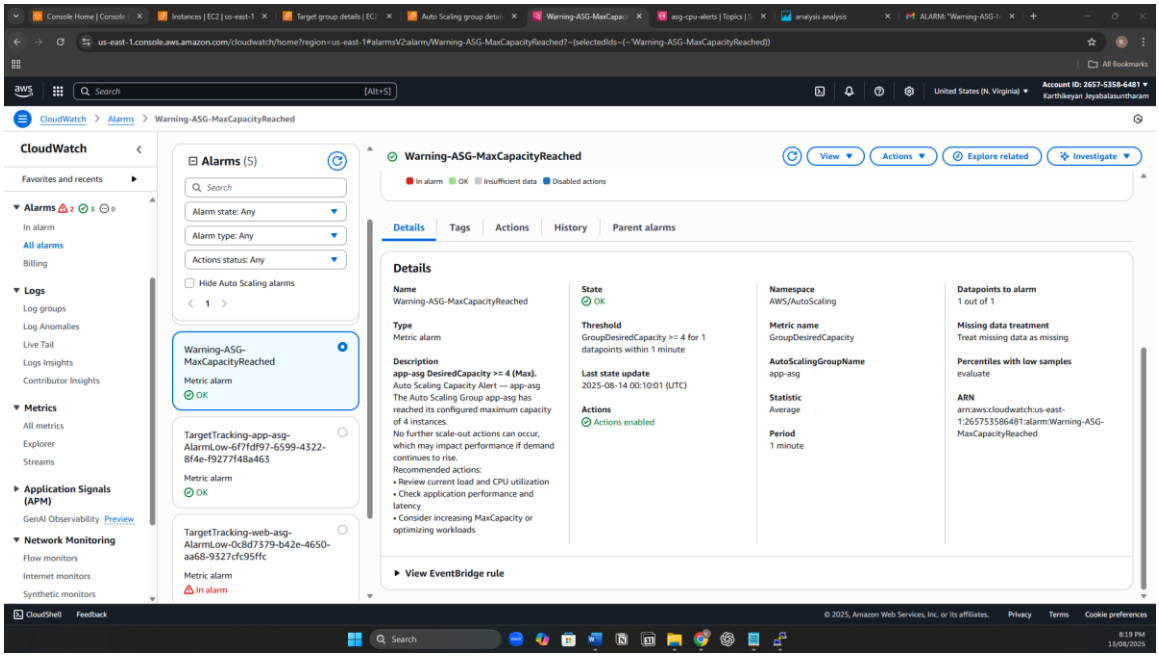

WebServerTG
Actions
Details
arn:aws:elasticloadbalancing:us-east-1:265753586481:targetgroup/WebServerTG/72ec628e8ab4a863
Target type
Instance
Protocol : Port
HTTP: 80
Protocol version
HTTP1
VPC
vpc-018d63db50fc89933
IP address type
IPv4
Load balancer
FrontendLB
2
Total targets
2
Healthy
0 Anomalous
0
Unhealthy
0
Unused
0
Initial
0
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags
Registered targets (2) Info
Anomaly mitigation: Not applicable
Deregister
Register targets
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.
Filter targets
Instance ID
Name
Port
Zone
Health status
Health status details
Administrative o...
Override details
Launch...
Anomaly
i-0130b4a6345518056
80
us-east-1a (us...
Healthy
-
No override
No override is curre...
August 13...
Normal
i-0e80f3a597fe2f18b
WebServer
80
us-east-1a (us...
Healthy
-
No override
No override is curre...
August 13...
Normal

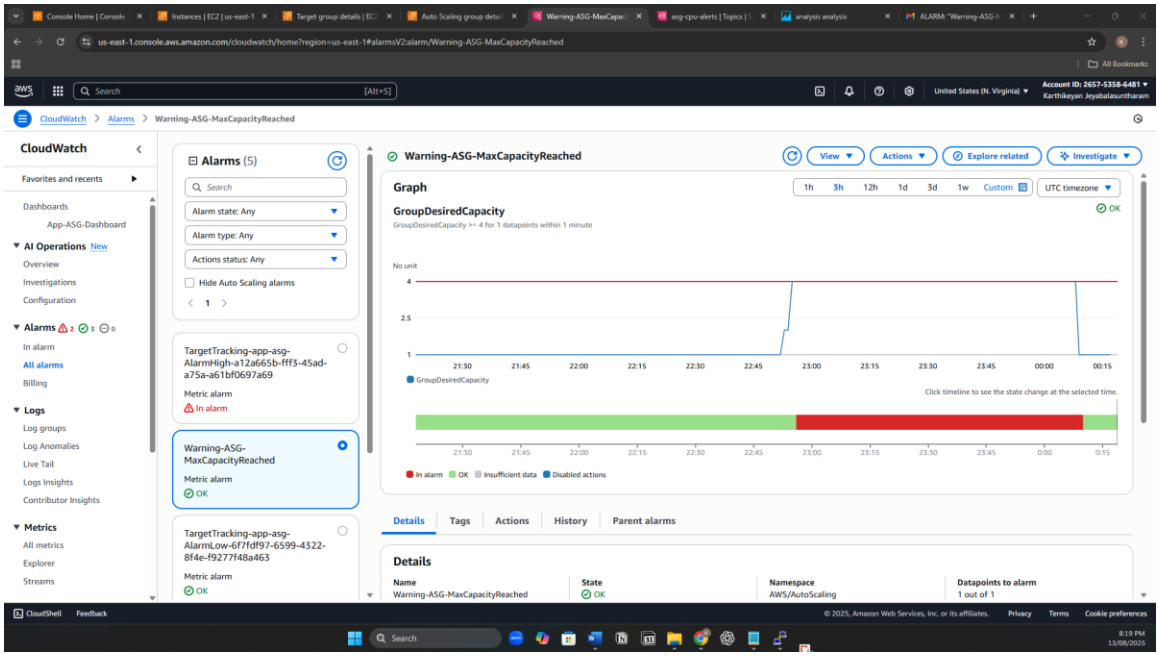

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroupDetails:id=app-asg;view=details
EC2 > Auto Scaling groups > app-asg
app-asg
app-asg Capacity overview
Edit
arn:aws:autoscaling:us-east-1:265753586481:autoScalingGroup:60ddbdf8-96a9-450b-91f6-b913eceb30c5:autoScalingGroupName/app-asg
Desired capacity
1
Scaling limits (Min - Max)
1 - 4
Desired capacity type
Units (number of instances)
Status
Updating capacity
Date created
Sun Aug 10 2025 11:07:14 GMT-0400 (Eastern Daylight Time)
Details
Integrations - new
Automatic scaling
Instance management
Instance refresh
Activity
Monitoring
Launch template
Edit
Launch template
lt-00b25cdfe96b47d25
app-asg-lt
AMI ID
ami-00f123071fb25c000
Instance type
t2.small
Owner
arn:aws:iam::265753586481:root
Version
Default
Security groups
-
Security group IDs
sg-08c217257b62fb956
Create time
Sun Aug 10 2025 11:01:06 GMT-0400 (Eastern Daylight Time)
Description
v1 - ami v1
Storage (volumes)
-
Key pair name
karthicode
Request Spot Instances
No
View details in the launch template console
Network
Edit
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

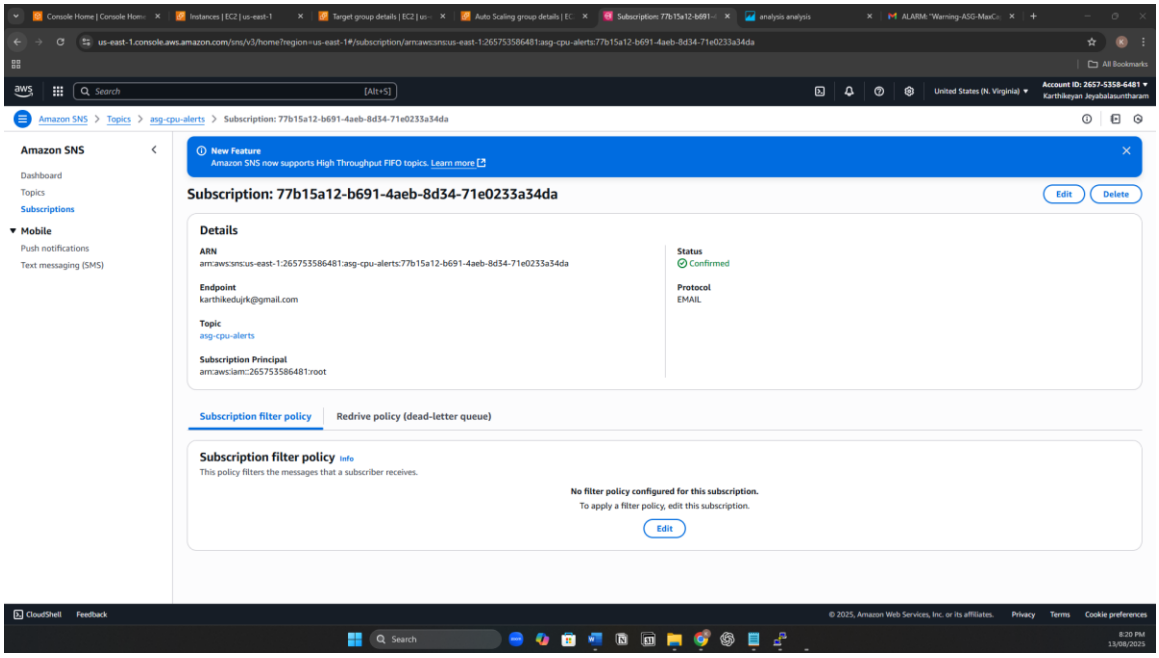

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroupDetails:id=app-asg;view=scaling
EC2 > Auto Scaling groups > app-asg
Dynamic scaling policy created or edited successfully.
scaling policies along with dynamic scaling policies in the following situations: when your application demand changes quickly, but with a recurring pattern, or when your EC2 instances require more time to initialize.
Dynamic scaling policies (1) Info
Actions
Create dynamic scaling policy
Target Tracking Policy
Policy type
Target tracking scaling
Enabled or disabled
Enabled
Execute policy when
As required to maintain Average CPU utilization at 2
Take the action
Add or remove capacity units as required
Instances need
60 seconds to warm up before including in metric
Scale in
Enabled
Predictive scaling policies (0) Info
Actions
Create predictive scaling policy
Evaluation period
Evaluation based on 2 days
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

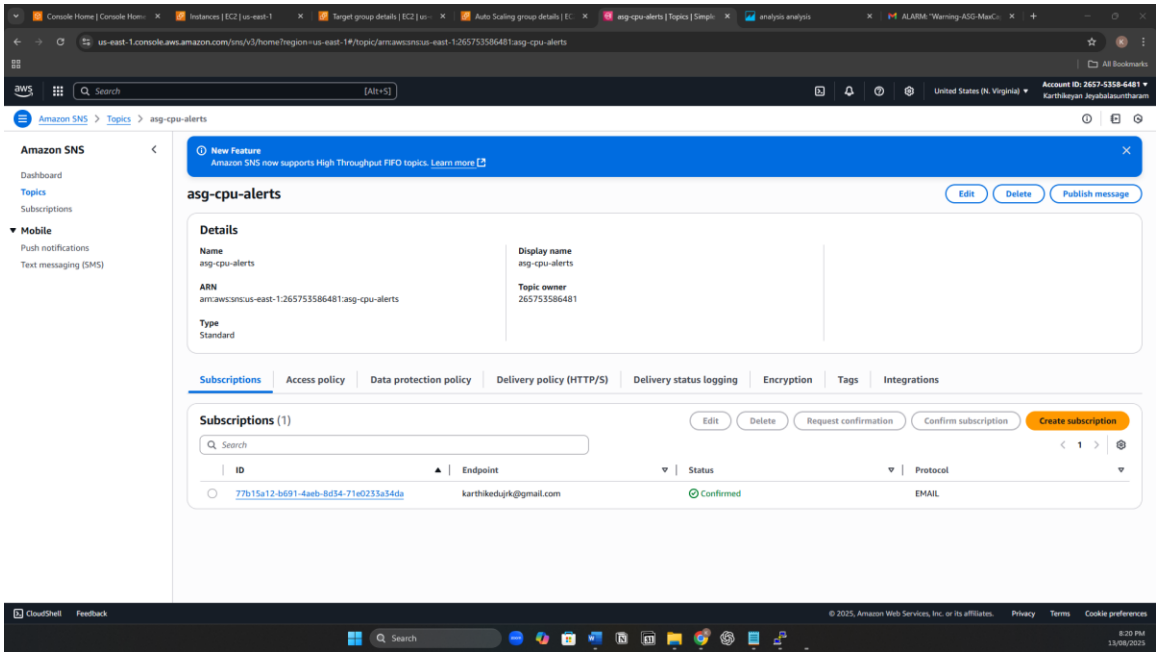

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroupDetails:id=web-asg;view=details
EC2 > Auto Scaling groups > web-asg
web-asg
web-asg Capacity overview
Edit
arn:aws:autoscaling:us-east-1:265753586481:autoScalingGroup:42bf655a-3a01-4cc1-a937-e55abc9d3630:autoScalingGroupName/web-asg
Desired capacity
1
Scaling limits (Min - Max)
1 - 4
Desired capacity type
Units (number of instances)
Status
-
Date created
Wed Aug 13 2025 15:19:57 GMT-0400 (Eastern Daylight Time)
Details
Integrations - new
Automatic scaling
Instance management
Instance refresh
Activity
Monitoring
Launch template
Edit
Launch template
lt-0211440a14d9e70b3
web-asg-lt
AMI ID
ami-011a6bae2d2dd3af0
Instance type
t2.small
Owner
arn:aws:iam::265753586481:root
Version
Default
Security groups
-
Security group IDs
sg-02229c87244e2ac78
Create time
Wed Aug 13 2025 15:17:04 GMT-0400 (Eastern Daylight Time)
Description
v1 - web-ami-v1
Storage (volumes)
-
Key pair name
karthicode
Request Spot Instances
No
View details in the launch template console
Network
Edit


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroupDetails:id=web-asg;view=scaling
EC2 > Auto Scaling groups > web-asg
Details
Integrations - new
Automatic scaling
Instance management
Instance refresh
Activity
Monitoring
Scaling policies resize your Auto Scaling group to meet changes in demand. With reactive dynamic scaling policies, you can track specific CloudWatch metrics and take action when the CloudWatch alarm threshold is met. Use predictive scaling policies along with dynamic scaling policies in the following situations: when your application demand changes quickly, but with a recurring pattern, or when your EC2 instances require more time to initialize.
Dynamic scaling policies (1) Info
Actions
Create dynamic scaling policy
ALB Request Count per Target
Policy type
Target tracking scaling
Enabled or disabled
Enabled
Execute policy when
As required to maintain Application Load Balancer request count per target at 60
Take the action
Add or remove capacity units as required
Instances need
60 seconds to warm up before including in metric
Scale in
Enabled
Predictive scaling policies (0) Info
Actions
Create predictive scaling policy


CloudWatch
Alarms (5)
Warning-ASG-MaxCapacityReached
Details
Tags
Actions
History
Parent alarms
Name
Warning-ASG-MaxCapacityReached
State
OK
Namespace
AWS/AutoScaling
Datapoints to alarm
1 out of 1
Type
Metric alarm
Threshold
GroupDesiredCapacity >= 4 for 1 datapoints within 1 minute
Metric name
GroupDesiredCapacity
Missing data treatment
Treat missing data as missing
Description
app-asg DesiredCapacity >= 4 (Max).
Auto Scaling Capacity Alert — app-asg
The Auto Scaling Group app-asg has reached its configured maximum capacity of 4 instances.
No further scale-out actions can occur, which may impact performance if demand continues to rise.
Recommended actions:
• Review current load and CPU utilization
• Check application performance and latency
• Consider increasing MaxCapacity or optimizing workloads
Last state update
2025-08-14 00:10:01 (UTC)
AutoScalingGroupName
app-asg
Percentiles with low samples
evaluate
Actions
Actions enabled
Statistic
Average
ARN
arn:aws:cloudwatch:us-east-1:265753586481:alarm:Warning-ASG-MaxCapacityReached
Period
1 minute
View EventBridge rule

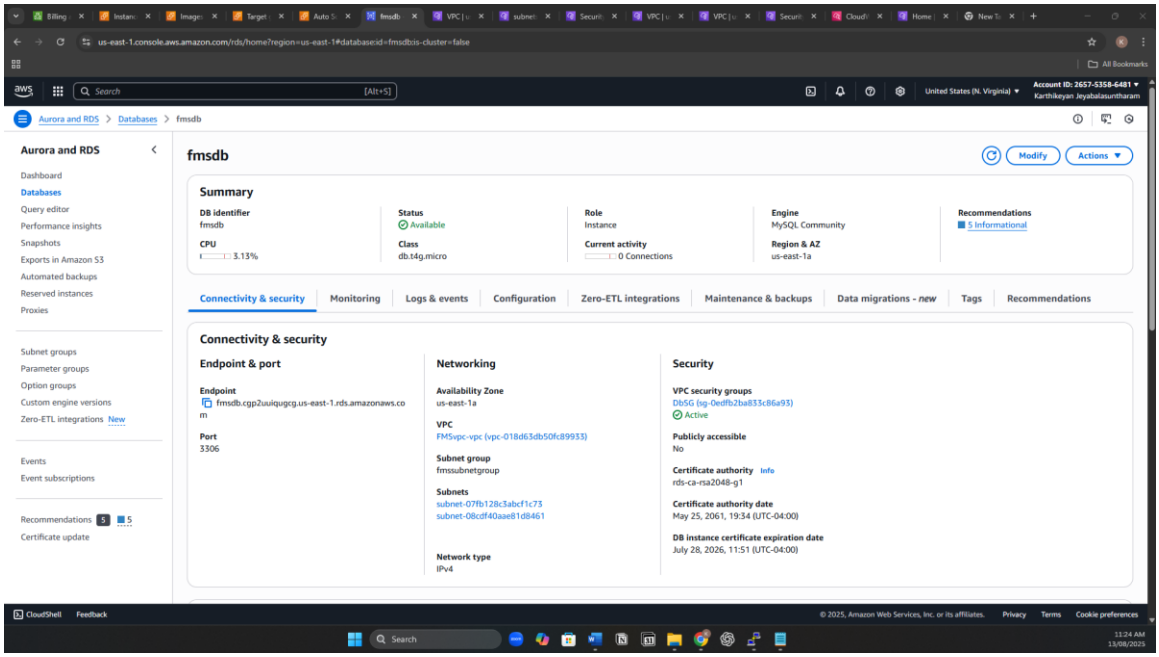

CloudWatch
Alarms (5)
Warning-ASG-MaxCapacityReached
Graph
GroupDesiredCapacity
Details
Tags
Actions
History
Parent alarms
Name
Warning-ASG-MaxCapacityReached
State
OK
Namespace
AWS/AutoScaling
Datapoints to alarm
1 out of 1

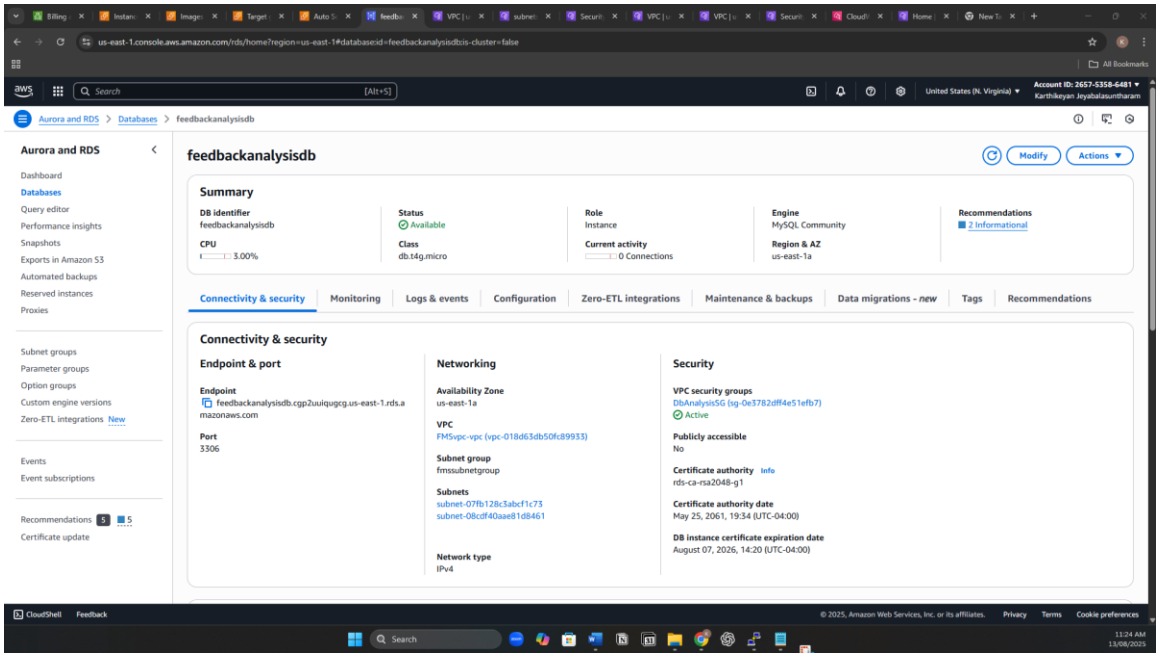


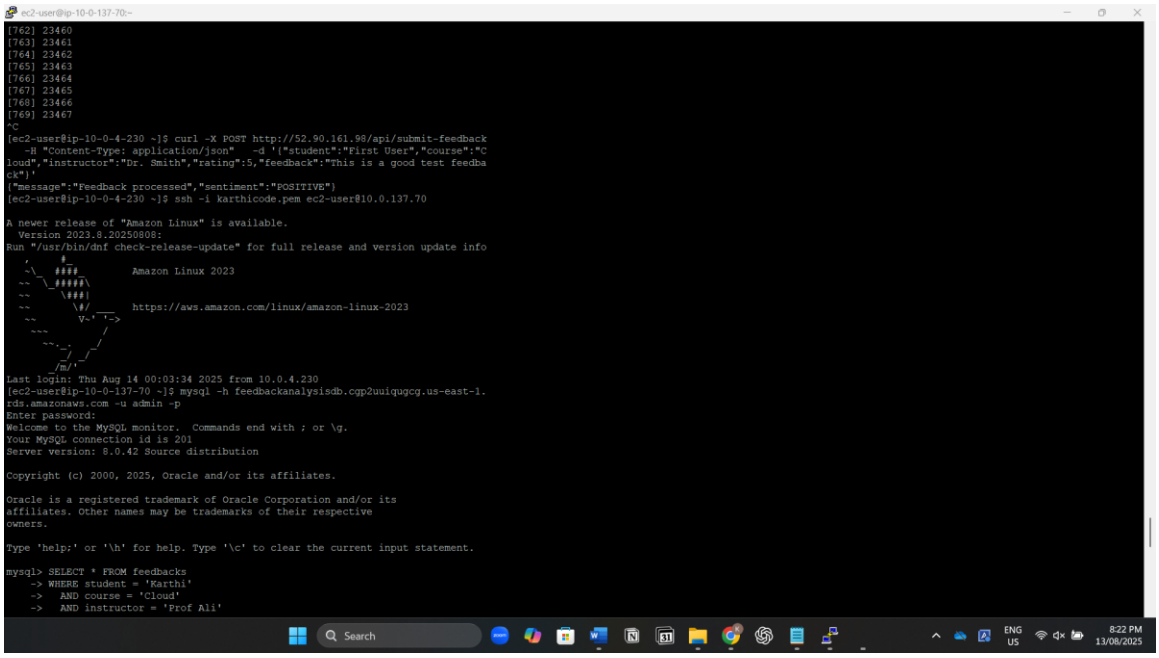


Database Setup

- Launch Amazon RDS MySQL in private DB subnets.
- Assign DB-SG to RDS instance.
- Set database credentials and note the endpoint.
- Update application configuration to connect to RDS endpoint with secure credentials.

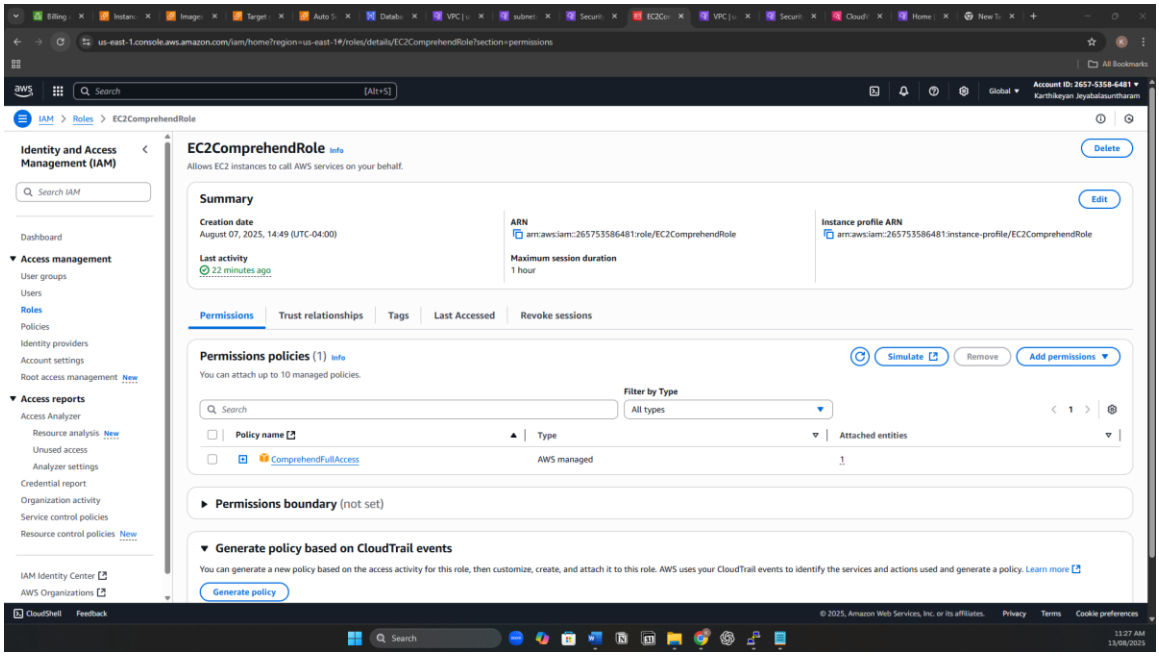

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databaseid=fmsdb;is-cluster=false
Search [Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
Aurora and RDS > Databases > fmsdb
Aurora and RDS
Dashboard
Databases
Query editor
Performance insights
Snapshots
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 5 5
Certificate update
fmsdb
Modify
Actions
Summary
DB identifier
fmsdb
Status
Available
Role
Instance
Engine
MySQL Community
Recommendations
5 Informational
CPU
3.13%
Class
db.t4g.micro
Current activity
0 Connections
Region & AZ
us-east-1a
Connectivity & security
Monitoring
Logs & events
Configuration
Zero-ETL integrations
Maintenance & backups
Data migrations - new
Tags
Recommendations
Connectivity & security
Endpoint & port
Endpoint
fmsdb.cgp2uuiqugcg.us-east-1.rds.amazonaws.com
Port
3306
Networking
Availability Zone
us-east-1a
VPC
FMSvpc-vpc (vpc-018d63db50fc89933)
Subnet group
fmssubnetgroup
Subnets
subnet-07fb128c3abcf1c73
subnet-08cdf40aae81d8461
Network type
IPv4
Security
VPC security groups
Db5G (sg-0edfb2ba833c86a93)
Active
Publicly accessible
No
Certificate authority Info
rds-ca-rsa2048-g1
Certificate authority date
May 25, 2061, 19:34 (UTC-04:00)
DB instance certificate expiration date
July 28, 2026, 11:51 (UTC-04:00)
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
11:24 AM
13/08/2025

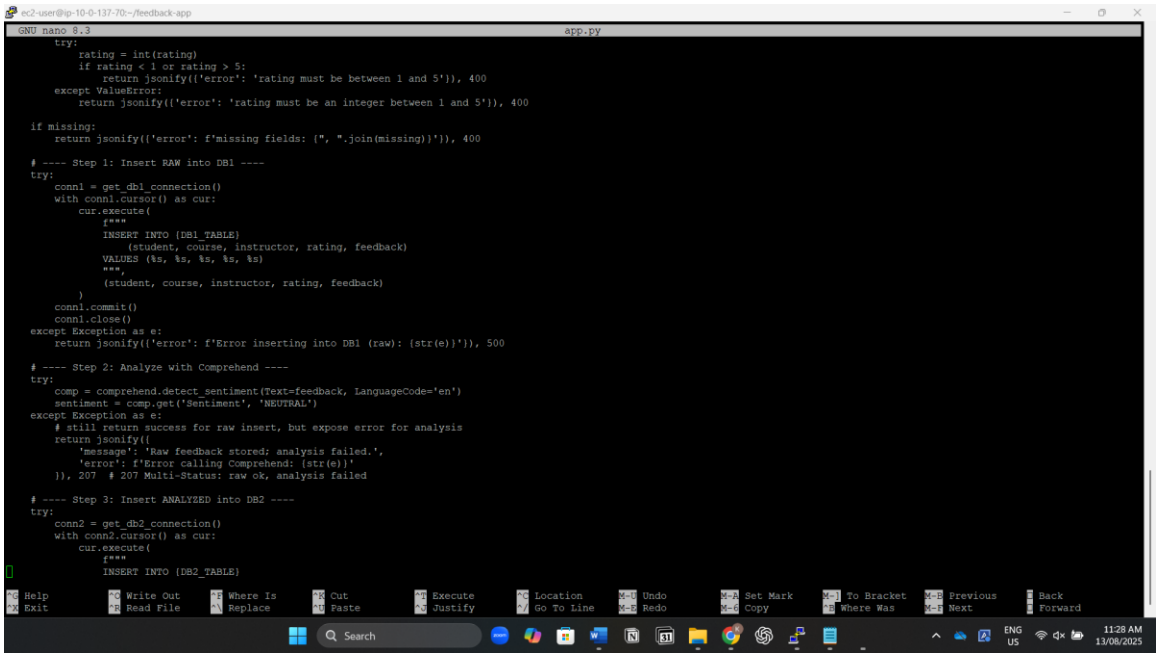

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databaseid=feedbackanalysisdb;is-cluster=false
Search [Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
Aurora and RDS > Databases > feedbackanalysisdb
Aurora and RDS
Dashboard
Databases
Query editor
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 5 5
Certificate update
feedbackanalysisdb
Modify
Actions
Summary
DB identifier
feedbackanalysisdb
Status
Available
Role
Instance
Engine
MySQL Community
Recommendations
2 Informational
CPU
3.00%
Class
db.t4g.micro
Current activity
0 Connections
Region & AZ
us-east-1a
Connectivity & security
Monitoring
Logs & events
Configuration
Zero-ETL integrations
Maintenance & backups
Data migrations - new
Tags
Recommendations
Connectivity & security
Endpoint & port
Endpoint
feedbackanalysisdb.cgp2uuiqugcg.us-east-1.rds.amazonaws.com
Port
3306
Networking
Availability Zone
us-east-1a
VPC
FMSvpc-vpc (vpc-018d63db50fc89933)
Subnet group
fmssubnetgroup
Subnets
subnet-07fb128c3abcf1c73
subnet-08cdf40aae81d8461
Network type
IPv4
Security
VPC security groups
DbAnalysis5G (sg-0e3782dff4e51efb7)
Active
Publicly accessible
No
Certificate authority Info
rds-ca-rsa2048-g1
Certificate authority date
May 25, 2061, 19:34 (UTC-04:00)
DB instance certificate expiration date
August 07, 2026, 14:20 (UTC-04:00)
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
11:24 AM
13/08/2025



AWS Comprehend Integration

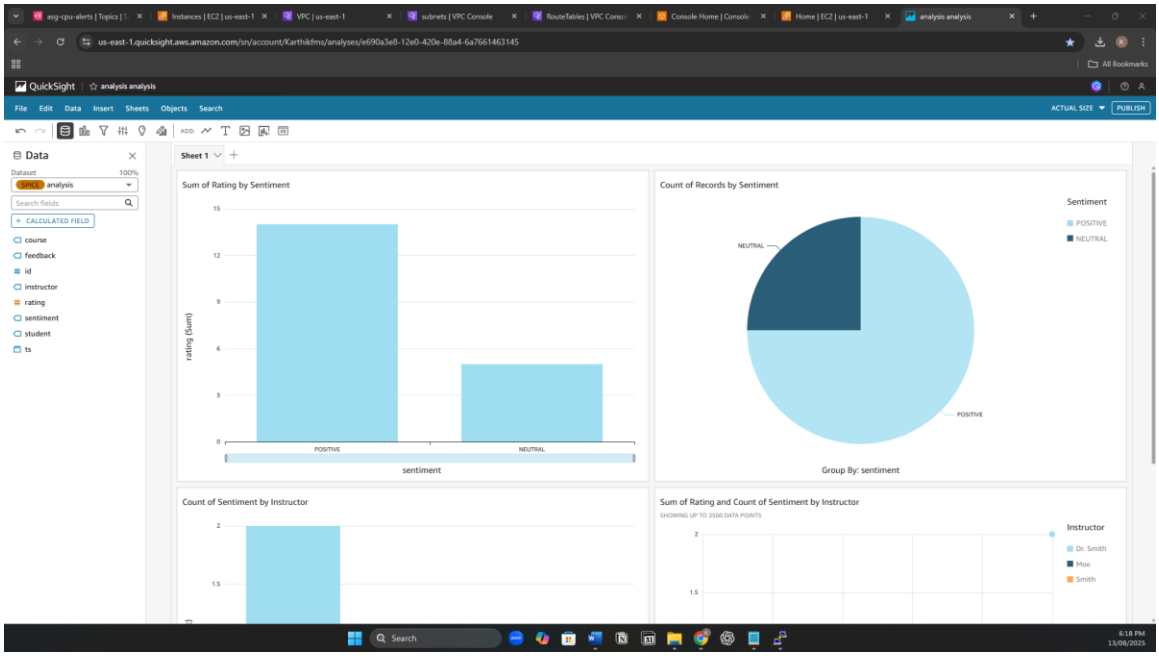
- Create an IAM Role for EC2 with permission comprehend:FullAccess.
- Attach the IAM role to App Tier EC2 instances.
- Update application backend code to:
    - Call AWS Comprehend's API with user feedback text.
    - Store sentiment result (Positive, Negative, Neutral) along with feedback in RDS.





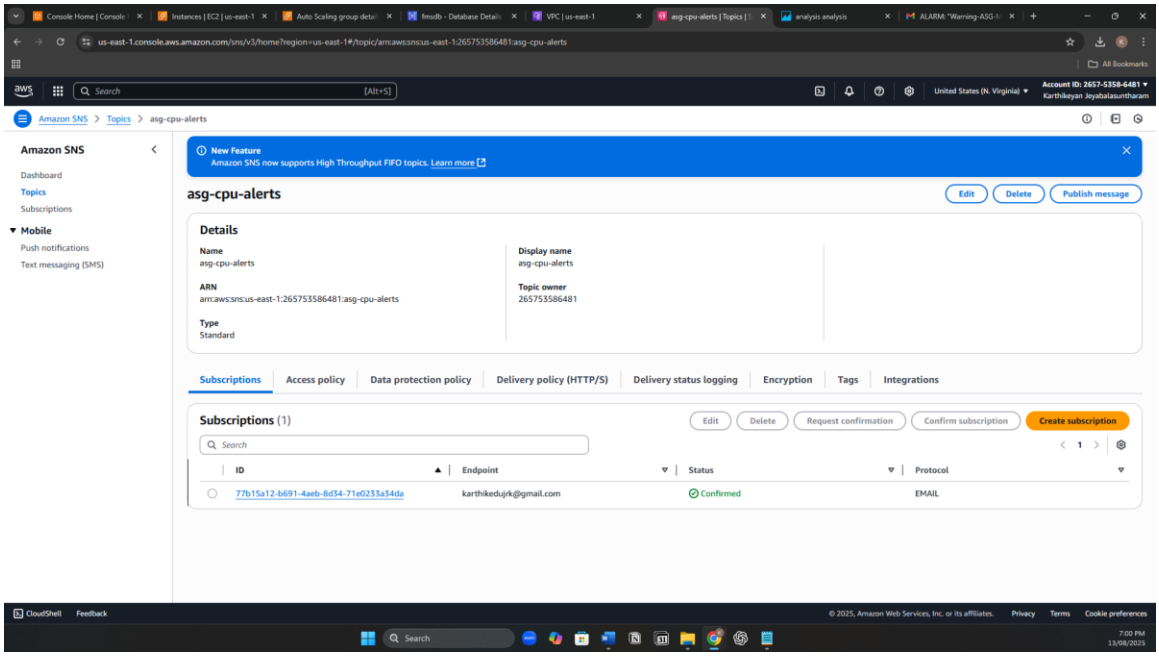
Visualization with QuickSight

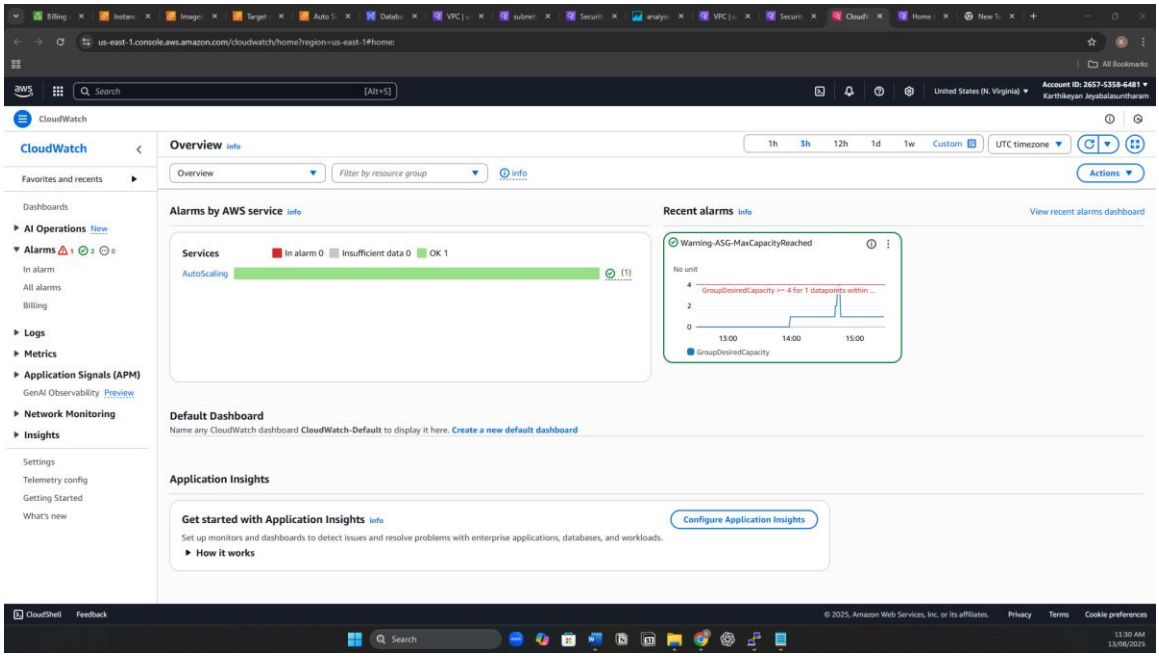
- Ensure QuickSight has a VPC connection to access RDS in private subnets.
- In QuickSight, create a new data source and connect to the RDS MySQL instance.
- Create datasets from feedback and sentiment tables.
- Build visual dashboards showing sentiment distribution, rating averages, and trends over time.

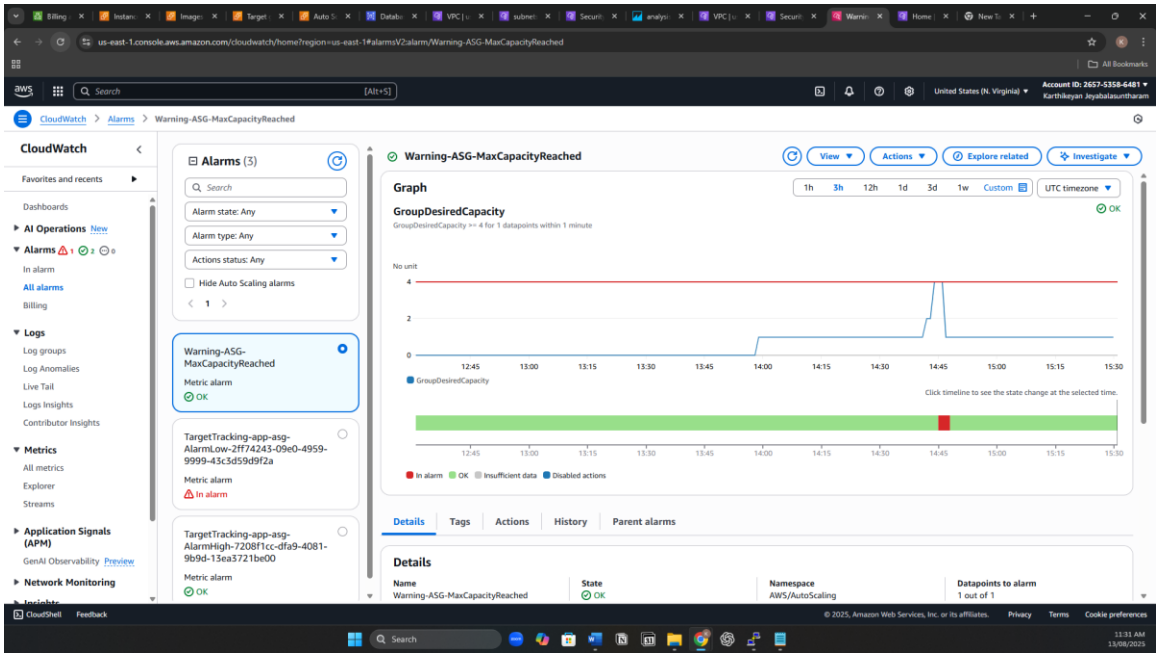


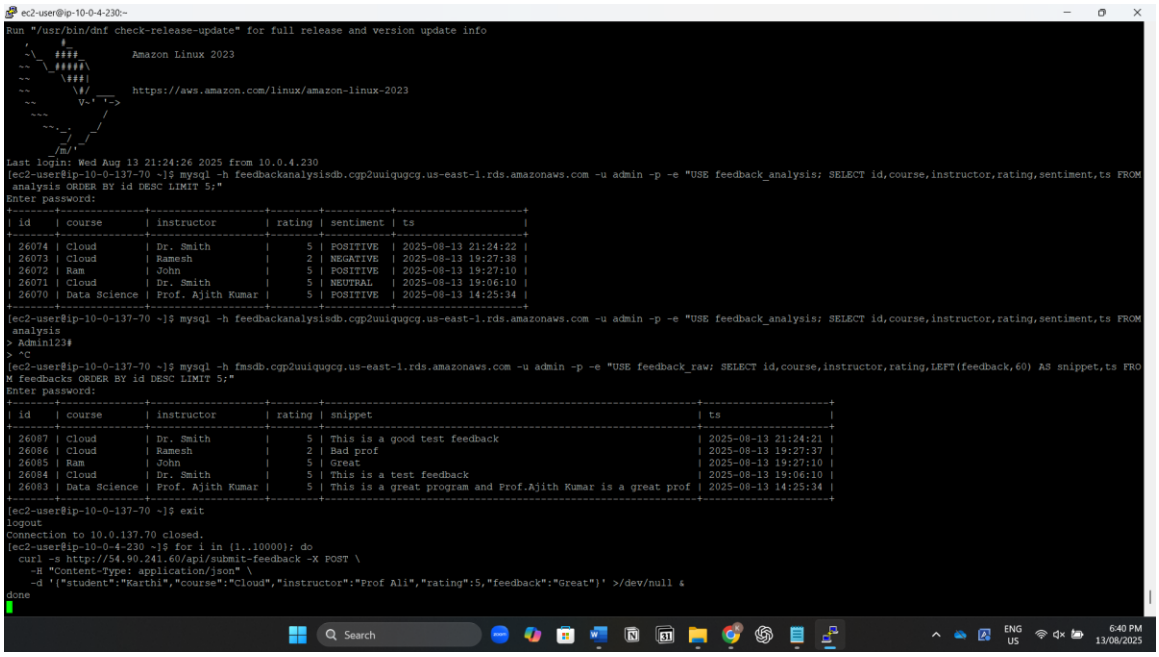
Monitoring & Logging

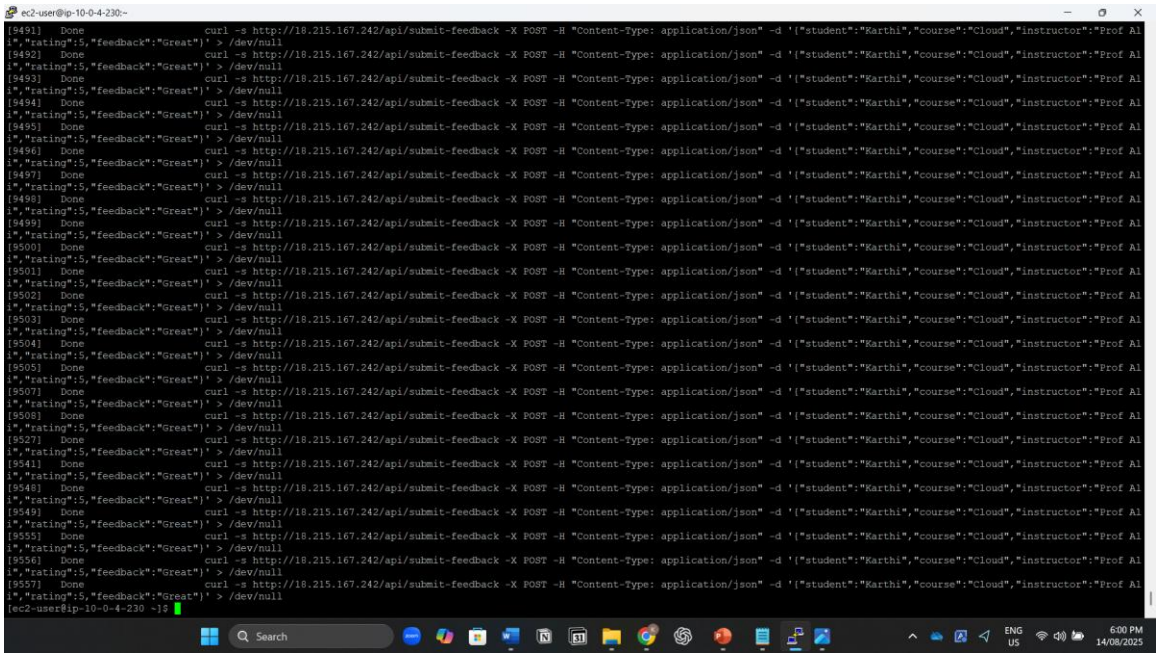
- Create CloudWatch Alarms for:
    - High CPU utilization on web/app instances.
- Configure alarm actions to send notifications to SNS topic and send email notification.
- Review metrics and alarms regularly to ensure system health and performance.

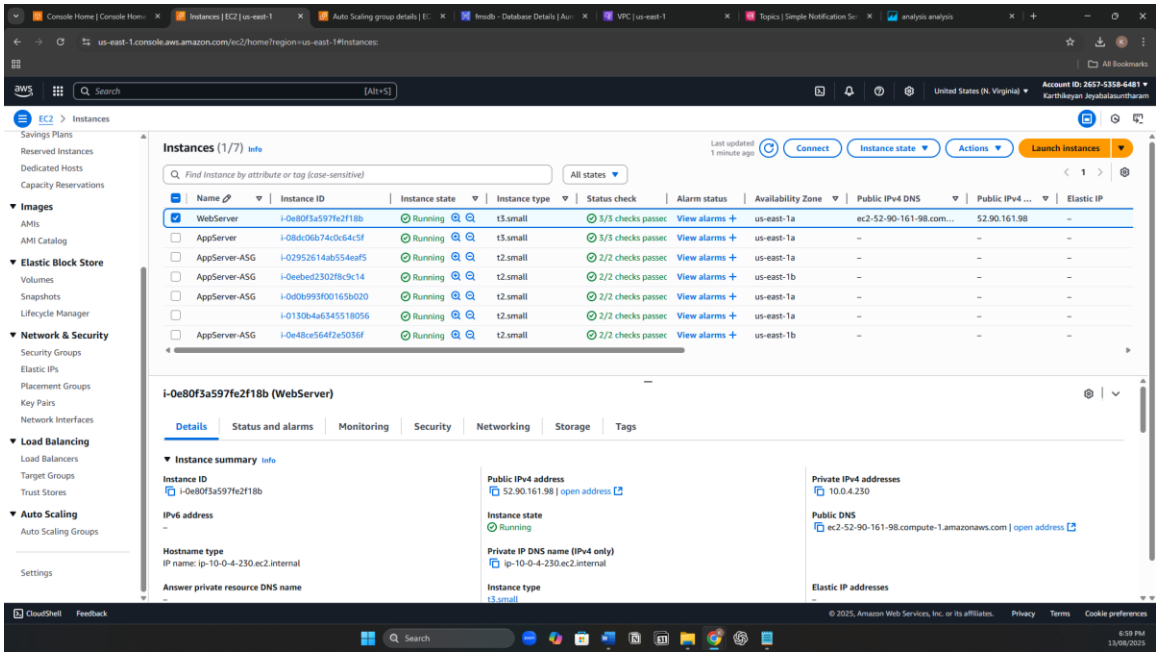


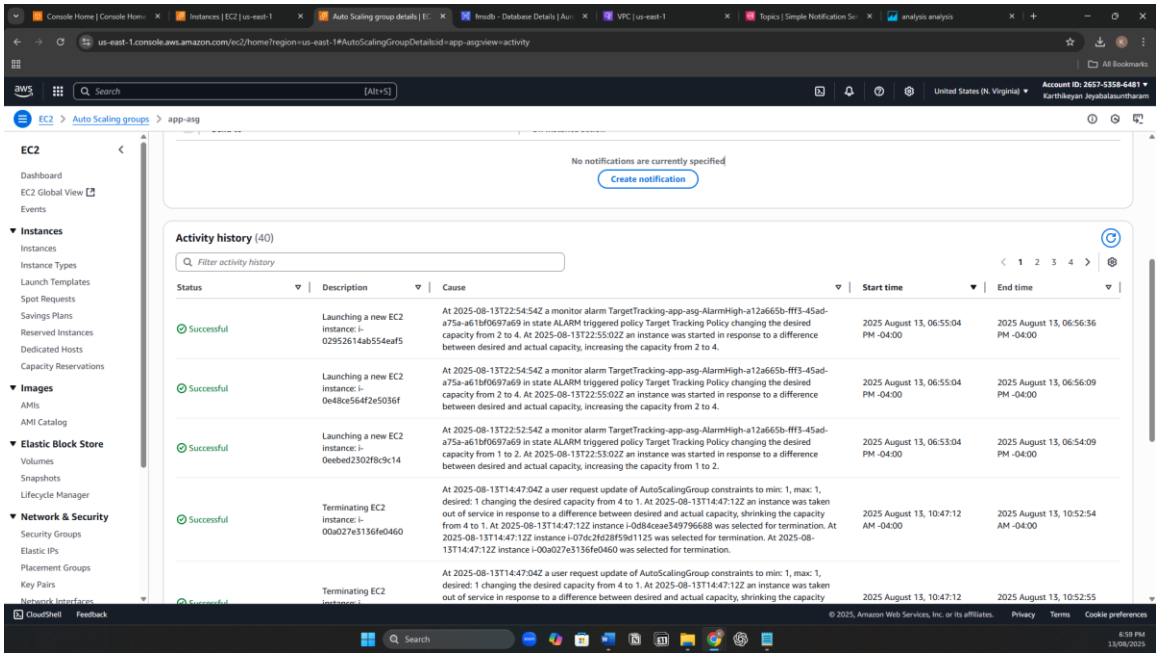


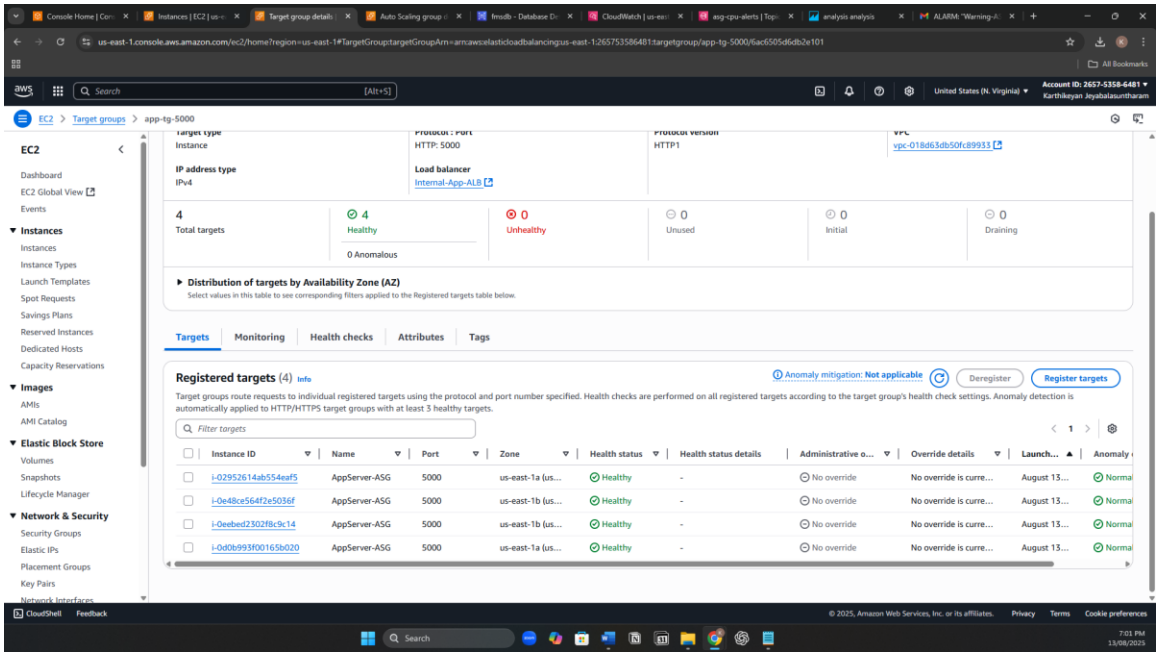

us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#alarmsV2:alarm/Warning-ASG-MaxCapacityReached
CloudWatch > Alarms > Warning-ASG-MaxCapacityReached
CloudWatch
Alarms (3)
Warning-ASG-MaxCapacityReached
Graph
GroupDesiredCapacity
Details
Tags
Actions
History
Parent alarms
Name
Warning-ASG-MaxCapacityReached
State
OK
Namespace
AWS/AutoScaling
Datapoints to alarm
1 out of 1

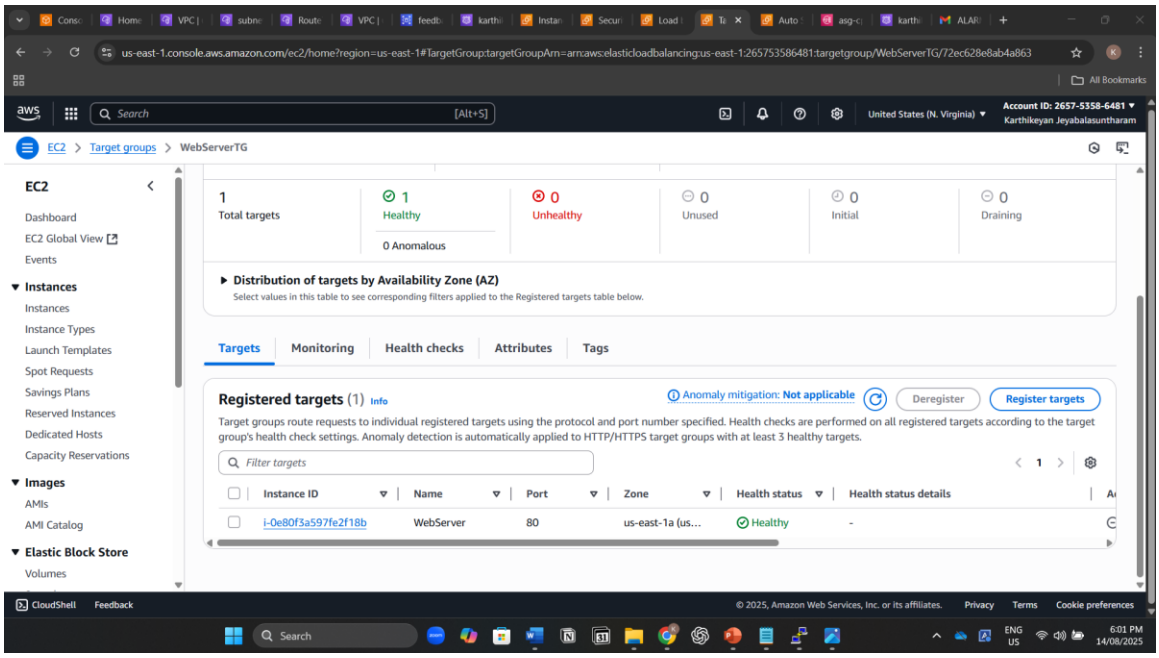

ec2-user@ip-10-0-4-230:~
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Wed Aug 13 21:24:26 2025 from 10.0.4.230
[ec2-user@ip-10-0-137-70 ~]$ mysql -h feedbackanalysisdb.cgp2uuiqugcg.us-east-1.rds.amazonaws.com -u admin -p -e "USE feedback_analysis; SELECT id,course,instructor,rating,sentiment,ts FROM analysis ORDER BY id DESC LIMIT 5;"
Enter password:
id	course	instructor	rating	sentiment	ts
26074	Cloud	Dr. Smith	5	POSITIVE	2025-08-13 21:24:22
26073	Cloud	Ramesh	2	NEGATIVE	2025-08-13 19:27:38
26072	Ram	John	5	POSITIVE	2025-08-13 19:27:10
26071	Cloud	Dr. Smith	5	NEUTRAL	2025-08-13 19:06:10
26070	Data Science	Prof. Ajith Kumar	5	POSITIVE	2025-08-13 14:25:34
[ec2-user@ip-10-0-137-70 ~]$ mysql -h fmsdb.cgp2uuiqugcg.us-east-1.rds.amazonaws.com -u admin -p -e "USE feedback_raw; SELECT id,course,instructor,rating,LEFT(feedback,60) AS snippet,ts FROM feedbacks ORDER BY id DESC LIMIT 5;"
Enter password:
id	course	instructor	rating	snippet	ts
26087	Cloud	Dr. Smith	5	This is a good test feedback	2025-08-13 21:24:21
26086	Cloud	Ramesh	2	Bad prof	2025-08-13 19:27:37
26085	Ram	John	5	Great	2025-08-13 19:27:10
26084	Cloud	Dr. Smith	5	This is a test feedback	2025-08-13 19:06:10
26083	Data Science	Prof. Ajith Kumar	5	This is a great program and Prof.Ajith Kumar is a great prof	2025-08-13 14:25:34
[ec2-user@ip-10-0-137-70 ~]$ exit
logout
Connection to 10.0.137.70 closed.
[ec2-user@ip-10-0-4-230 ~]$ for i in {1..10000}; do
curl -s http://54.90.241.60/api/submit-feedback -X POST \
-H "Content-Type: application/json" \
-d '{"student":"Karthi","course":"Cloud","instructor":"Prof Ali","rating":5,"feedback":"Great"}' >/dev/null &
done

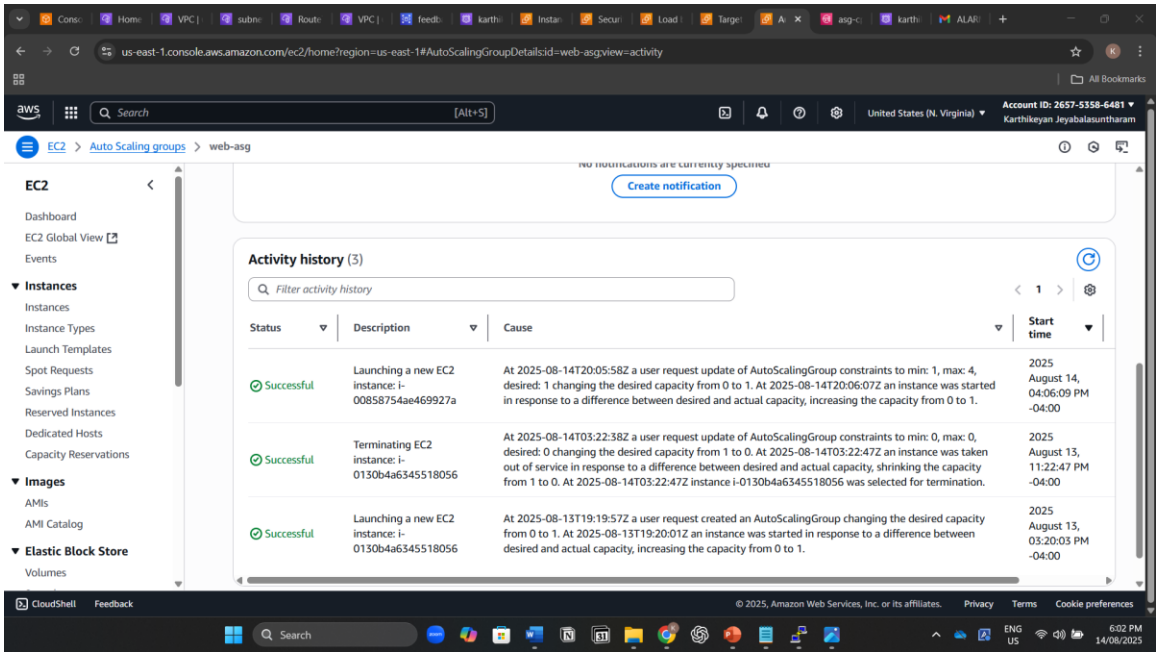

ec2-user@ip-10-0-4-230:~
[9491] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9492] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9493] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9494] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9495] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9496] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9497] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9498] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9499] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9500] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9501] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9502] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9503] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9504] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9505] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9507] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9508] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9527] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9541] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9548] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9549] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9555] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9556] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[9557] Done curl -s http://18.215.167.242/api/submit-feedback -X POST -H "Content-Type: application/json" -d '{"student":"Karthi","course":"Cloud","instructor":"Prof Al
i","rating":5,"feedback":"Great"}' > /dev/null
[ec2-user@ip-10-0-4-230 ~]$
Search
ENG
US
6:00 PM
14/08/2025

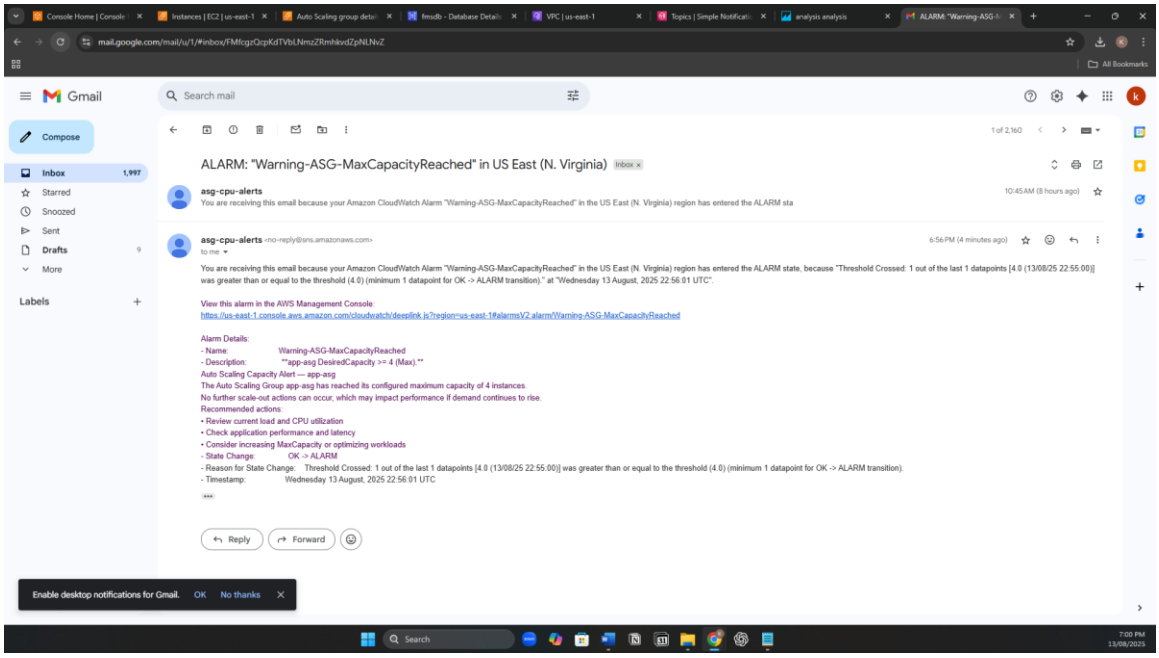

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2657-5358-6481
Karthikeyan Jeyabalasuntharam
EC2 > Instances
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores
Auto Scaling
Auto Scaling Groups
Settings
Instances (1/7) Info
Last updated 1 minute 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 status
Availability Zone
Public IPv4 DNS
Public IPv4 ...
Elastic IP
WebServer i-0e80f3a597fe2f18b Running t3.small 3/3 checks passec View alarms us-east-1a ec2-52-90-161-98.com... 52.90.161.98
AppServer i-08dc06b74c0c64c5f Running t3.small 3/3 checks passec View alarms us-east-1a
AppServer-ASG i-02952614ab554eaf5 Running t2.small 2/2 checks passec View alarms us-east-1a
AppServer-ASG i-0eebed2302f8c9c14 Running t2.small 2/2 checks passec View alarms us-east-1b
AppServer-ASG i-0d0b993f00165b020 Running t2.small 2/2 checks passec View alarms us-east-1a
i-0130b4a6345518056 Running t2.small 2/2 checks passec View alarms us-east-1a
AppServer-ASG i-0e48ce564f2e5036f Running t2.small 2/2 checks passec View alarms us-east-1b
i-0e80f3a597fe2f18b (WebServer)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-0e80f3a597fe2f18b
IPv6 address
Hostname type
IP name: ip-10-0-4-230.ec2.internal
Answer private resource DNS name
Public IPv4 address
52.90.161.98 | open address
Instance state
Running
Private IP DNS name (IPv4 only)
ip-10-0-4-230.ec2.internal
Instance type
t3.small
Private IPv4 addresses
10.0.4.230
Public DNS
ec2-52-90-161-98.compute-1.amazonaws.com | open address
Elastic IP addresses
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
6:59 PM
13/08/2025

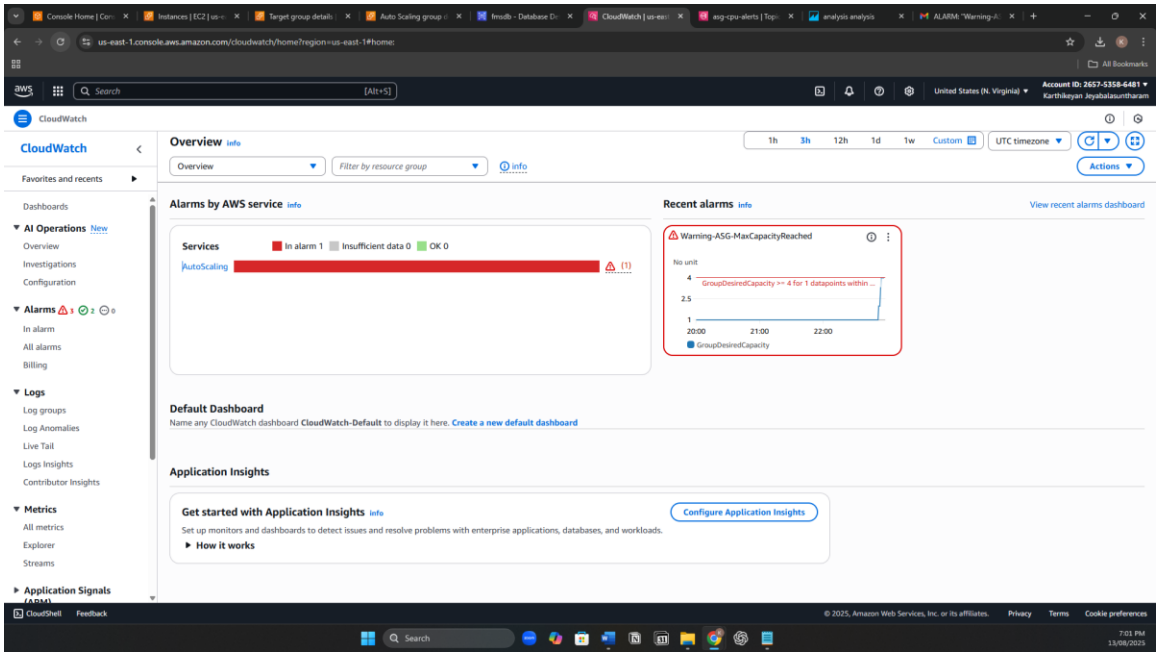

No notifications are currently specified
Create notification
Activity history (40)
Filter activity history
Status
Description
Cause
Start time
End time
Successful
Launching a new EC2 instance: i-02952614ab554eaf5
At 2025-08-13T22:54:54Z a monitor alarm TargetTracking-app-asg-AlarmHigh-a12a665b-fff3-45ad-a75a-a61bf0697a69 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 2 to 4. At 2025-08-13T22:55:02Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 2 to 4.
2025 August 13, 06:55:04 PM -04:00
2025 August 13, 06:56:36 PM -04:00
Launching a new EC2 instance: i-0e48ce564f2e5036f
2025 August 13, 06:56:09 PM -04:00
Launching a new EC2 instance: i-0eebed2302f8c9c14
At 2025-08-13T22:52:54Z a monitor alarm TargetTracking-app-asg-AlarmHigh-a12a665b-fff3-45ad-a75a-a61bf0697a69 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 1 to 2. At 2025-08-13T22:53:02Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 1 to 2.
2025 August 13, 06:53:04 PM -04:00
2025 August 13, 06:54:09 PM -04:00
Terminating EC2 instance: i-00a027e3136fe0460
At 2025-08-13T14:47:04Z a user request update of AutoScalingGroup constraints to min: 1, max: 1, desired: 1 changing the desired capacity from 4 to 1. At 2025-08-13T14:47:12Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 4 to 1. At 2025-08-13T14:47:12Z instance i-0d84ceae349796688 was selected for termination. At 2025-08-13T14:47:12Z instance i-07dc2fd28f59d1125 was selected for termination. At 2025-08-13T14:47:12Z instance i-00a027e3136fe0460 was selected for termination.
2025 August 13, 10:47:12 AM -04:00
2025 August 13, 10:52:54 AM -04:00
Terminating EC2 instance: i-
At 2025-08-13T14:47:04Z a user request update of AutoScalingGroup constraints to min: 1, max: 1, desired: 1 changing the desired capacity from 4 to 1. At 2025-08-13T14:47:12Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity
2025 August 13, 10:47:12
2025 August 13, 10:52:55

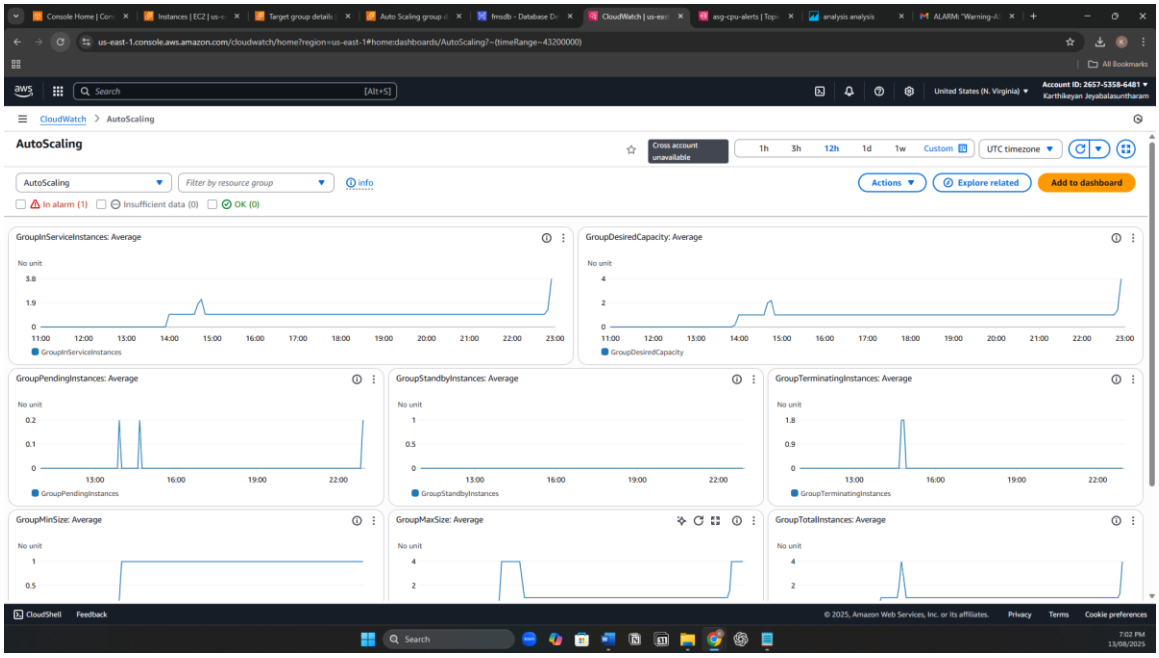

Instance
HTTP: 5000
HTTP1
vpc-018d63db50fc89933
IP address type
IPv4
Load balancer
Internal-App-ALB
4
Total targets
4
Healthy
0 Anomalous
0
Unhealthy
0
Unused
0
Initial
0
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags
Registered targets (4) Info
Anomaly mitigation: Not applicable
Deregister
Register targets
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.
Filter targets
Instance ID
Name
Port
Zone
Health status
Health status details
Administrative o...
Override details
Launch...
Anomaly
i-02952614ab554eaf5
AppServer-ASG
5000
us-east-1a (us...
Healthy
No override
No override is curre...
August 13...
i-0e48ce564f2e5036f
AppServer-ASG
5000
us-east-1b (us...
Healthy
No override
No override is curre...
August 13...
i-0eebed2302f8c9c14
AppServer-ASG
5000
us-east-1b (us...
Healthy
No override
No override is curre...
August 13...
i-0d0b993f00165b020
AppServer-ASG
5000
us-east-1a (us...
Healthy
No override
No override is curre...
August 13...


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#TargetGroup:targetGroupArn=arn:aws:elasticloadbalancing:us-east-1:265753586481:targetgroup/WebServerTG/72ec628e8ab4a863
EC2 > Target groups > WebServerTG
1 Total targets
1 Healthy
0 Unhealthy
0 Unused
0 Initial
0 Draining
0 Anomalous
Distribution of targets by Availability Zone (AZ)
Targets
Monitoring
Health checks
Attributes
Tags
Registered targets (1)
Anomaly mitigation: Not applicable
Deregister
Register targets
Instance ID
Name
Port
Zone
Health status
Health status details
i-0e80f3a597fe2f18b
WebServer
80
us-east-1a (us...
Healthy

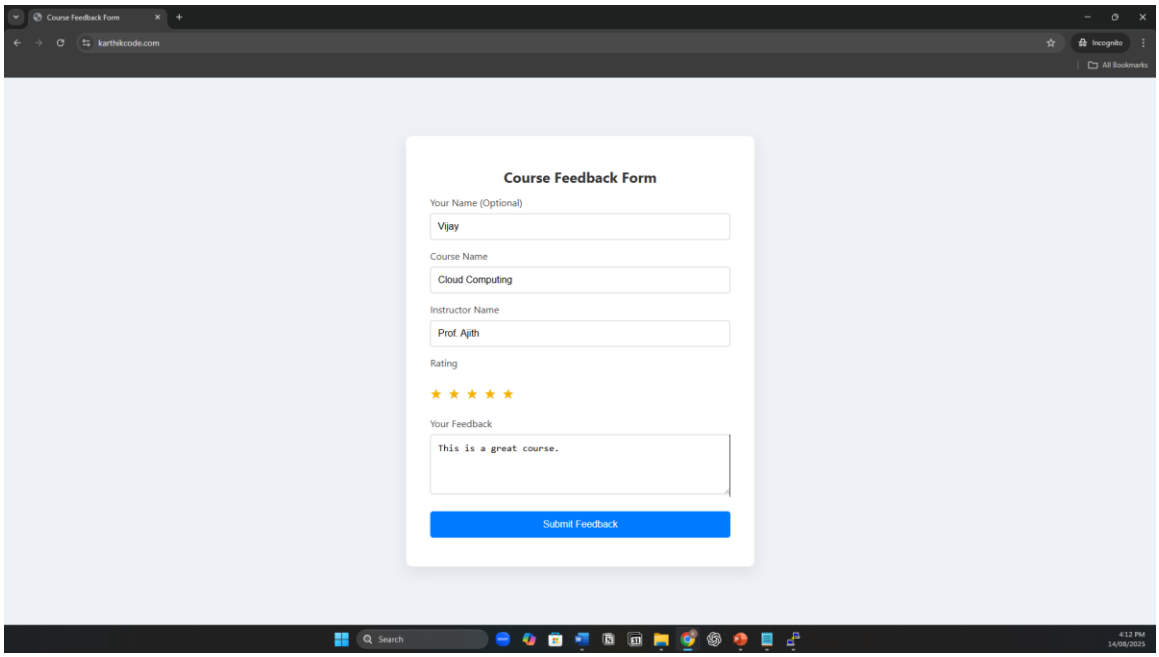

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroupDetails:id=web-asg;view=activity
EC2 > Auto Scaling groups > web-asg
Create notification
Activity history (3)
Status
Description
Cause
Start time
Successful
Launching a new EC2 instance: i-00858754ae469927a
At 2025-08-14T20:05:58Z a user request update of AutoScalingGroup constraints to min: 1, max: 4, desired: 1 changing the desired capacity from 0 to 1. At 2025-08-14T20:06:07Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 1.
2025 August 14, 04:06:09 PM -04:00
Successful
Terminating EC2 instance: i-0130b4a6345518056
At 2025-08-14T03:22:38Z a user request update of AutoScalingGroup constraints to min: 0, max: 0, desired: 0 changing the desired capacity from 1 to 0. At 2025-08-14T03:22:47Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 1 to 0. At 2025-08-14T03:22:47Z instance i-0130b4a6345518056 was selected for termination.
2025 August 13, 11:22:47 PM -04:00
Successful
Launching a new EC2 instance: i-0130b4a6345518056
At 2025-08-13T19:19:57Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 1. At 2025-08-13T19:20:01Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 1.
2025 August 13, 03:20:03 PM -04:00

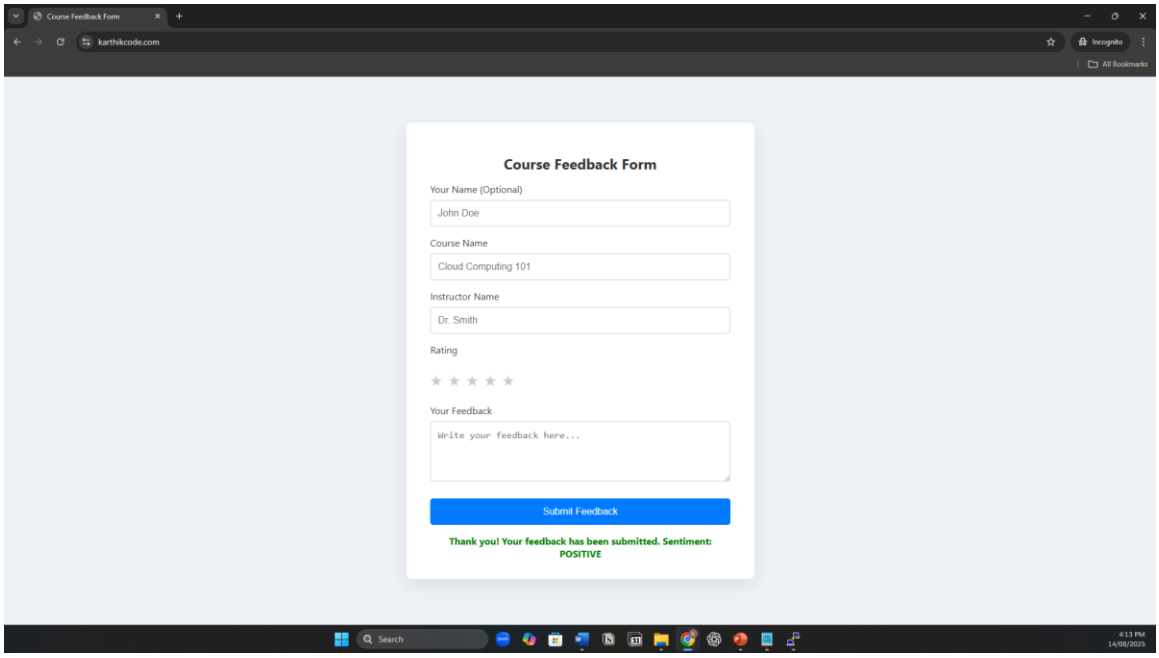

Gmail
Search mail
Compose
Inbox 1,997
Starred
Snoozed
Sent
Drafts
More
Labels
ALARM: "Warning-ASG-MaxCapacityReached" in US East (N. Virginia)
asg-cpu-alerts
You are receiving this email because your Amazon CloudWatch Alarm "Warning-ASG-MaxCapacityReached" in the US East (N. Virginia) region has entered the ALARM sta
10:45 AM (8 hours ago)
asg-cpu-alerts <no-reply@sns.amazonaws.com>
6:56 PM (4 minutes ago)
to me
You are receiving this email because your Amazon CloudWatch Alarm "Warning-ASG-MaxCapacityReached" in the US East (N. Virginia) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [4.0 (13/08/25 22:55:00)] was greater than or equal to the threshold (4.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Wednesday 13 August, 2025 22:56:01 UTC".
View this alarm in the AWS Management Console:
https://us-east-1.console.aws.amazon.com/cloudwatch/deeplink.js?region=us-east-1#alarmsV2:alarm/Warning-ASG-MaxCapacityReached
Alarm Details:
- Name: Warning-ASG-MaxCapacityReached
- Description: **app-asg DesiredCapacity >= 4 (Max).**
Auto Scaling Capacity Alert — app-asg
The Auto Scaling Group app-asg has reached its configured maximum capacity of 4 instances.
No further scale-out actions can occur, which may impact performance if demand continues to rise.
Recommended actions:
• Review current load and CPU utilization
• Check application performance and latency
• Consider increasing MaxCapacity or optimizing workloads
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [4.0 (13/08/25 22:55:00)] was greater than or equal to the threshold (4.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Wednesday 13 August, 2025 22:56:01 UTC
Reply
Forward
Enable desktop notifications for Gmail.
OK
No thanks

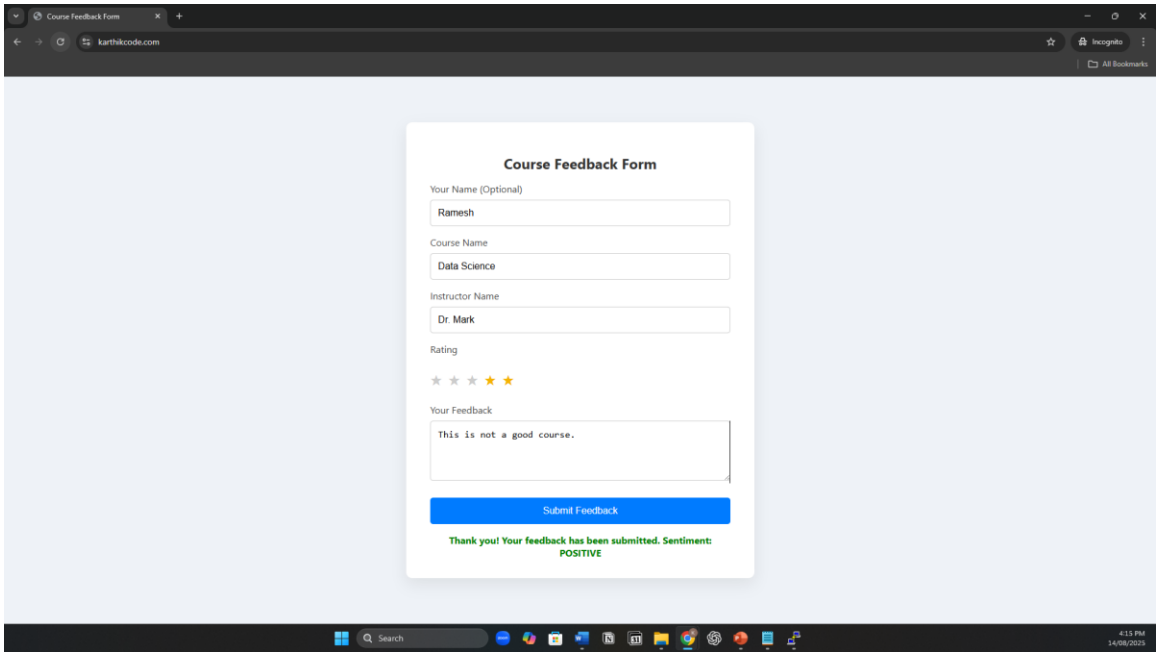

CloudWatch
Overview
Favorites and recents
Dashboards
AI Operations
Overview
Investigations
Configuration
Alarms
In alarm
All alarms
Billing
Logs
Log groups
Log Anomalies
Live Tail
Logs Insights
Contributor Insights
Metrics
All metrics
Explorer
Streams
Application Signals
Alarms by AWS service
Services
In alarm 1
Insufficient data 0
OK 0
AutoScaling
Recent alarms
View recent alarms dashboard
Warning-ASG-MaxCapacityReached
No unit
GroupDesiredCapacity >= 4 for 1 datapoints within ...
GroupDesiredCapacity
Default Dashboard
Name any CloudWatch dashboard CloudWatch-Default to display it here. Create a new default dashboard
Application Insights
Get started with Application Insights
Configure Application Insights
Set up monitors and dashboards to detect issues and resolve problems with enterprise applications, databases, and workloads.
How it works
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

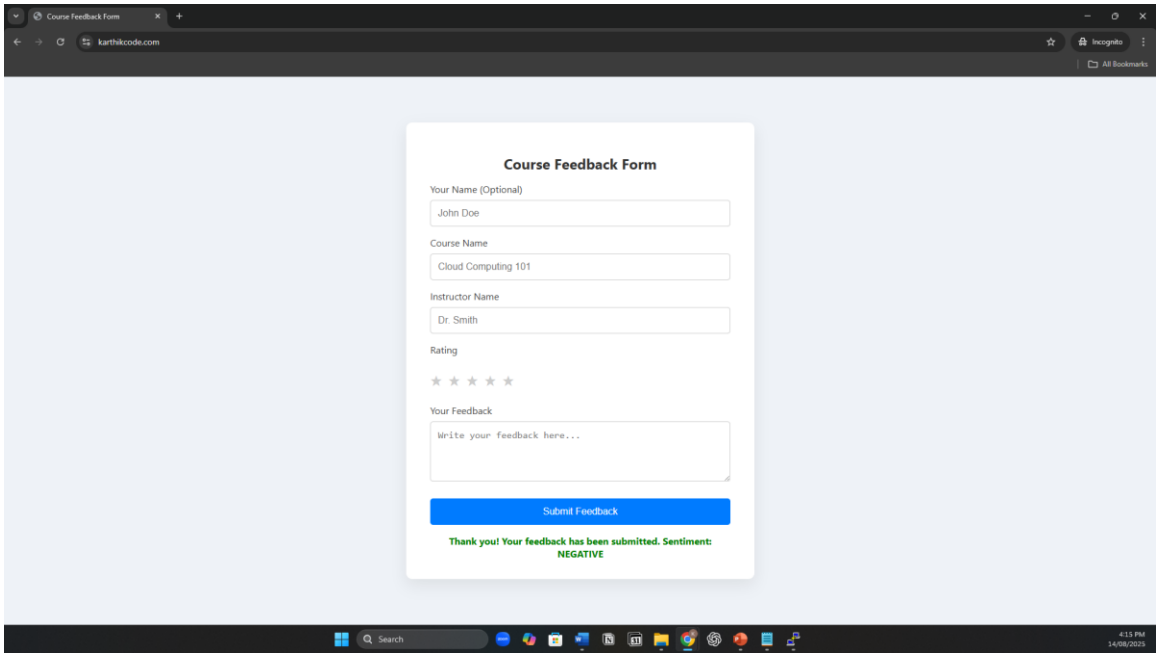

AutoScaling
GroupInServiceInstances: Average
GroupDesiredCapacity: Average
GroupPendingInstances: Average
GroupStandbyInstances: Average
GroupTerminatingInstances: Average
GroupMinSize: Average
GroupMaxSize: Average
GroupTotalInstances: Average

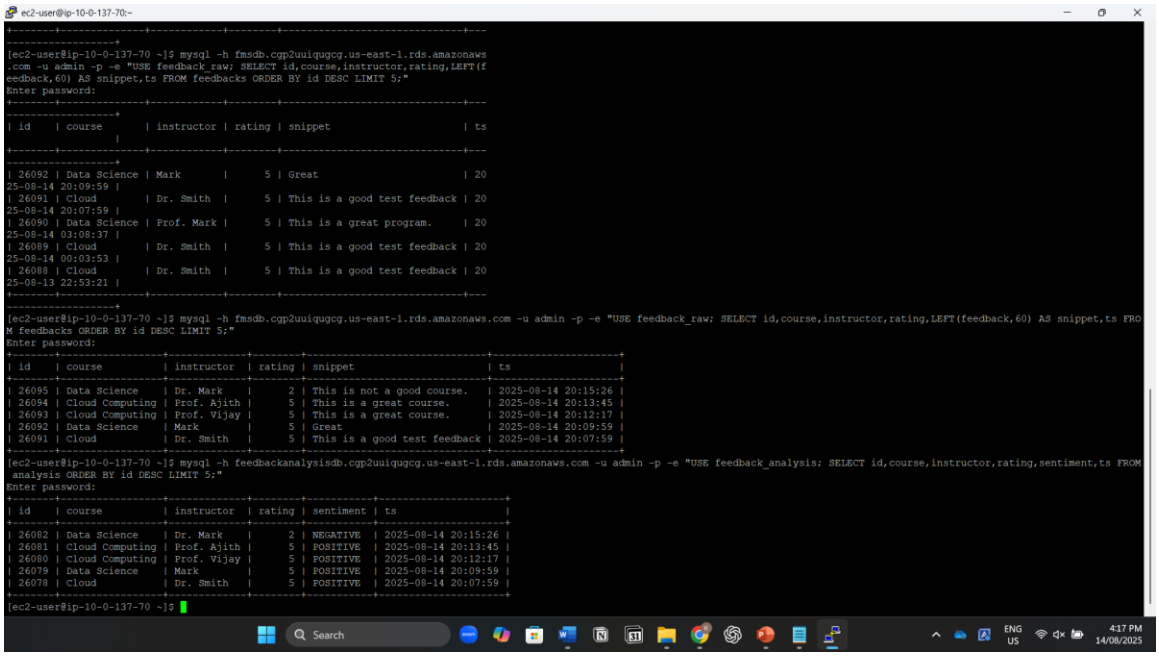
Web Page


Course Feedback Form
Your Name (Optional)
Vijay
Course Name
Cloud Computing
Instructor Name
Prof. Ajith
Rating
Your Feedback
This is a great course.
Submit Feedback


Course Feedback Form
karthikcode.com
Incognito
All Bookmarks
Course Feedback Form
Your Name (Optional)
John Doe
Course Name
Cloud Computing 101
Instructor Name
Dr. Smith
Rating
Your Feedback
Write your feedback here...
Submit Feedback
Thank you! Your feedback has been submitted. Sentiment: POSITIVE
Search
4:13 PM
14/08/2025


Course Feedback Form
karthikcode.com
Incognito
All Bookmarks
Course Feedback Form
Your Name (Optional)
Ramesh
Course Name
Data Science
Instructor Name
Dr. Mark
Rating
Your Feedback
This is not a good course.
Submit Feedback
Thank you! Your feedback has been submitted. Sentiment: POSITIVE
Search
4:15 PM
14/08/2025


Course Feedback Form
karthikcode.com
Incognito
All Bookmarks
Course Feedback Form
Your Name (Optional)
John Doe
Course Name
Cloud Computing 101
Instructor Name
Dr. Smith
Rating
Your Feedback
Write your feedback here...
Submit Feedback
Thank you! Your feedback has been submitted. Sentiment: NEGATIVE
Search
4:15 PM
14/08/2025


ec2-user@ip-10-0-137-70:~
[ec2-user@ip-10-0-137-70 ~]$ mysql -h fmsdb.cgp2uuiqugcg.us-east-1.rds.amazonaws
.com -u admin -p -e "USE feedback_raw; SELECT id,course,instructor,rating,LEFT(f
eedback,60) AS snippet,ts FROM feedbacks ORDER BY id DESC LIMIT 5;"
Enter password:
id	course	instructor	rating	snippet	ts
26092	Data Science	Mark	5	Great	2025-08-14 20:09:59
26091	Cloud	Dr. Smith	5	This is a good test feedback	2025-08-14 20:07:59
26090	Data Science	Prof. Mark	5	This is a great program.	2025-08-14 03:08:37
26089	Cloud	Dr. Smith	5	This is a good test feedback	2025-08-14 00:03:53
26088	Cloud	Dr. Smith	5	This is a good test feedback	2025-08-13 22:53:21
[ec2-user@ip-10-0-137-70 ~]$ mysql -h fmsdb.cgp2uuiqugcg.us-east-1.rds.amazonaws.com -u admin -p -e "USE feedback_raw; SELECT id,course,instructor,rating,LEFT(feedback,60) AS snippet,ts FRO
M feedbacks ORDER BY id DESC LIMIT 5;"
Enter password:
id	course	instructor	rating	snippet	ts
26095	Data Science	Dr. Mark	2	This is not a good course.	2025-08-14 20:15:26
26094	Cloud Computing	Prof. Ajith	5	This is a great course.	2025-08-14 20:13:45
26093	Cloud Computing	Prof. Vijay	5	This is a great course.	2025-08-14 20:12:17
26092	Data Science	Mark	5	Great	2025-08-14 20:09:59
26091	Cloud	Dr. Smith	5	This is a good test feedback	2025-08-14 20:07:59
[ec2-user@ip-10-0-137-70 ~]$ mysql -h feedbackanalysisdb.cgp2uuiqugcg.us-east-1.rds.amazonaws.com -u admin -p -e "USE feedback_analysis; SELECT id,course,instructor,rating,sentiment,ts FROM
 analysis ORDER BY id DESC LIMIT 5;"
Enter password:
id	course	instructor	rating	sentiment	ts
26082	Data Science	Dr. Mark	2	NEGATIVE	2025-08-14 20:15:26
26081	Cloud Computing	Prof. Ajith	5	POSITIVE	2025-08-14 20:13:45
26080	Cloud Computing	Prof. Vijay	5	POSITIVE	2025-08-14 20:12:17
26079	Data Science	Mark	5	POSITIVE	2025-08-14 20:09:59
26078	Cloud	Dr. Smith	5	POSITIVE	2025-08-14 20:07:59
[ec2-user@ip-10-0-137-70 ~]$
Search
ENG
US
4:17 PM
14/08/2025

Conclusion

The AWS Feedback Management System demonstrates how a modern, cloud-native application can combine scalability, security, and intelligence to deliver real business value. By leveraging a three-tier architecture with Auto Scaling Groups, the system adapts seamlessly to changes in traffic, always ensuring consistent performance while optimizing costs.

Integration with AWS Comprehend enables real-time sentiment analysis, transforming raw user feedback into actionable insights. Amazon QuickSight provides interactive dashboards that make it easy to monitor trends and identify areas for improvement. The use of Amazon SNS for Auto Scaling notifications ensures that operational teams are immediately informed of scaling events, while Amazon CloudWatch maintains continuous visibility into the system's health.