DURHAM
COLLEGE
SUCCESS MATTERS

**Cloud Computing**

**Final Exam**

**Course: COSC-1209**

**Shavez Azmi**

**Student ID: 100938606**

# Implementing DevOps solutions Test #2

A few months ago, you were hired by Wild Rydes to develop a highly scalable, highly redundant website hosted on AWS. After testing your deployment, the CTO has decided to bring you back in to help convert their entire infrastructure stack into Infrastructure as Code (IaC).

As Wild Rydes is an AWS native organization, they've requested you develop the entire IaC script using AWS CloudFormation. They have provided a description of their infrastructure below.

**Note**: it is not a requirement to deploy your scripts to an AWS account however you may wish to do so when testing your solution.

Wild Rydes currently makes use of ECS to run their monolithic application. The application is containerized using docker and runs on ECS Fargate. The ECS service is scalable based on load and runs in two separate subnets behind an application load balancer.

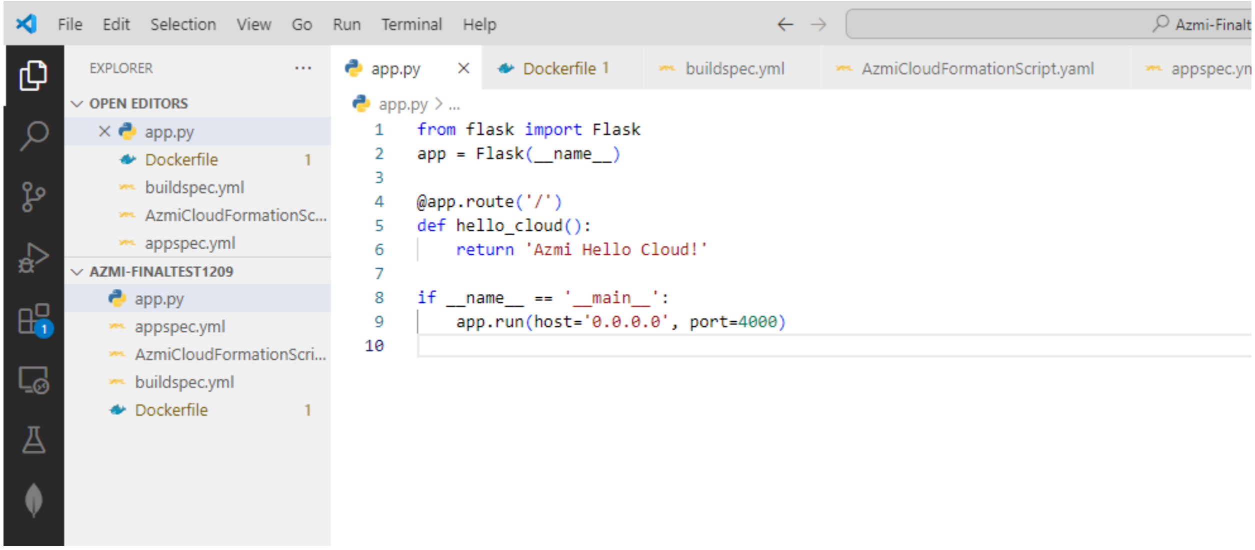
The application is updated via a CI/CD CodePipeline that pulls from a GitHub repository, builds, and pushes the image based on the docker file, and updates the ECS service with the new image. Wild Rydes has alarms set at each stage of the pipeline to determine build failures or an overall successful deployment.

Using the above documentation, determine what resources need to be deployed and create a CloudFormation script that will deploy them.

**Ensure you submit your complete CloudFormation script(s) before the end of the submission period.**

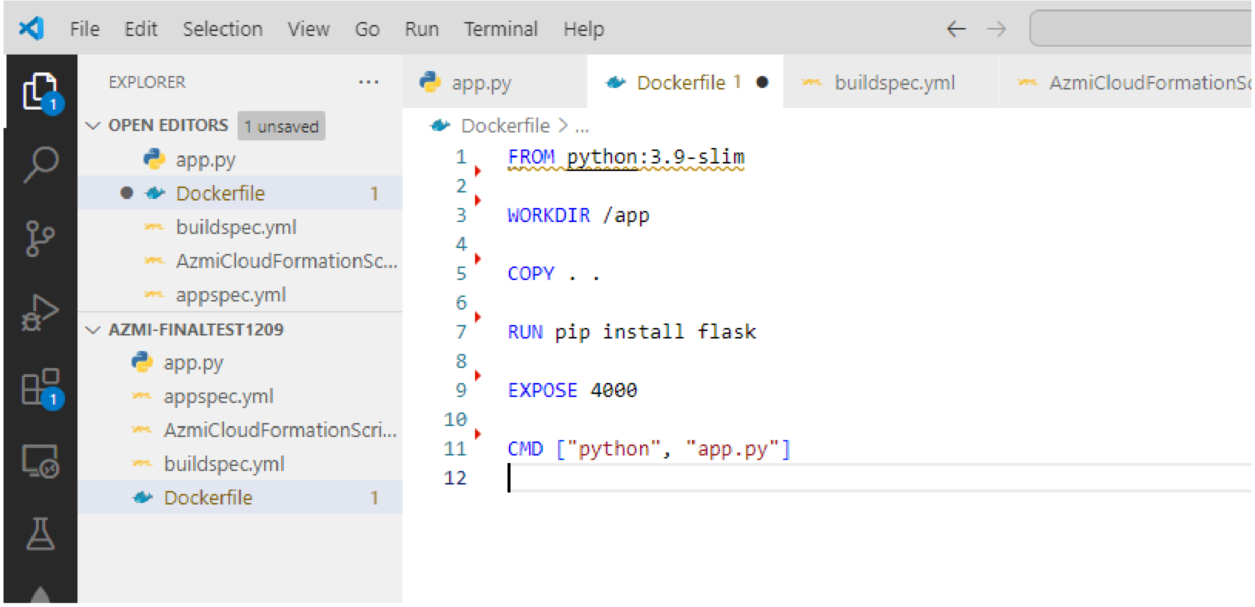
## Files Created

### App.py



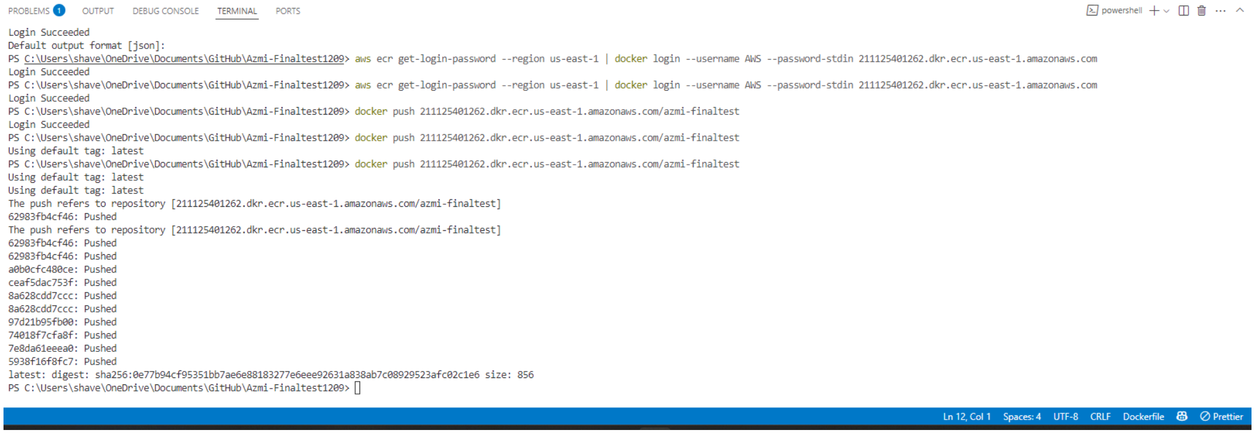
```python
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_cloud():
    return 'Azmi Hello Cloud!'

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=4000)
```

### DockerFile



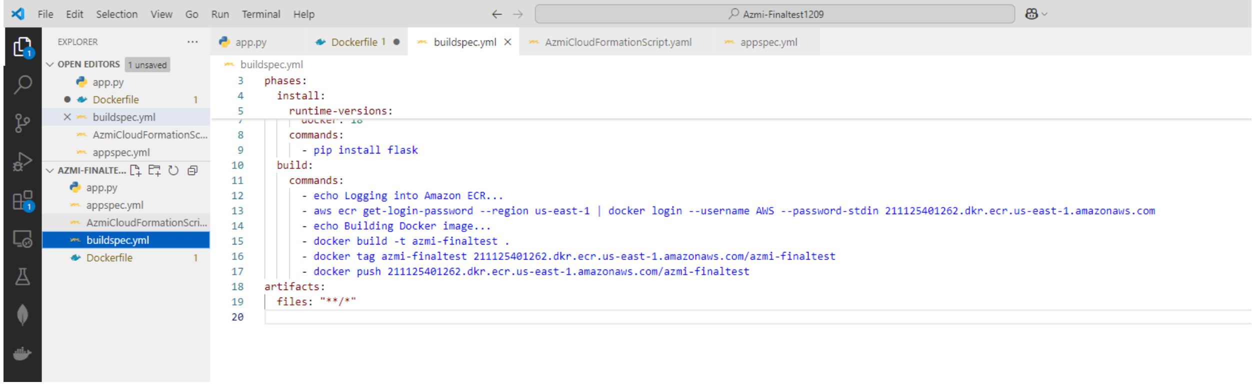
```dockerfile
FROM python:3.9-slim

WORKDIR /app

COPY . .

RUN pip install flask

EXPOSE 4000

CMD ["python", "app.py"]
```

## DockerImage Push



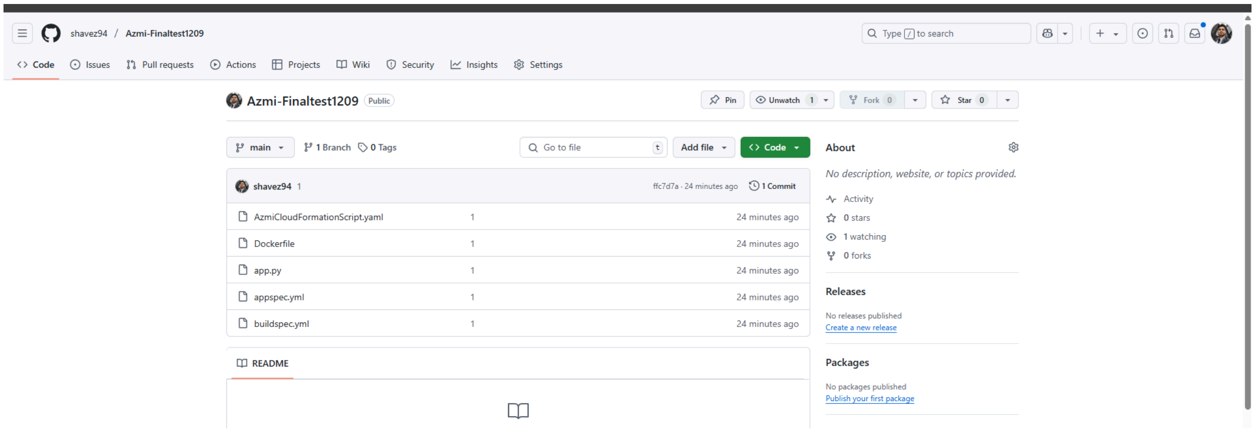
```
Login Succeeded
Default output format [json]:
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 211125401262.dkr.ecr.us-east-1.amazonaws.com
Login Succeeded
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 211125401262.dkr.ecr.us-east-1.amazonaws.com
Login Succeeded
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> docker push 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
Login Succeeded
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> docker push 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
Using default tag: latest
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> docker push 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
Using default tag: latest
Using default tag: latest
The push refers to repository [211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest]
62983fb4cf46: Pushed
The push refers to repository [211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest]
62983fb4cf46: Pushed
62983fb4cf46: Pushed
a0b0cfc480ce: Pushed
ceaf5dac753f: Pushed
8a628cdd7ccc: Pushed
8a628cdd7ccc: Pushed
97d21b95fb00: Pushed
74018f7cfa8f: Pushed
7e8da61eeea0: Pushed
5938f16f8fc7: Pushed
latest: digest: sha256:0e77b94cf95351bb7ae6e88183277e6eee92631a838ab7c08929523afc02c1e6 size: 856
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> 
```

## BuildSpec.yml



```
buildspec.yml
phases:
  install:
    runtime-versions:
      docker: 18
    commands:
      - pip install flask
  build:
    commands:
      - echo Logging into Amazon ECR...
      - aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 211125401262.dkr.ecr.us-east-1.amazonaws.com
      - echo Building Docker image...
      - docker build -t azmi-finaltest .
      - docker tag azmi-finaltest 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
      - docker push 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
artifacts:
  files: "**/*"
```

## Github Repo



Azmi-Finaltest1209 Public

shavez94 1 ffc7d7a · 24 minutes ago 1 Commit

| | | |
|---|---|---|
| AzmiCloudFormationScript.yaml | 1 | 24 minutes ago |
| Dockerfile | 1 | 24 minutes ago |
| app.py | 1 | 24 minutes ago |
| appspec.yml | 1 | 24 minutes ago |
| buildspec.yml | 1 | 24 minutes ago |

About

*No description, website, or topics provided.*

Activity
0 stars
1 watching
0 forks

Releases

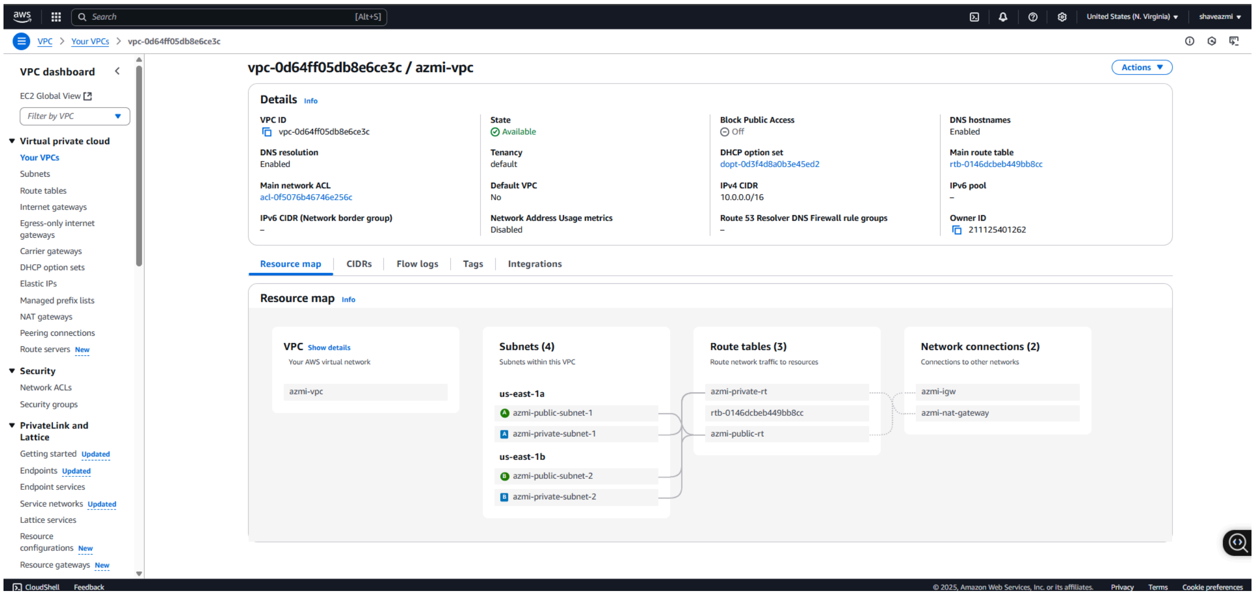
No releases published
Create a new release

Packages

No packages published
Publish your first package

# Resources Created

## VPC



vpc-0d64ff05db8e6ce3c / azmi-vpc

**Details** Info

| | | | |
|---|---|---|---|
| VPC ID<br>vpc-0d64ff05db8e6ce3c | State<br>Available | Block Public Access<br>Off | DNS hostnames<br>Enabled |
| DNS resolution<br>Enabled | Tenancy<br>default | DHCP option set<br>dopt-0d3f4d8a0b3e45ed2 | Main route table<br>rtb-0146dcbeb449bb8cc |
| Main network ACL<br>acl-0f5076b46746e256c | Default VPC<br>No | IPv4 CIDR<br>10.0.0.0/16 | IPv6 pool<br>– |
| IPv6 CIDR (Network border group)<br>– | Network Address Usage metrics<br>Disabled | Route 53 Resolver DNS Firewall rule groups<br>– | Owner ID<br>211125401262 |

**Resource map** Info

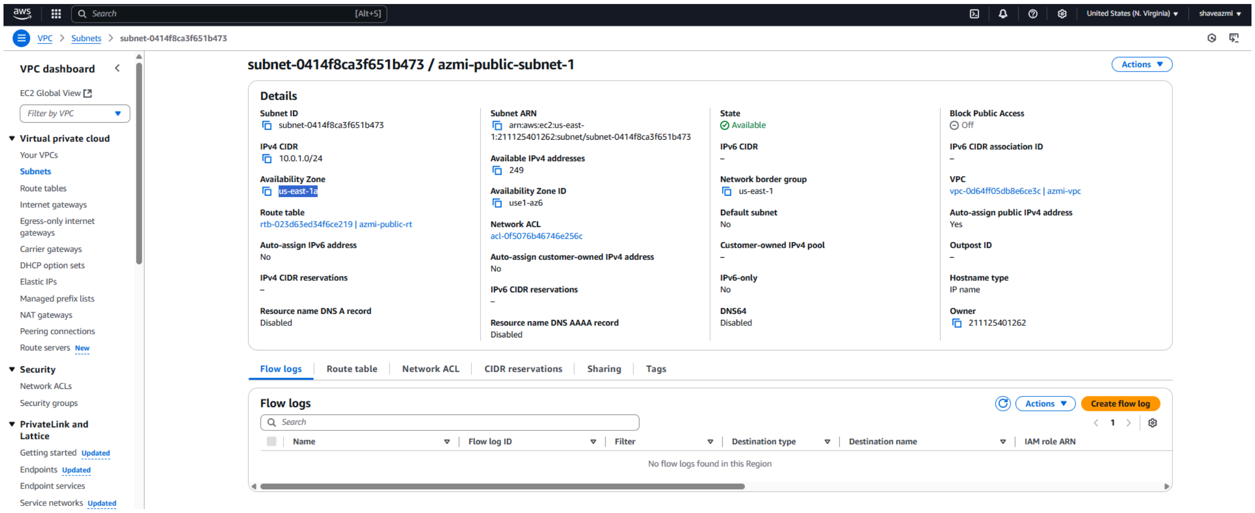
VPC Show details
Your AWS virtual network
azmi-vpc

Subnets (4)
Subnets within this VPC
us-east-1a
azmi-public-subnet-1
azmi-private-subnet-1
us-east-1b
azmi-public-subnet-2
azmi-private-subnet-2

Route tables (3)
Route network traffic to resources
azmi-private-rt
rtb-0146dcbeb449bb8cc
azmi-public-rt

Network connections (2)
Connections to other networks
azmi-igw
azmi-nat-gateway

## Public Subnet1



subnet-0414f8ca3f651b473 / azmi-public-subnet-1

**Details**

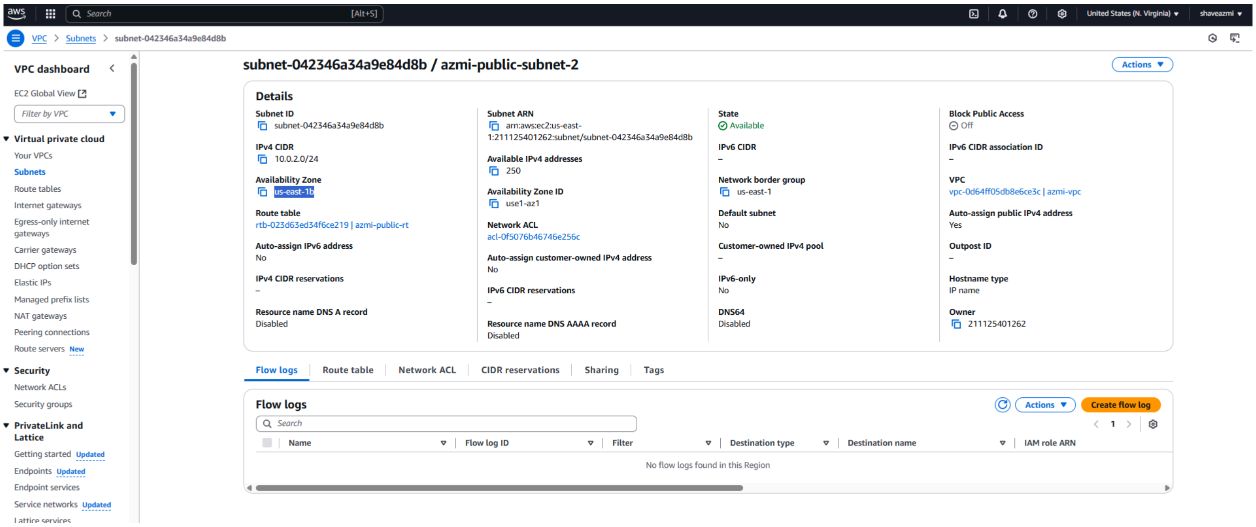
| | | | |
|---|---|---|---|
| Subnet ID<br>subnet-0414f8ca3f651b473 | Subnet ARN<br>arn:aws:ec2:us-east-1:211125401262:subnet/subnet-0414f8ca3f651b473 | State<br>Available | Block Public Access<br>Off |
| IPv4 CIDR<br>10.0.1.0/24 | Available IPv4 addresses<br>249 | IPv6 CIDR<br>– | IPv6 CIDR association ID<br>– |
| Availability Zone<br>us-east-1a | Availability Zone ID<br>use1-az6 | Network border group<br>us-east-1 | VPC<br>vpc-0d64ff05db8e6ce3c \| azmi-vpc |
| Route table<br>rtb-023d63ed34f6ce219 \| azmi-public-rt | Network ACL<br>acl-0f5076b46746e256c | Default subnet<br>No | Auto-assign public IPv4 address<br>Yes |
| Auto-assign IPv6 address<br>No | Auto-assign customer-owned IPv4 address<br>No | Customer-owned IPv4 pool<br>– | Outpost ID<br>– |
| IPv4 CIDR reservations<br>– | IPv6 CIDR reservations<br>– | IPv6-only<br>No | Hostname type<br>IP name |
| Resource name DNS A record<br>Disabled | Resource name DNS AAAA record<br>Disabled | DNS64<br>Disabled | Owner<br>211125401262 |

**Flow logs**

| Name | Flow log ID | Filter | Destination type | Destination name | IAM role ARN |
|---|---|---|---|---|---|

No flow logs found in this Region

# Public Subnet2



## subnet-042346a34a9e84d8b / azmi-public-subnet-2

### Details

**Subnet ID**
subnet-042346a34a9e84d8b

**IPv4 CIDR**
10.0.2.0/24

**Availability Zone**
us-east-1b

**Route table**
rtb-023d63ed34f6ce219 | azmi-public-rt

**Auto-assign IPv6 address**
No

**IPv4 CIDR reservations**
–

**Resource name DNS A record**
Disabled

**Subnet ARN**
arn:aws:ec2:us-east-1:211125401262:subnet/subnet-042346a34a9e84d8b

**Available IPv4 addresses**
250

**Availability Zone ID**
use1-az1

**Network ACL**
acl-0f5076b46746e256c

**Auto-assign customer-owned IPv4 address**
No

**IPv6 CIDR reservations**
–

**Resource name DNS AAAA record**
Disabled

**State**
Available

**IPv6 CIDR**
–

**Network border group**
us-east-1

**Default subnet**
No

**Customer-owned IPv4 pool**
–

**IPv6-only**
No

**DNS64**
Disabled

**Block Public Access**
Off

**IPv6 CIDR association ID**
–

**VPC**
vpc-0d64ff05db8e6ce3c | azmi-vpc

**Auto-assign public IPv4 address**
Yes

**Outpost ID**
–

**Hostname type**
IP name

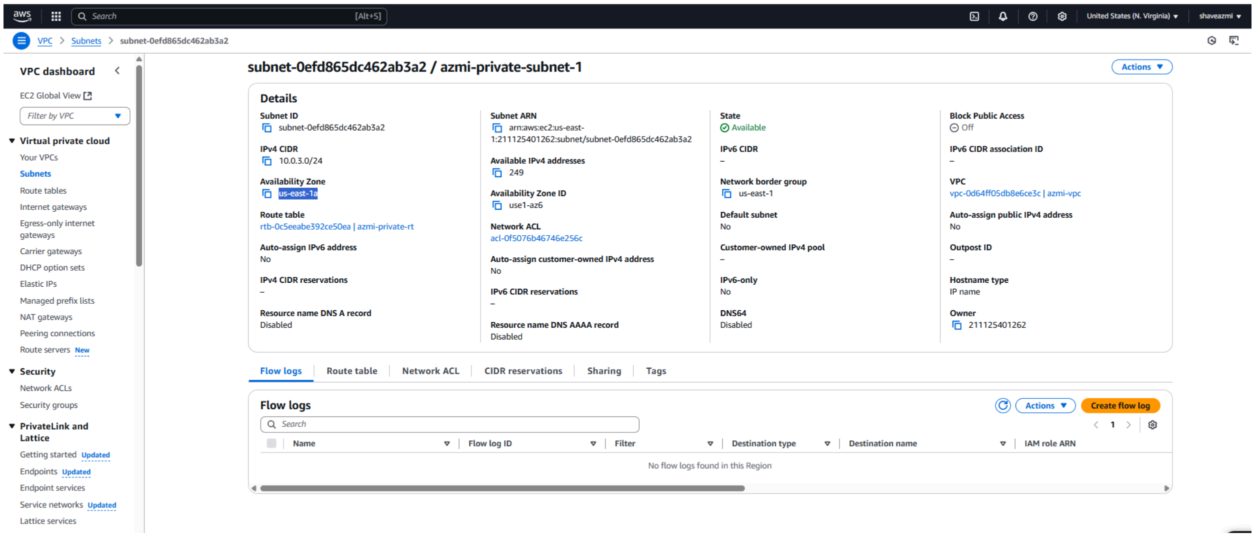
**Owner**
211125401262

### Flow logs

| □ | Name | Flow log ID | Filter | Destination type | Destination name | IAM role ARN |
|---|---|---|---|---|---|---|

No flow logs found in this Region

# Private Subnet1



## subnet-0efd865dc462ab3a2 / azmi-private-subnet-1

### Details

**Subnet ID**
subnet-0efd865dc462ab3a2

**IPv4 CIDR**
10.0.3.0/24

**Availability Zone**
us-east-1a

**Route table**
rtb-0c5eeabe392ce50ea | azmi-private-rt

**Auto-assign IPv6 address**
No

**IPv4 CIDR reservations**
–

**Resource name DNS A record**
Disabled

**Subnet ARN**
arn:aws:ec2:us-east-1:211125401262:subnet/subnet-0efd865dc462ab3a2

**Available IPv4 addresses**
249

**Availability Zone ID**
use1-az6

**Network ACL**
acl-0f5076b46746e256c

**Auto-assign customer-owned IPv4 address**
No

**IPv6 CIDR reservations**
–

**Resource name DNS AAAA record**
Disabled

**State**
Available

**IPv6 CIDR**
–

**Network border group**
us-east-1

**Default subnet**
No

**Customer-owned IPv4 pool**
–

**IPv6-only**
No

**DNS64**
Disabled

**Block Public Access**
Off

**IPv6 CIDR association ID**
–

**VPC**
vpc-0d64ff05db8e6ce3c | azmi-vpc

**Auto-assign public IPv4 address**
No

**Outpost ID**
–

**Hostname type**
IP name

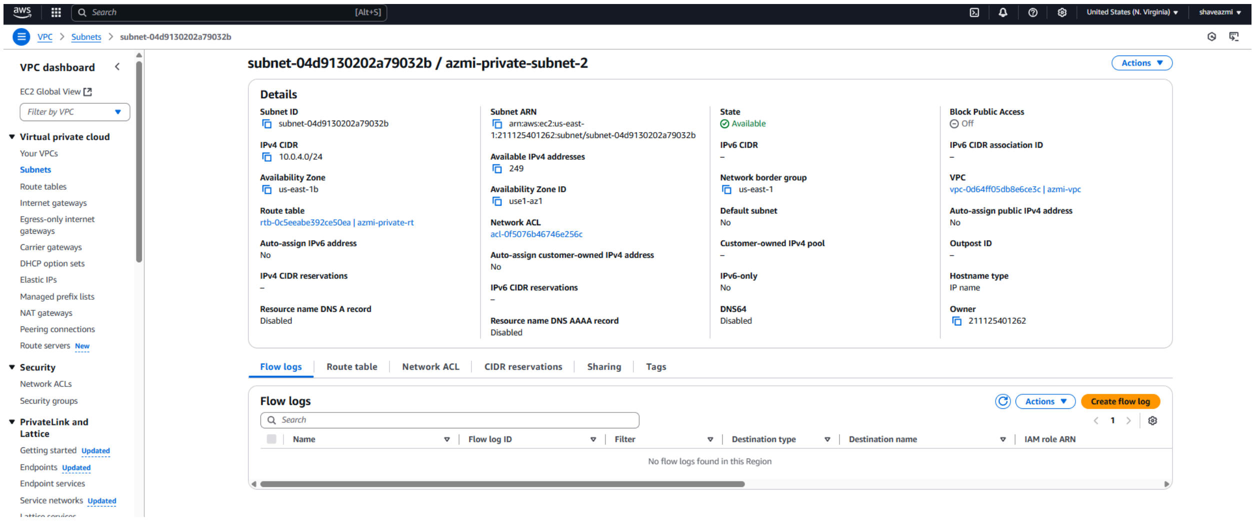
**Owner**
211125401262

### Flow logs

| □ | Name | Flow log ID | Filter | Destination type | Destination name | IAM role ARN |
|---|---|---|---|---|---|---|

No flow logs found in this Region

# PrivateSubnet2



subnet-04d9130202a79032b / azmi-private-subnet-2

Details

Subnet ID
subnet-04d9130202a79032b

IPv4 CIDR
10.0.4.0/24

Availability Zone
us-east-1b

Route table
rtb-0c5eeabe392ce50ea | azmi-private-rt

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

Resource name DNS A record
Disabled

Subnet ARN
arn:aws:ec2:us-east-1:211125401262:subnet/subnet-04d9130202a79032b

Available IPv4 addresses
249

Availability Zone ID
use1-az1

Network ACL
acl-0f5076b46746e256c

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

Resource name DNS AAAA record
Disabled

State
Available

IPv6 CIDR
–

Network border group
us-east-1

Default subnet
No

Customer-owned IPv4 pool
–

IPv6-only
No

DNS64
Disabled

Block Public Access
Off

IPv6 CIDR association ID
–

VPC
vpc-0d64ff05db8e6ce3c | azmi-vpc

Auto-assign public IPv4 address
No

Outpost ID
–

Hostname type
IP name

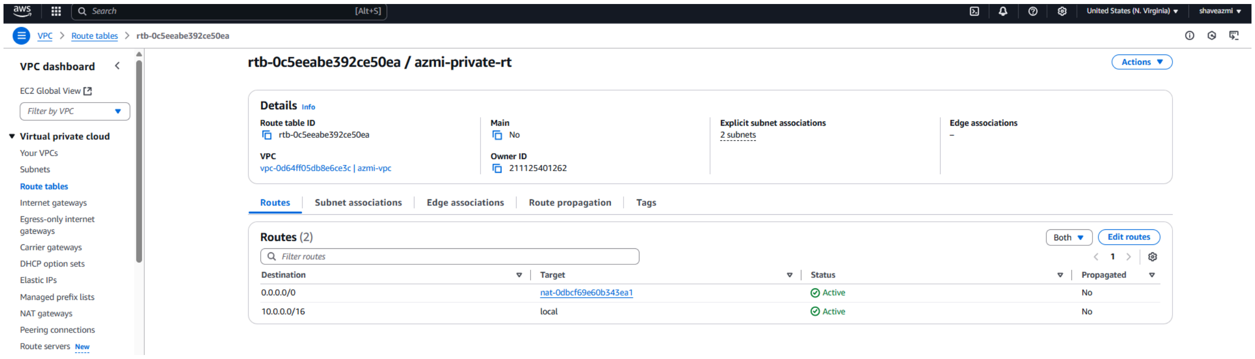
Owner
211125401262

Flow logs

| Name | Flow log ID | Filter | Destination type | Destination name | IAM role ARN |
|---|---|---|---|---|---|

No flow logs found in this Region

# PrivateRouteTable



rtb-0c5eeabe392ce50ea / azmi-private-rt

Details Info

Route table ID
rtb-0c5eeabe392ce50ea

VPC
vpc-0d64ff05db8e6ce3c | azmi-vpc

Main
No

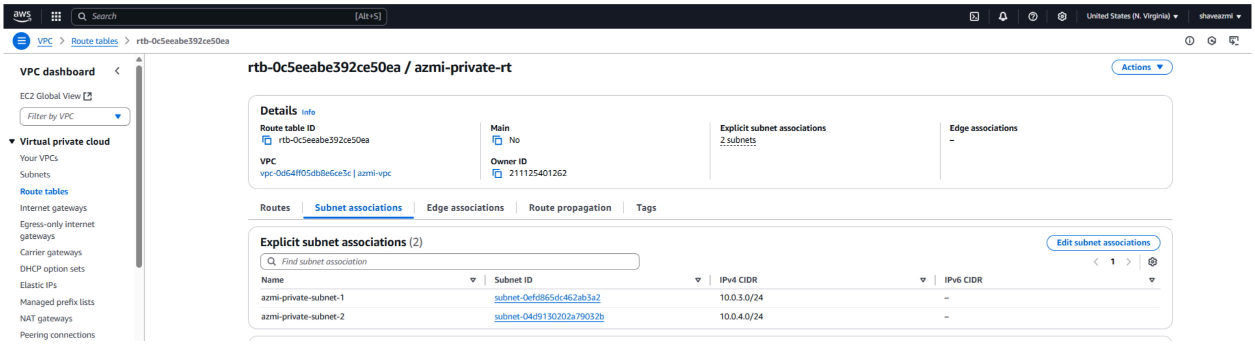
Owner ID
211125401262

Explicit subnet associations
2 subnets

Edge associations
–

Routes (2)

| Destination | Target | Status | Propagated |
|---|---|---|---|
| 0.0.0.0/0 | nat-0dbcf69e60b343ea1 | Active | No |
| 10.0.0.0/16 | local | Active | No |



rtb-0c5eeabe392ce50ea / azmi-private-rt

Details Info

Route table ID
rtb-0c5eeabe392ce50ea

VPC
vpc-0d64ff05db8e6ce3c | azmi-vpc

Main
No

Owner ID
211125401262

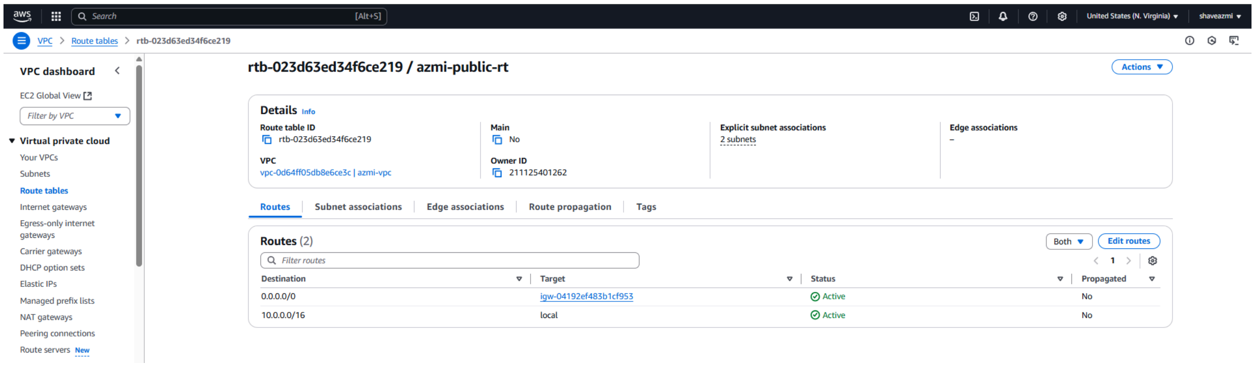
Explicit subnet associations
2 subnets

Edge associations
–

Explicit subnet associations (2)

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|---|---|---|---|
| azmi-private-subnet-1 | subnet-0efd865dc462ab3a2 | 10.0.3.0/24 | – |
| azmi-private-subnet-2 | subnet-04d9130202a79032b | 10.0.4.0/24 | – |

## PublicRouteTable



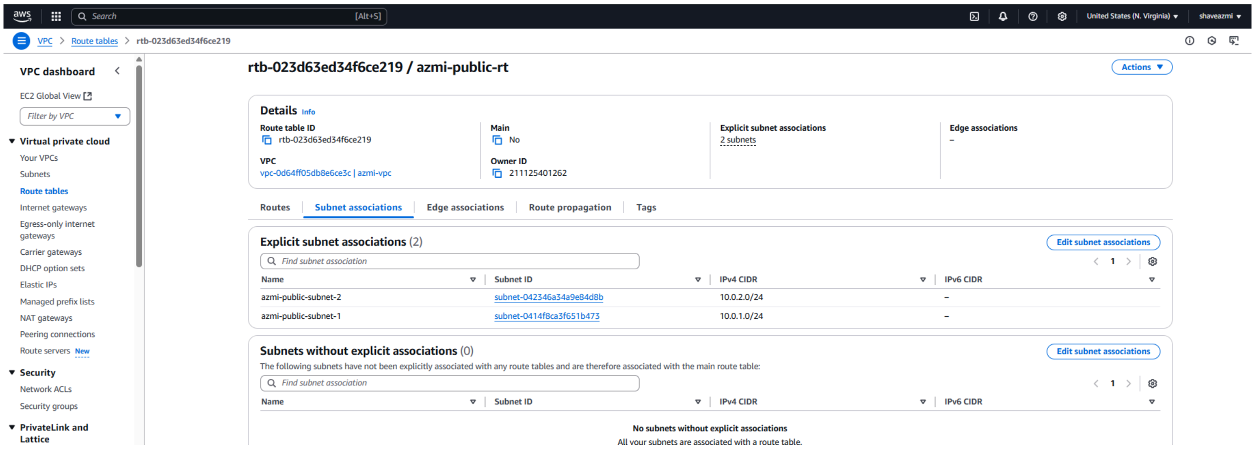
**rtb-023d63ed34f6ce219 / azmi-public-rt**

Details

Route table ID: rtb-023d63ed34f6ce219
Main: No
Explicit subnet associations: 2 subnets
Edge associations: –
VPC: vpc-0d64ff05db8e6ce3c | azmi-vpc
Owner ID: 211125401262

Routes (2)

| Destination | Target | Status | Propagated |
|---|---|---|---|
| 0.0.0.0/0 | igw-04192ef483b1cf953 | Active | No |
| 10.0.0.0/16 | local | Active | No |



**rtb-023d63ed34f6ce219 / azmi-public-rt**

Details

Route table ID: rtb-023d63ed34f6ce219
Main: No
Explicit subnet associations: 2 subnets
Edge associations: –
VPC: vpc-0d64ff05db8e6ce3c | azmi-vpc
Owner ID: 211125401262

Explicit subnet associations (2)

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|---|---|---|---|
| azmi-public-subnet-2 | subnet-042346a34a9e84d8b | 10.0.2.0/24 | – |
| azmi-public-subnet-1 | subnet-0414f8ca3f651b473 | 10.0.1.0/24 | – |

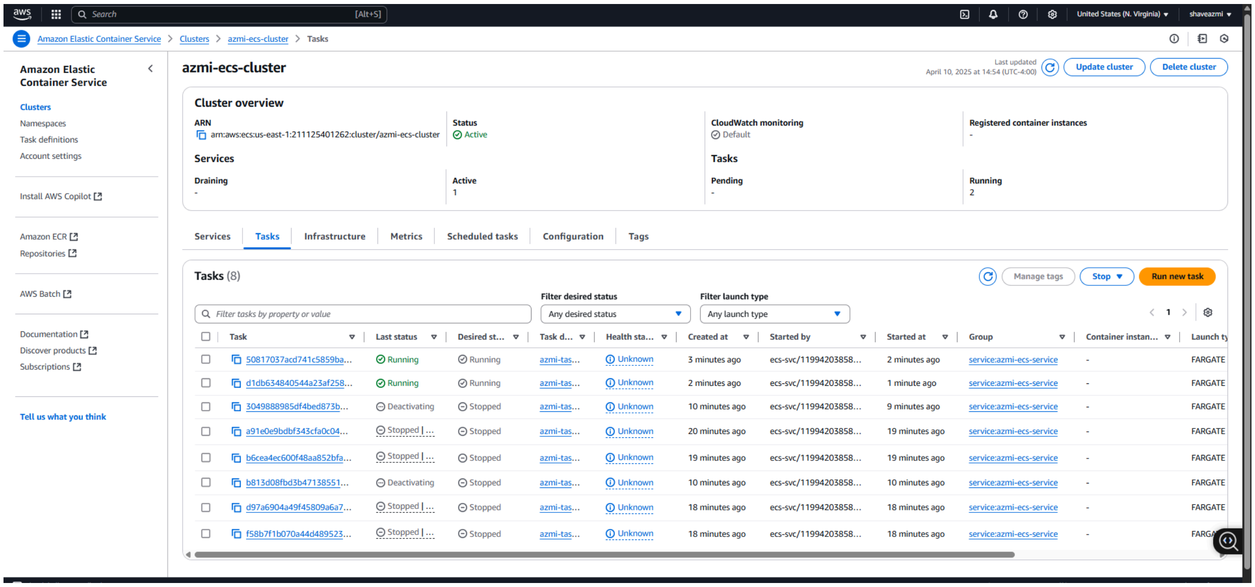
Subnets without explicit associations (0)

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|---|---|---|---|

No subnets without explicit associations
All your subnets are associated with a route table.

## ECSTaskExecutionRole



**azmi-ecs-cluster**

Last updated April 10, 2025 at 14:54 (UTC-4:00)

Cluster overview

ARN: arn:aws:ecs:us-east-1:211125401262:cluster/azmi-ecs-cluster
Status: Active
CloudWatch monitoring: Default
Registered container instances: -

Services
Draining: -
Active: 1

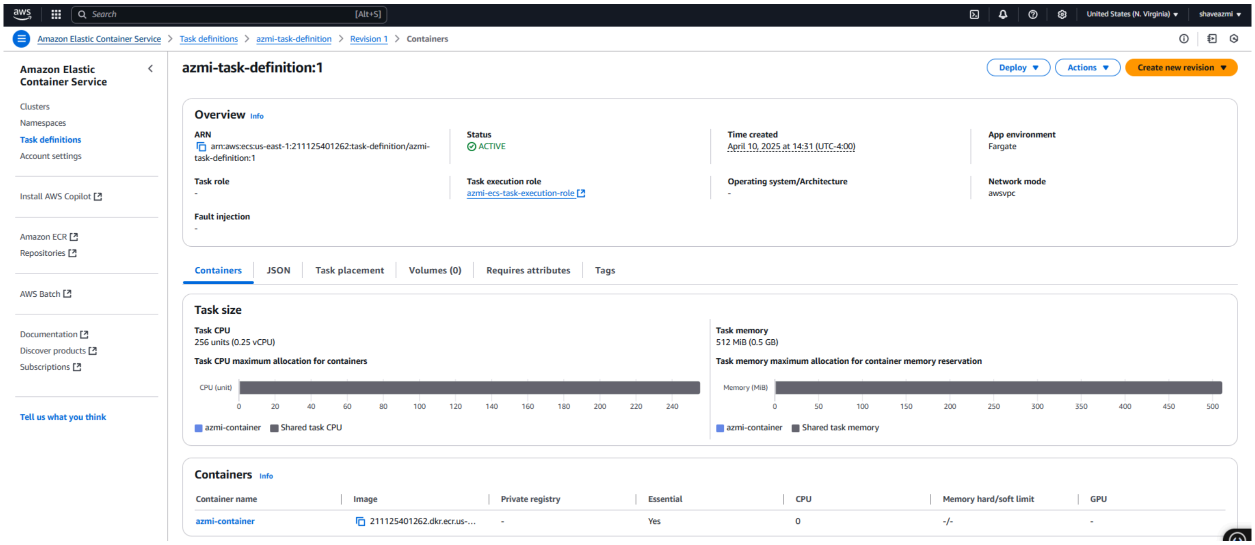
Tasks
Pending: -
Running: 2

Tasks (8)

Filter desired status: Any desired status
Filter launch type: Any launch type

| Task | Last status | Desired st... | Task d... | Health sta... | Created at | Started by | Started at | Group | Container instan... | Launch ty |
|---|---|---|---|---|---|---|---|---|---|---|
| 50817037acd741c5859ba... | Running | Running | azmi-tas... | Unknown | 3 minutes ago | ecs-svc/11994203858... | 2 minutes ago | service:azmi-ecs-service | - | FARGATE |
| d1db634840544a23af258... | Running | Running | azmi-tas... | Unknown | 2 minutes ago | ecs-svc/11994203858... | 1 minute ago | service:azmi-ecs-service | - | FARGATE |
| 3049888985df4bed873b... | Deactivating | Stopped | azmi-tas... | Unknown | 10 minutes ago | ecs-svc/11994203858... | 9 minutes ago | service:azmi-ecs-service | - | FARGATE |
| a91e0e9bdbf343cfa0c04... | Stopped \| ... | Stopped | azmi-tas... | Unknown | 20 minutes ago | ecs-svc/11994203858... | 19 minutes ago | service:azmi-ecs-service | - | FARGATE |
| b6cea4ec600f48aa852bfa... | Stopped \| ... | Stopped | azmi-tas... | Unknown | 19 minutes ago | ecs-svc/11994203858... | 19 minutes ago | service:azmi-ecs-service | - | FARGATE |
| b813d08fbd3b47138551... | Deactivating | Stopped | azmi-tas... | Unknown | 10 minutes ago | ecs-svc/11994203858... | 10 minutes ago | service:azmi-ecs-service | - | FARGATE |
| d97a6904a49f45809a6a7... | Stopped \| ... | Stopped | azmi-tas... | Unknown | 18 minutes ago | ecs-svc/11994203858... | 18 minutes ago | service:azmi-ecs-service | - | FARGATE |
| f58b7f1b070a44d489523... | Stopped \| ... | Stopped | azmi-tas... | Unknown | 18 minutes ago | ecs-svc/11994203858... | 18 minutes ago | service:azmi-ecs-service | - | FARGA |

# ECSTaskDefinition



## azmi-task-definition:1

### Overview Info

**ARN**
arn:aws:ecs:us-east-1:211125401262:task-definition/azmi-task-definition:1

**Status**
ACTIVE

**Time created**
April 10, 2025 at 14:31 (UTC-4:00)

**App environment**
Fargate

**Task role**
-

**Task execution role**
azmi-ecs-task-execution-role

**Operating system/Architecture**
-

**Network mode**
awsvpc

**Fault injection**
-

### Task size

**Task CPU**
256 units (0.25 vCPU)

**Task CPU maximum allocation for containers**

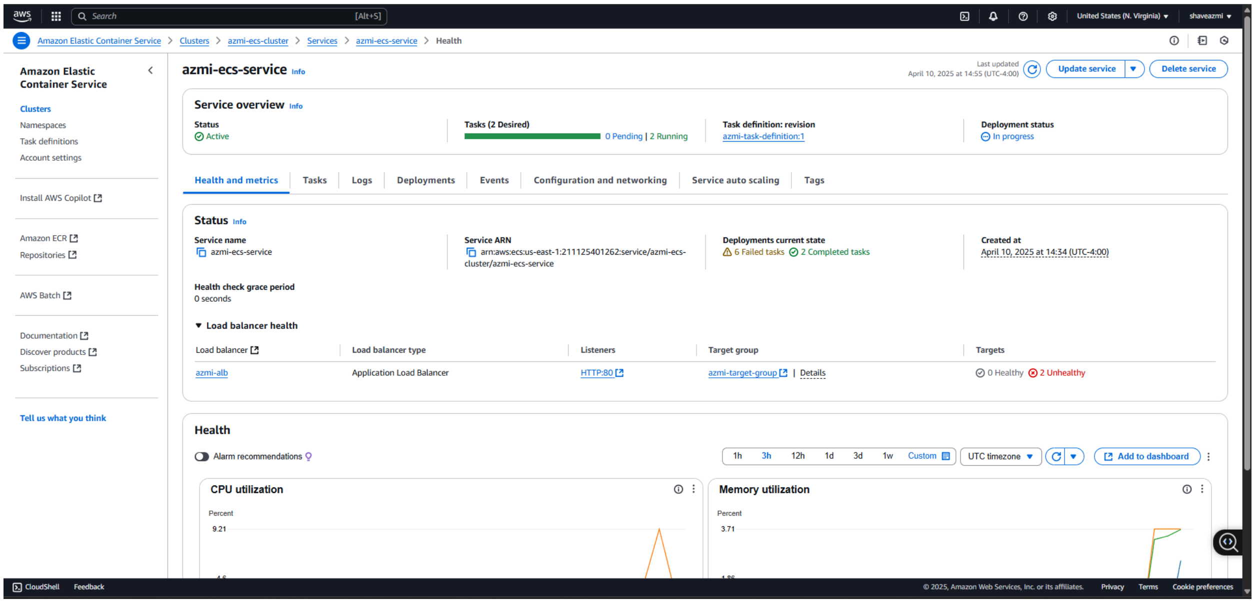
**Task memory**
512 MiB (0.5 GB)

**Task memory maximum allocation for container memory reservation**

### Containers Info

| Container name | Image | Private registry | Essential | CPU | Memory hard/soft limit | GPU |
|---|---|---|---|---|---|---|
| azmi-container | 211125401262.dkr.ecr.us-... | - | Yes | 0 | -/- | - |

# ECSService



## azmi-ecs-service Info

Last updated
April 10, 2025 at 14:55 (UTC-4:00)

### Service overview Info

**Status**
Active

**Tasks (2 Desired)**
0 Pending | 2 Running

**Task definition: revision**
azmi-task-definition:1

**Deployment status**
In progress

### Status Info

**Service name**
azmi-ecs-service

**Service ARN**
arn:aws:ecs:us-east-1:211125401262:service/azmi-ecs-cluster/azmi-ecs-service

**Deployments current state**
6 Failed tasks 2 Completed tasks

**Created at**
April 10, 2025 at 14:34 (UTC-4:00)

**Health check grace period**
0 seconds

**▼ Load balancer health**

| Load balancer | Load balancer type | Listeners | Target group | Targets |
|---|---|---|---|---|
| azmi-alb | Application Load Balancer | HTTP:80 | azmi-target-group \| Details | 0 Healthy 2 Unhealthy |

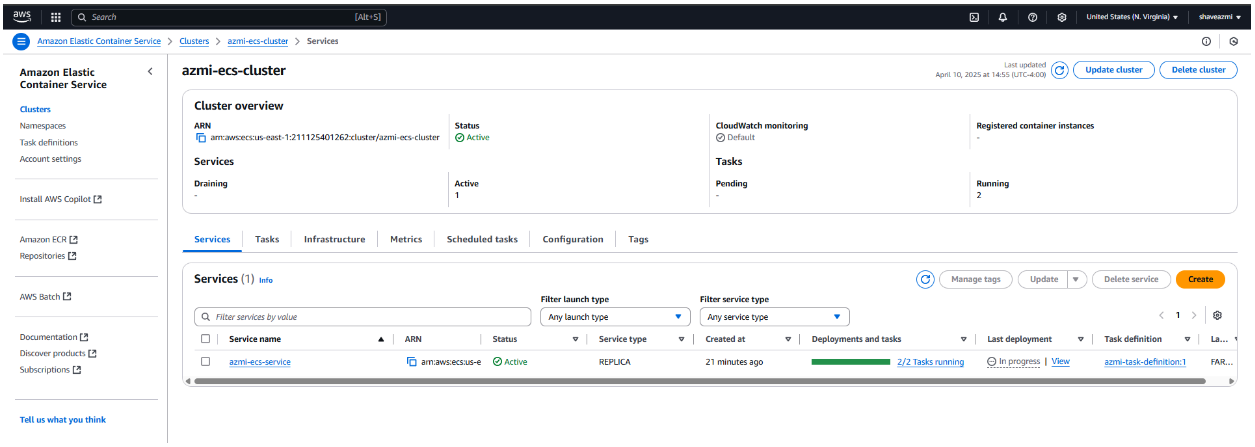
### Health

Alarm recommendations

**CPU utilization**

**Memory utilization**

# ECS Cluster



## azmi-ecs-cluster

Last updated
April 10, 2025 at 14:55 (UTC-4:00)

### Cluster overview

**ARN**
arn:aws:ecs:us-east-1:211125401262:cluster/azmi-ecs-cluster

**Status**
Active

**CloudWatch monitoring**
Default

**Registered container instances**
-

**Services**

**Draining**
-

**Active**
1

**Tasks**

**Pending**
-

**Running**
2

### Services (1) Info

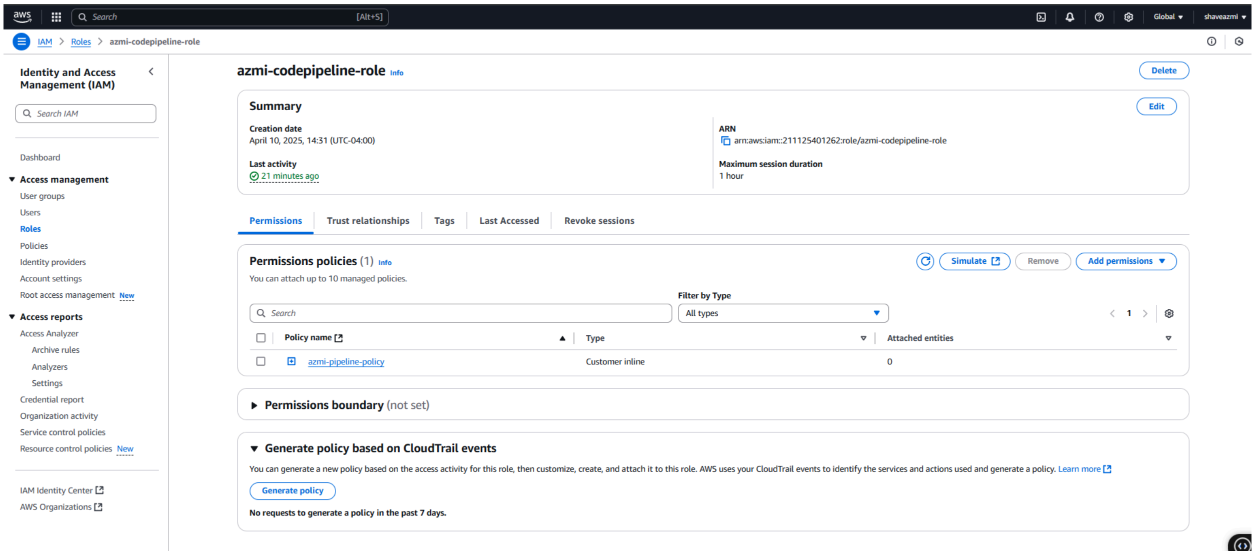
Filter services by value

Filter launch type: Any launch type

Filter service type: Any service type

| ☐ | Service name | ARN | Status | Service type | Created at | Deployments and tasks | Last deployment | Task definition | La... |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | azmi-ecs-service | arn:aws:ecs:us-e | Active | REPLICA | 21 minutes ago | 2/2 Tasks running | In progress \| View | azmi-task-definition:1 | FAR... |

# CodePipelineRole



## azmi-codepipeline-role Info

### Summary

**Creation date**
April 10, 2025, 14:31 (UTC-04:00)

**Last activity**
21 minutes ago

**ARN**
arn:aws:iam::211125401262:role/azmi-codepipeline-role

**Maximum session duration**
1 hour

### Permissions policies (1) Info

You can attach up to 10 managed policies.

Search

Filter by Type: All types

| ☐ | Policy name | Type | Attached entities |
|---|---|---|---|
| ☐ | azmi-pipeline-policy | Customer inline | 0 |

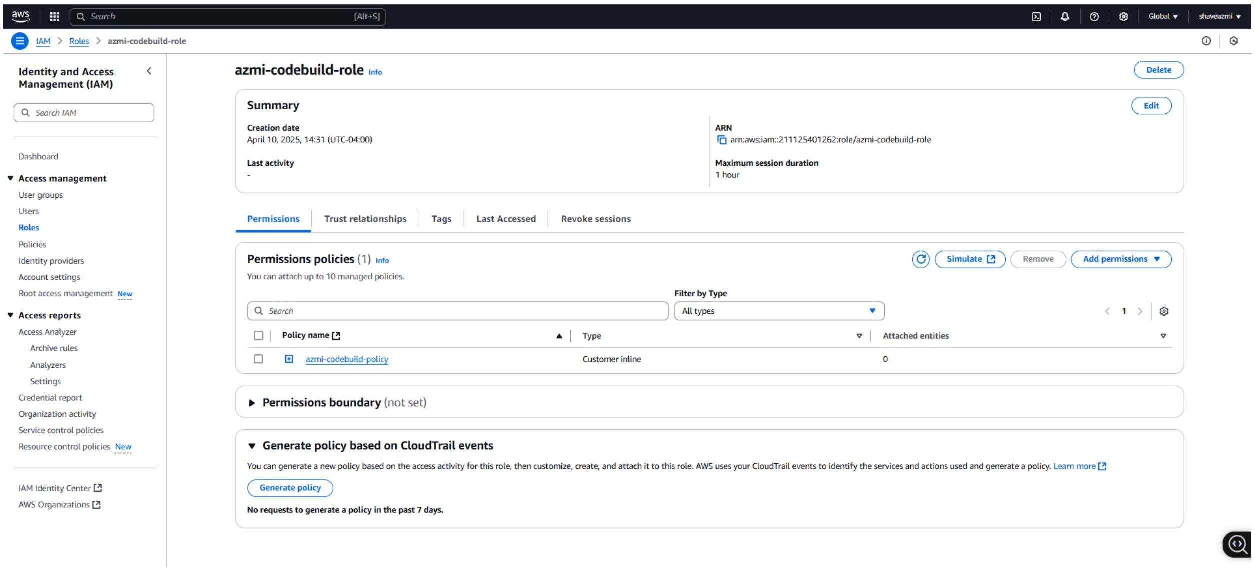
### Permissions boundary (not set)

### Generate policy based on CloudTrail events

You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

**No requests to generate a policy in the past 7 days.**

## CodeBuildRole



**azmi-codebuild-role**

**Summary**

**Creation date**
April 10, 2025, 14:31 (UTC-04:00)

**Last activity**
-

**ARN**
arn:aws:iam::211125401262:role/azmi-codebuild-role

**Maximum session duration**
1 hour

**Permissions policies** (1)

You can attach up to 10 managed policies.

Filter by Type: All types

| Policy name | Type | Attached entities |
|---|---|---|
| azmi-codebuild-policy | Customer inline | 0 |

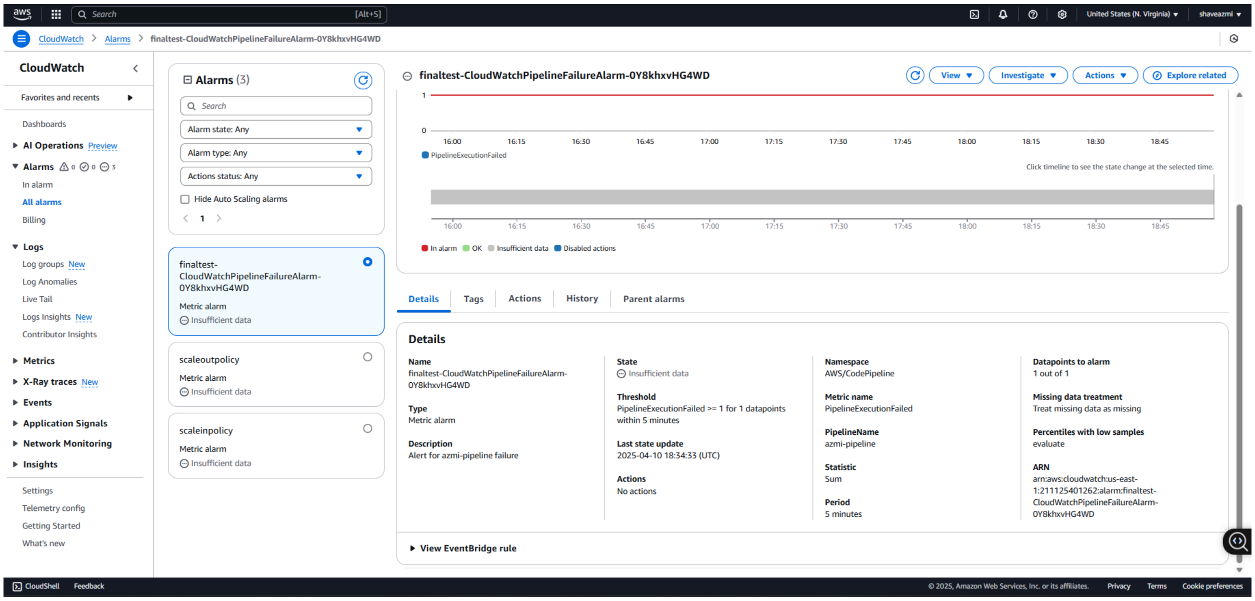
**Permissions boundary** (not set)

**Generate policy based on CloudTrail events**

You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

**No requests to generate a policy in the past 7 days.**

## CloudWatchPipelineFailureAlarm



finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD
Metric alarm
Insufficient data

scaleoutpolicy
Metric alarm
Insufficient data

scaleinpolicy
Metric alarm
Insufficient data

**finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD**

**Details**

**Name**
finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD

**Type**
Metric alarm

**Description**
Alert for azmi-pipeline failure

**State**
Insufficient data

**Threshold**
PipelineExecutionFailed >= 1 for 1 datapoints within 5 minutes

**Last state update**
2025-04-10 18:34:33 (UTC)

**Actions**
No actions

**Namespace**
AWS/CodePipeline

**Metric name**
PipelineExecutionFailed

**PipelineName**
azmi-pipeline

**Statistic**
Sum

**Period**
5 minutes

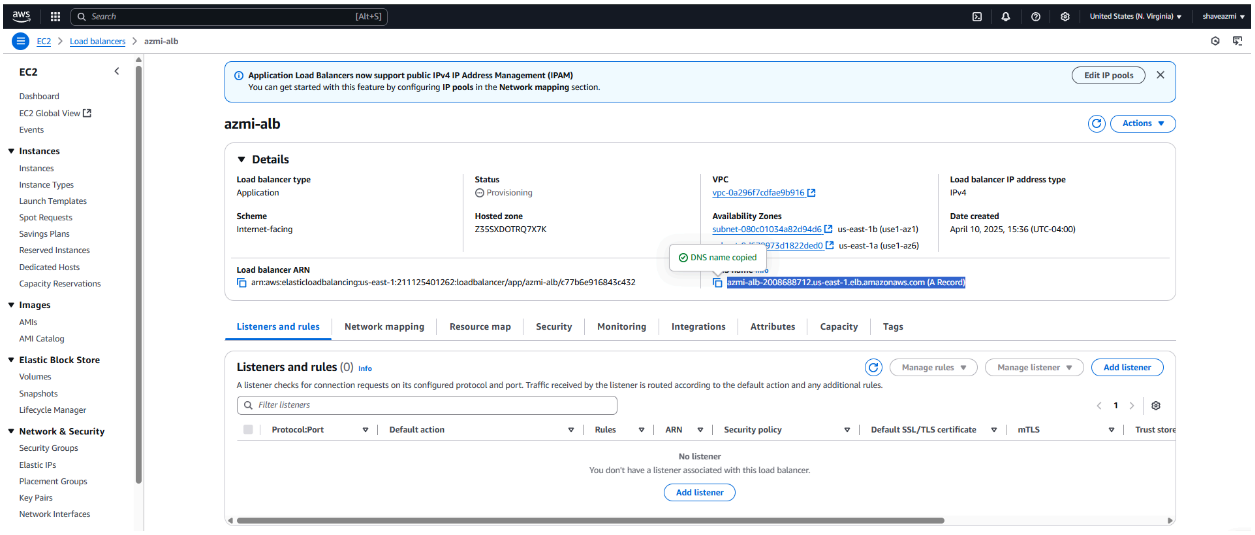
**Datapoints to alarm**
1 out of 1

**Missing data treatment**
Treat missing data as missing

**Percentiles with low samples**
evaluate

**ARN**
arn:aws:cloudwatch:us-east-1:211125401262:alarm:finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD

## ApplicationLoadBalancer



**Application Load Balancers now support public IPv4 IP Address Management (IPAM)**
You can get started with this feature by configuring **IP pools** in the **Network mapping** section.

**azmi-alb**

**Details**

**Load balancer type**
Application

**Status**
Provisioning

**VPC**
vpc-0a296f7cdfae9b916

**Load balancer IP address type**
IPv4

**Scheme**
Internet-facing

**Hosted zone**
Z35SXDOTRQ7X7K

**Availability Zones**
subnet-080c01034a82d94d6 us-east-1b (use1-az1)
...973d1822ded0 us-east-1a (use1-az6)

**Date created**
April 10, 2025, 15:36 (UTC-04:00)

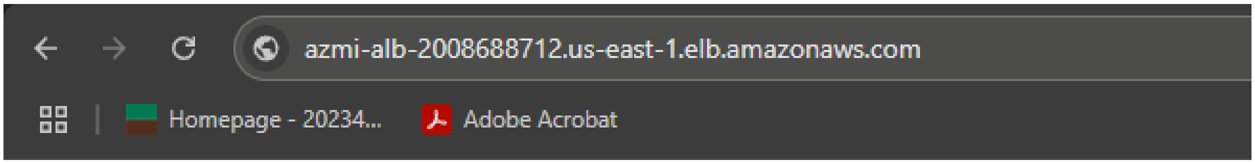
**Load balancer ARN**
arn:aws:elasticloadbalancing:us-east-1:211125401262:loadbalancer/app/azmi-alb/c77b6e916843c432

DNS name copied
azmi-alb-2008688712.us-east-1.elb.amazonaws.com (A Record)

**Listeners and rules** (0) Info
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.

No listener
You don't have a listener associated with this load balancer.

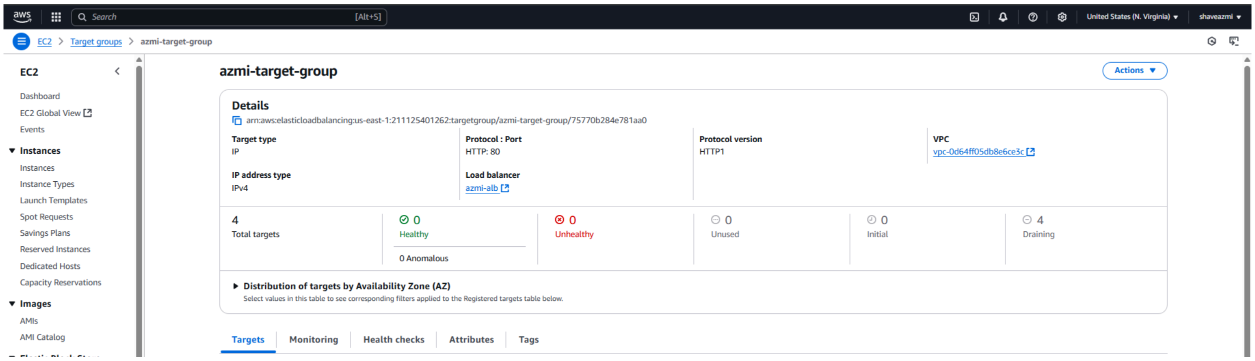
## Website loading



azmi-alb-2008688712.us-east-1.elb.amazonaws.com

# Azmi Hello Cloud!

## ALBTargetGroup



**azmi-target-group**

**Details**
arn:aws:elasticloadbalancing:us-east-1:211125401262:targetgroup/azmi-target-group/75770b284e781aa0

**Target type**
IP

**Protocol : Port**
HTTP: 80

**Protocol version**
HTTP1

**VPC**
vpc-0d64ff05db8e6ce3c

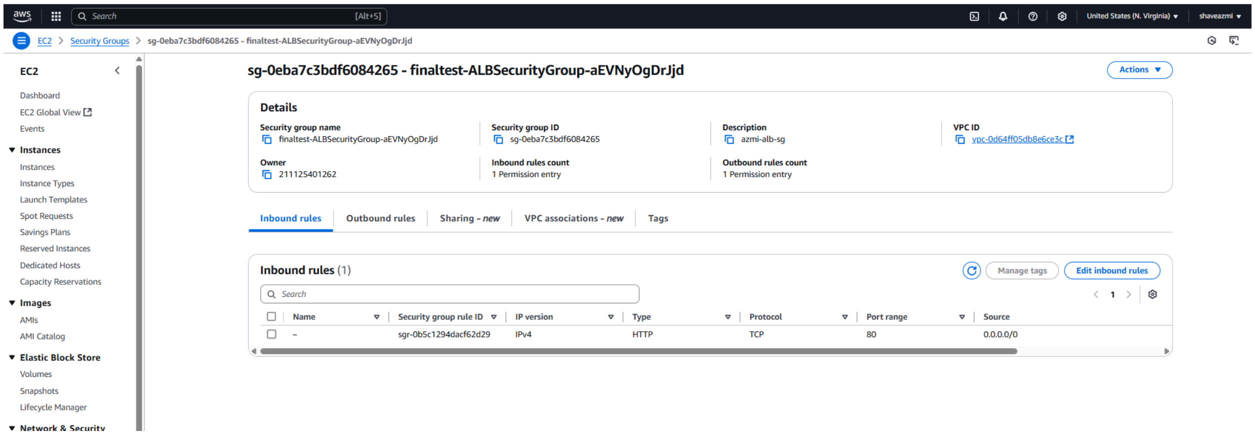
**IP address type**
IPv4

**Load balancer**
azmi-alb

| 4 | 0 | 0 | 0 | 0 | 4 |
|---|---|---|---|---|---|
| Total targets | Healthy<br>0 Anomalous | Unhealthy | Unused | Initial | Draining |

**Distribution of targets by Availability Zone (AZ)**
Select values in this table to see corresponding filters applied to the Registered targets table below.

# ALBSecurityGroup



sg-0eba7c3bdf6084265 - finaltest-ALBSecurityGroup-aEVNyOgDrJjd

Details

Security group name: finaltest-ALBSecurityGroup-aEVNyOgDrJjd

Security group ID: sg-0eba7c3bdf6084265

Description: azmi-alb-sg

VPC ID: vpc-0d64ff05db8e6ce3c

Owner: 211125401262

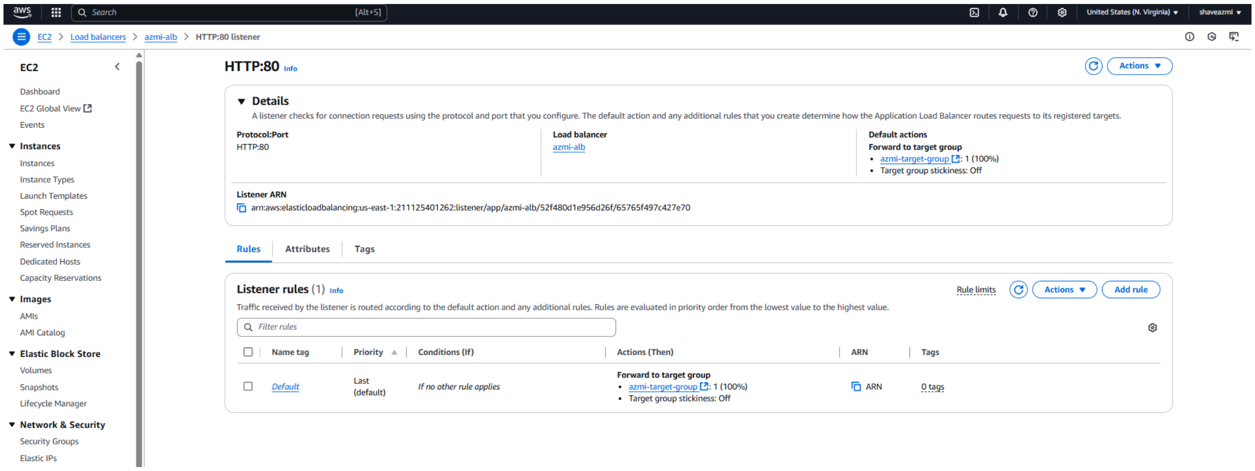
Inbound rules count: 1 Permission entry

Outbound rules count: 1 Permission entry

Inbound rules (1)

| ☐ | Name | Security group rule ID | IP version | Type | Protocol | Port range | Source |
|---|---|---|---|---|---|---|---|
| ☐ | – | sgr-0b5c1294dacf62d29 | IPv4 | HTTP | TCP | 80 | 0.0.0.0/0 |

# ALBListener



HTTP:80

Details

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.

Protocol:Port: HTTP:80

Load balancer: azmi-alb

Default actions
Forward to target group
- azmi-target-group: 1 (100%)
- Target group stickiness: Off

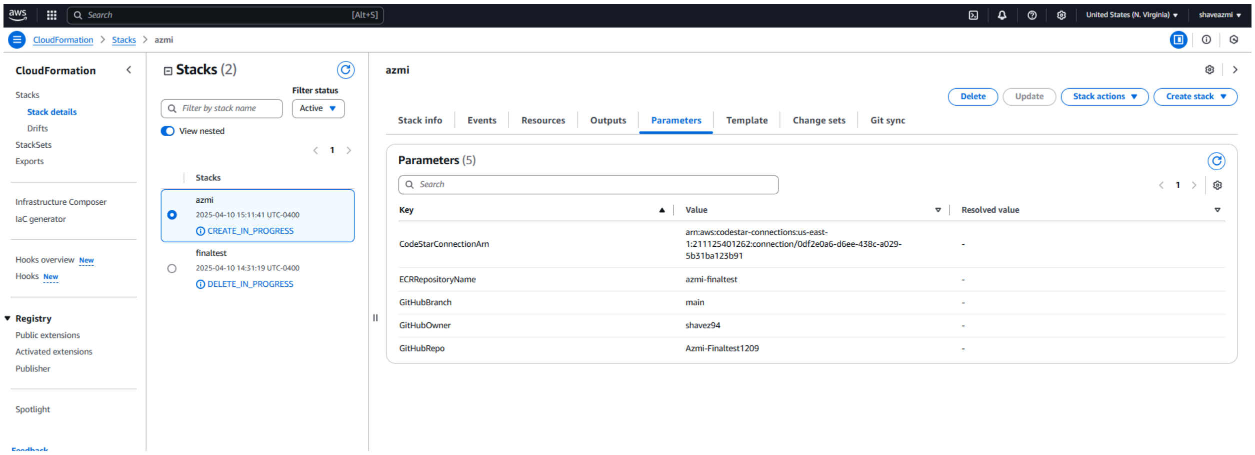
Listener ARN: arn:aws:elasticloadbalancing:us-east-1:211125401262:listener/app/azmi-alb/52f480d1e956d26f/65765f497c427e70

Listener rules (1)

Traffic received by the listener is routed according to the default action and any additional rules. Rules are evaluated in priority order from the lowest value to the highest value.

| ☐ | Name tag | Priority | Conditions (If) | Actions (Then) | ARN | Tags |
|---|---|---|---|---|---|---|
| ☐ | *Default* | Last<br>(default) | *If no other rule applies* | Forward to target group<br>• azmi-target-group: 1 (100%)<br>• Target group stickiness: Off | ARN | 0 tags |

# Stack Parameter



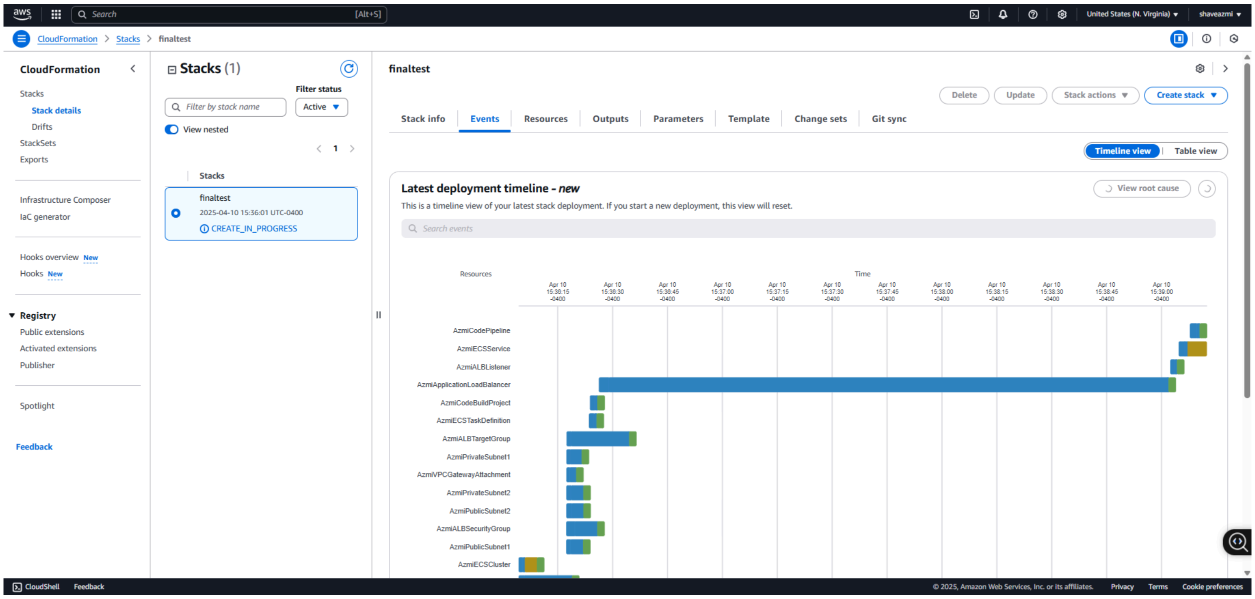
azmi
2025-04-10 15:11:41 UTC-0400
CREATE_IN_PROGRESS

finaltest
2025-04-10 14:31:19 UTC-0400
DELETE_IN_PROGRESS

Parameters (5)

| Key | Value | Resolved value |
|---|---|---|
| CodeStarConnectionArn | arn:aws:codestar-connections:us-east-1:211125401262:connection/0df2e0a6-d6ee-438c-a029-5b31ba123b91 | - |
| ECRRepositoryName | azmi-finaltest | - |
| GitHubBranch | main | - |
| GitHubOwner | shavez94 | - |
| GitHubRepo | Azmi-Finaltest1209 | - |

# Stack Running

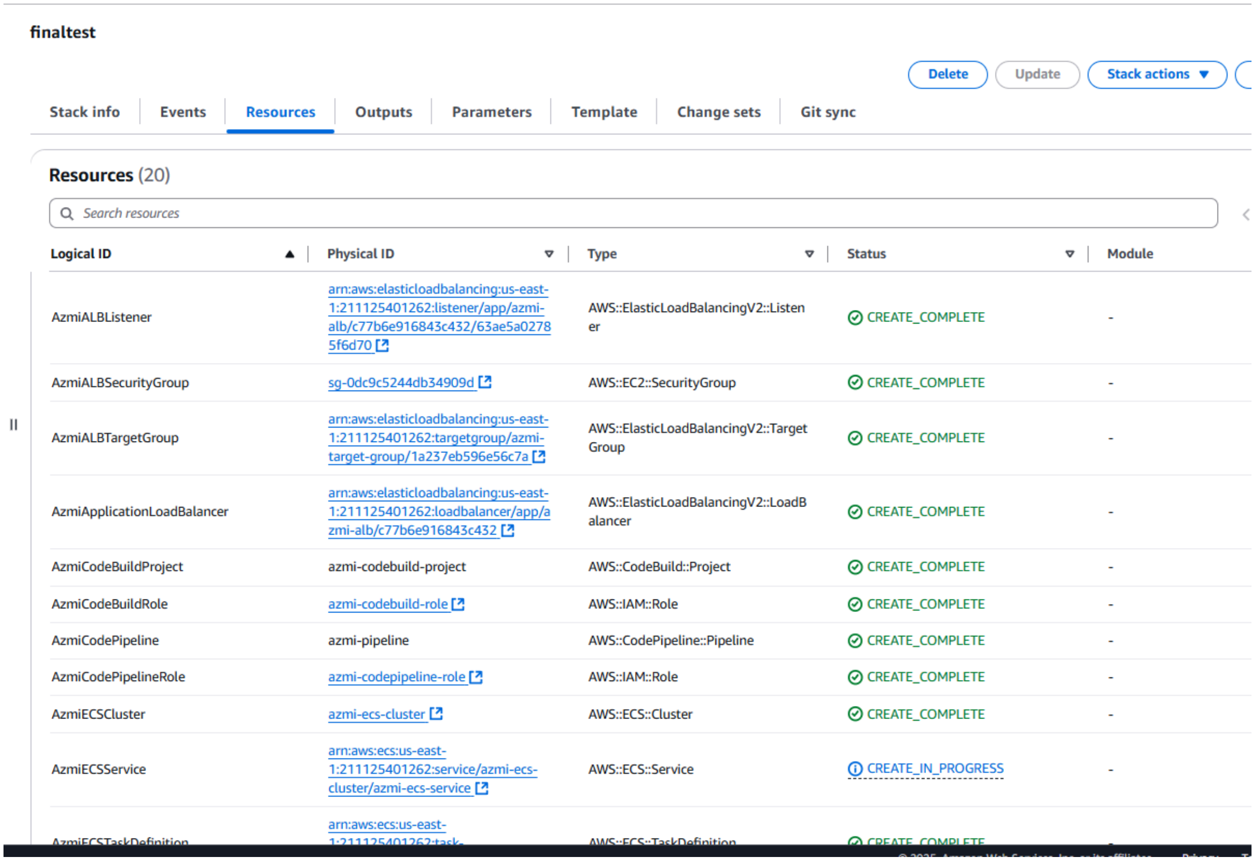


finaltest
2025-04-10 15:36:01 UTC-0400
CREATE_IN_PROGRESS

Latest deployment timeline - *new*

This is a timeline view of your latest stack deployment. If you start a new deployment, this view will reset.

© 2025, Amazon Web Services, Inc. or its affiliates.

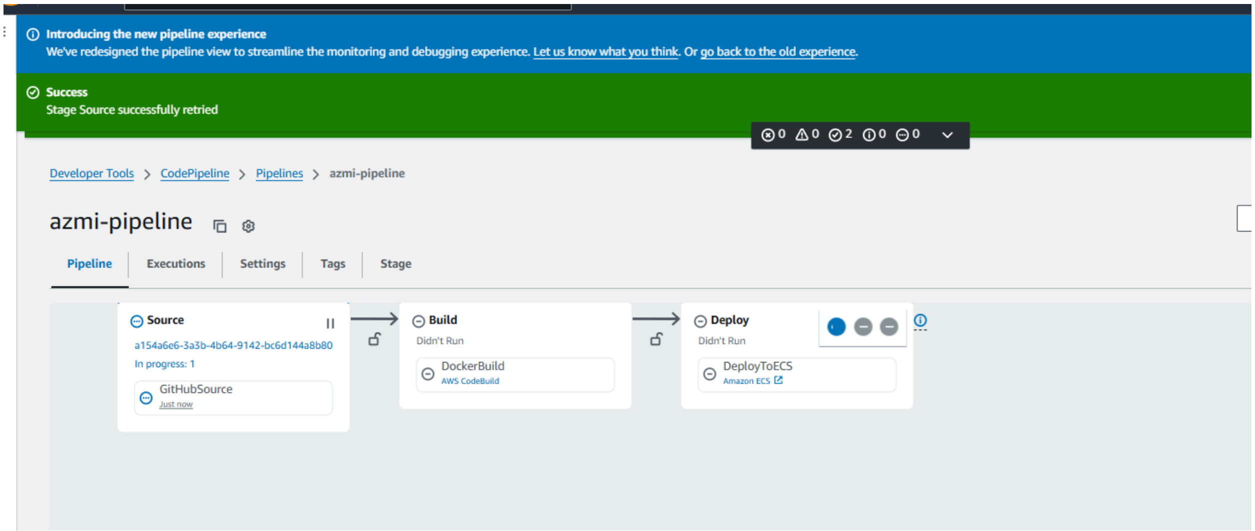


finaltest

Resources (20)

| Logical ID ▲ | Physical ID | Type | Status | Module |
|---|---|---|---|---|
| AzmiALBListener | arn:aws:elasticloadbalancing:us-east-1:211125401262:listener/app/azmi-alb/c77b6e916843c432/63ae5a02785f6d70 | AWS::ElasticLoadBalancingV2::Listener | CREATE_COMPLETE | - |
| AzmiALBSecurityGroup | sg-0dc9c5244db34909d | AWS::EC2::SecurityGroup | CREATE_COMPLETE | - |
| AzmiALBTargetGroup | arn:aws:elasticloadbalancing:us-east-1:211125401262:targetgroup/azmi-target-group/1a237eb596e56c7a | AWS::ElasticLoadBalancingV2::TargetGroup | CREATE_COMPLETE | - |
| AzmiApplicationLoadBalancer | arn:aws:elasticloadbalancing:us-east-1:211125401262:loadbalancer/app/azmi-alb/c77b6e916843c432 | AWS::ElasticLoadBalancingV2::LoadBalancer | CREATE_COMPLETE | - |
| AzmiCodeBuildProject | azmi-codebuild-project | AWS::CodeBuild::Project | CREATE_COMPLETE | - |
| AzmiCodeBuildRole | azmi-codebuild-role | AWS::IAM::Role | CREATE_COMPLETE | - |
| AzmiCodePipeline | azmi-pipeline | AWS::CodePipeline::Pipeline | CREATE_COMPLETE | - |
| AzmiCodePipelineRole | azmi-codepipeline-role | AWS::IAM::Role | CREATE_COMPLETE | - |
| AzmiECSCluster | azmi-ecs-cluster | AWS::ECS::Cluster | CREATE_COMPLETE | - |
| AzmiECSService | arn:aws:ecs:us-east-1:211125401262:service/azmi-ecs-cluster/azmi-ecs-service | AWS::ECS::Service | CREATE_IN_PROGRESS | - |
| AzmiECSTaskDefinition | arn:aws:ecs:us-east-1:211125401262:task- | AWS::ECS::TaskDefinition | CREATE_COMPLETE | - |

## Pipeline executed



Introducing the new pipeline experience
We've redesigned the pipeline view to streamline the monitoring and debugging experience. Let us know what you think. Or go back to the old experience.

Success
Stage Source successfully retried

azmi-pipeline

Source
a154a6e6-3a3b-4b64-9142-bc6d144a8b80
In progress: 1
GitHubSource
Just now

Build
Didn't Run
DockerBuild
AWS CodeBuild

Deploy
Didn't Run
DeployToECS
Amazon ECS