



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.

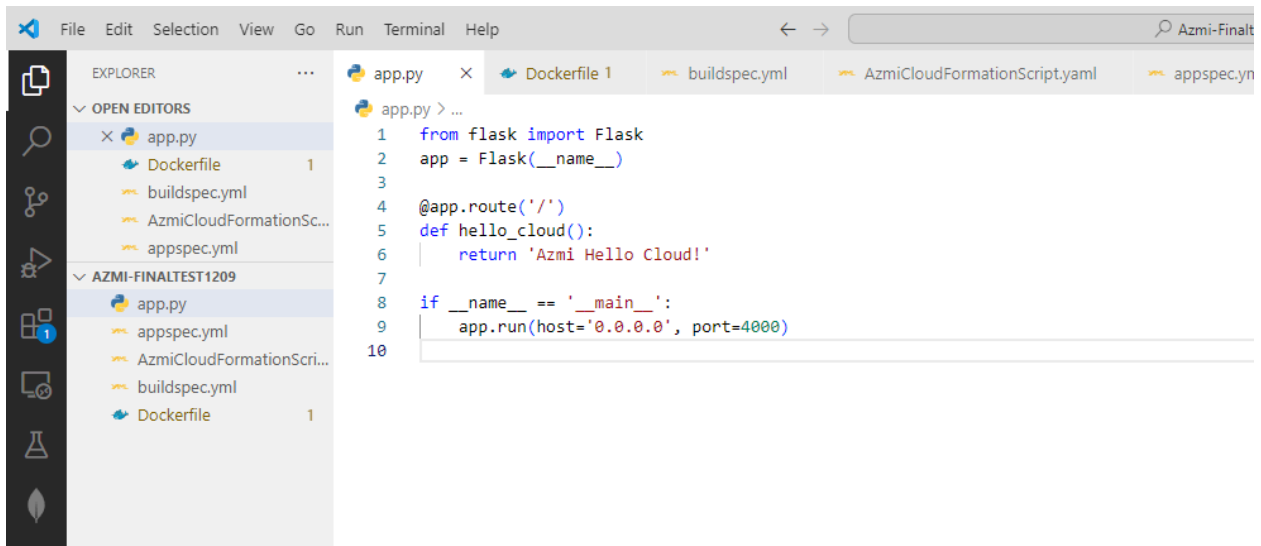
The application is updated via a CI/CD CodePipeline that pulls from a GitHub repository, builds, and pushes the image based on the docker file, and updates the ECS service with the new image. Wild Rydes has alarms set at each stage of the pipeline to determine build failures or an overall successful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

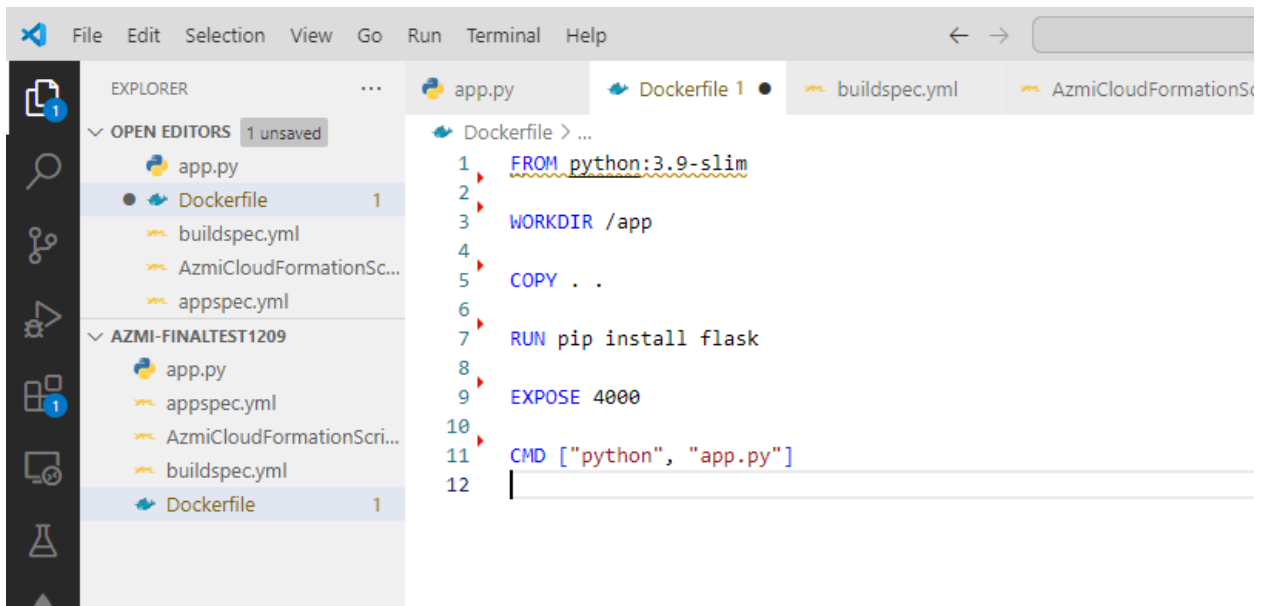
App.py



The screenshot shows the Visual Studio Code interface with the 'app.py' file open in the editor. The Explorer sidebar on the left shows the project structure, including the 'AZMI-FINALTEST1209' folder. The editor displays the following Python code:

```
1 from flask import Flask
2 app = Flask(__name__)
3
4 @app.route('/')
5 def hello_cloud():
6     return 'Azmi Hello Cloud!'
7
8 if __name__ == '__main__':
9     app.run(host='0.0.0.0', port=4000)
10
```

DockerFile



The screenshot shows the Visual Studio Code interface with the 'Dockerfile' file open in the editor. The Explorer sidebar on the left shows the project structure, including the 'AZMI-FINALTEST1209' folder. The editor displays the following Dockerfile content:

```
1 FROM python:3.9-slim
2
3 WORKDIR /app
4
5 COPY . .
6
7 RUN pip install flask
8
9 EXPOSE 4000
10
11 CMD ["python", "app.py"]
12
```

DockerImage Push

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
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
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
a8b6cfc48bce: Pushed
ceaf5dac753f: Pushed
8a628cdd7ccc: Pushed
8a628cdd7ccc: Pushed
97d21b95fb00: Pushed
7401b97cfabf: Pushed
788da61eeea0: Pushed
5938f16f8fc7: Pushed
latest: digest: sha256:0e77b94cf95351bb7ae6e88183277e6eee92631a838ab7c88929523afc02c1e6 size: 856
PS C:\Users\shave\OneDrive\Documents\GitHub\Azmi-Finaltest1209> []
```

BuildSpec.yml

```
File Edit Selection View Go Run Terminal Help
Azmi-Finaltest1209
EXPLORER
app.py Dockerfile 1
buildspec.yml
AzmiCloudFormationScript.yml
appspec.yml
AZMI-FINALTEST1209
app.py
appspec.yml
AzmiCloudFormationScript.yml
buildspec.yml
Dockerfile 1
buildspec.yml
3 phases:
4 install:
5 runtime-versions:
6 docker: 1.0
7
8 commands:
9 - pip install flask
10 build:
11 commands:
12 - echo Logging into Amazon ECR...
13 - aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 211125401262.dkr.ecr.us-east-1.amazonaws.com
14 - echo Building Docker image...
15 - docker build -t azmi-finaltest .
16 - docker tag azmi-finaltest 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
17 - docker push 211125401262.dkr.ecr.us-east-1.amazonaws.com/azmi-finaltest
18 artifacts:
19 files: "**/*"
20
```

Github Repo

shavez94 / Azmi-Finaltest1209

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Azmi-Finaltest1209 Public

Pin Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file Code

shavez94 1

AzmiCloudFormationScript.yml	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

README

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 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

Resource configurations

Resource gateways

vpc-0d64ff05db8e6ce3c / azmi-vpc

Details

VPC ID

vpc-0d64ff05db8e6ce3c

State

Available

Block Public Access

Off

DNS hostnames

Enabled

DNS resolution

Enabled

Tenancy

default

DHCP option set

default

Main route table

rtb-0146dcbeb449bb8cc

Main network ACL

acl-0f5d76b46746c256c

Default VPC

No

IPv4 CIDR

10.0.0.0/16

IPv6 pool

-

IPv6 CIDR (Network border group)

-

Network Address Usage metrics

Disabled

Route 53 Resolver DNS Firewall rule groups

-

Owner ID

211125401262

Resource map

CIDRs

Flow logs

Tags

Integrations

Resource map

VPC

Subnets (4)

Route tables (3)

Network connections (2)

Public Subnet1

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

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

Details

Subnet ID

subnet-0414f8ca3f651b473

Subnet ARN

arn:aws:ec2:us-east-1:211125401262:subnet/subnet-0414f8ca3f651b473

State

Available

Block Public Access

Off

IPv4 CIDR

10.0.1.0/24

IPv6 CIDR

-

Network border group

us-east-1

Default subnet

No

Auto-assign public IPv4 address

Yes

Availability Zone

us-east-1a

Availability Zone ID

use1-az5

Customer-owned IPv4 pool

-

Route table

rtb-023963ed34f6c219 | azmi-public-rt

Network ACL

acl-0f5d76b46746c256c

IPV4 only

No

IPV6 only

No

Auto-assign IPv6 address

No

Auto-assign customer-owned IPv4 address

No

IPV4 CIDR reservations

-

IPV6 CIDR reservations

-

Resource name DNS A record

Disabled

Resource name DNS AAAA record

Disabled

Owner ID

211125401262

Flow logs

Route table

Network ACL

CIDR reservations

Sharing

Tags

Flow logs

Search

Name

Flow log ID

Filter

Destination type

Destination name

IAM role ARN

No flow logs found in this Region

The screenshot displays the AWS Management Console interface for a specific subnet. The top navigation bar shows the AWS logo, search functionality, and user information. The left-hand sidebar contains various service categories like VPC dashboard, Virtual private cloud, Security, PrivateLink and Lattice, and IAM roles. The main content area is titled "subnet-042346a34a9e84d8b / azmi-public-subnet-2". It features a detailed "Details" section organized into four columns:

- Subnet ID:** subnet-042346a34a9e84d8b
- IPv4 CIDR:** 10.0.2.0/24
- Availability Zone:** us-east-1b
- Route table:** rtb-023d63ed3f4fce219 | azmi-public-r
- 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-0f07fb646746e256c
- 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 IPv6 pool:** -
- IPv6-only:** No
- DNS64:** Disabled
- Block Public Access:** Off
- IPv6 CIDR association ID:** -
- VPC:** vpc-0d64ff05db8e6ec3c | azmi-vpc
- Auto-assign public IPv4 address:** Yes
- Outpost ID:** -
- Hostname type:** IP name
- Owner:** 211125401262

Below the details, there are tabs for "Flow logs", "Route table", "Network ACL", "CIDR reservations", "Sharing", and "Tags". The "Flow logs" tab is active, showing a search bar and a message stating "No flow logs found in this Region".

The screenshot displays the AWS Management Console interface for a VPC dashboard. The main focus is on the details of a specific subnet, 'subnet-Oefd865dc462ab3a2'. The console shows various configuration options and their current states.

Category	Value / State
Subnet ID	subnet-Oefd865dc462ab3a2
IPv4 CIDR	10.0.3.0/24
Availability Zone	us-east-1a
Route table	rtb-0c5eeabe392cc50ea 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-Oefd865dc462ab3a2
Available IPv4 addresses	249
Availability Zone ID	use1-a26
Network ACL	acl-0f5076b46746ae256c
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 IPv6 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

Below the details section, there are tabs for 'Flow logs', 'Route table', 'Network ACL', 'CIDR reservations', 'Sharing', and 'Tags'. The 'Flow logs' tab is selected, showing a search bar and a table with columns: Name, Flow log ID, Filter, Destination type, Destination name, and IAM role ARN. A message at the bottom states: 'No flow logs found in this Region'.

PrivateSubnet2

The screenshot shows the AWS VPC console interface. The left sidebar contains the VPC dashboard with links to Virtual private cloud, Security, PrivateLink and Latency, and Route tables. The main content area displays the details for the subnet **subnet-04d9130202a79032b / azmi-private-subnet-2**. The details are organized into four columns: Subnet ID, Subnet ARN, State, and Block Public Access. Below the details, there are tabs for Flow logs, Route table, Network ACL, CIDR reservations, Sharing, and Tags. The Flow logs tab is active, showing a search bar and a table with columns: Name, Flow log ID, Filter, Destination type, Destination name, and IAM role ARN. The table is currently empty, indicating no flow logs found in this Region.

subnets > subnet-04d9130202a79032b

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

Details

Subnet ID subnet-04d9130202a79032b	Subnet ARN arn:aws:ec2:us-east-1:211125401262:subnet/subnet-04d9130202a79032b	State Available	Block Public Access Off
IPv4 CIDR 10.0.4.0/24	Available IPv4 addresses 249	IPv6 CIDR -	IPv6 CIDR association ID -
Availability Zone us-east-1b	Available IPv4 addresses uset-az1	Network border group us-east-1	VPC vpc-0d64f05db8e6ce3c azmi-vpc
Route table rtb-0c5eeabe392ce50ea azmi-private-rt	Availability Zone ID us-east-1a	Default subnet No	Auto-assign public IPv4 address No
Auto-assign IPv6 address No	Network ACL acl-0f5076b46746e256c	Customer-owned IPv4 pool -	Outpost ID -
IPv4 CIDR reservations -	Auto-assign customer-owned IPv4 address No	IPv6-only No	Hostname type IP name
Resource name DNS A record Disabled	IPv6 CIDR reservations -	DNS64 Disabled	Owner 211125401262
	Resource name DNS AAAA record Disabled		

Flow logs

Search

Name	Flow log ID	Filter	Destination type	Destination name	IAM role ARN
No flow logs found in this Region					

PrivateRouteTable

The screenshot shows the AWS VPC console interface. The left sidebar contains the VPC dashboard with links to Virtual private cloud, Security, PrivateLink and Latency, and Route tables. The main content area displays the details for the route table **rtb-0c5eeabe392ce50ea / azmi-private-rt**. The details are organized into four columns: Route table ID, Main, Explicit subnet associations, and Edge associations. Below the details, there are tabs for Routes, Subnet associations, Edge associations, Route propagation, and Tags. The Routes tab is active, showing a search bar and a table with columns: Destination, Target, Status, and Propagated. The table contains two entries: 0.0.0.0/0 with target nat-0d8cf69a650b343ea1 and 10.0.0.0/16 with target local.

route tables > rtb-0c5eeabe392ce50ea

rtb-0c5eeabe392ce50ea / azmi-private-rt

Details

Route table ID rtb-0c5eeabe392ce50ea	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-0d64f05db8e6ce3c azmi-vpc	Owner ID 211125401262		

Routes

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0d8cf69a650b343ea1	Active	No
10.0.0.0/16	local	Active	No

The screenshot shows the AWS VPC console interface. The left sidebar contains the VPC dashboard with links to Virtual private cloud, Security, PrivateLink and Latency, and Route tables. The main content area displays the details for the route table **rtb-0c5eeabe392ce50ea / azmi-private-rt**. The details are organized into four columns: Route table ID, Main, Explicit subnet associations, and Edge associations. Below the details, there are tabs for Routes, Subnet associations, Edge associations, Route propagation, and Tags. The Subnet associations tab is active, showing a search bar and a table with columns: Name, Subnet ID, IPv4 CIDR, and IPv6 CIDR. The table contains two entries: azmi-private-subnet-1 with subnet-0ef0865dc462ab3a2 and 10.0.3.0/24, and azmi-private-subnet-2 with subnet-04d9130202a79032b and 10.0.4.0/24.

route tables > rtb-0c5eeabe392ce50ea

rtb-0c5eeabe392ce50ea / azmi-private-rt

Details

Route table ID rtb-0c5eeabe392ce50ea	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-0d64f05db8e6ce3c azmi-vpc	Owner ID 211125401262		

Subnet associations

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
azmi-private-subnet-1	subnet-0ef0865dc462ab3a2	10.0.3.0/24	-
azmi-private-subnet-2	subnet-04d9130202a79032b	10.0.4.0/24	-

PublicRouteTable

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

rtb-023d63ed34f6ce219 / azmi-public-rt

Details

Route table ID

rtb-023d63ed34f6ce219

VPC

vpc-0d64f05db8efec3c | azmi-vpc

Main

No

Owner ID

211125401262

Explicit subnet associations

2 subnets

Edge associations

-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (2)

Filter routes

Both

Edit routes

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

VPC

vpc-0d64f05db8efec3c | azmi-vpc

Main

No

Owner ID

211125401262

Explicit subnet associations

2 subnets

Edge associations

-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Explicit subnet associations (2)

Find subnet association

Edit subnet associations

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
azmi-public-subnet-2	subnet-042346a54a9eb4d8b	10.0.2.0/24	-
azmi-public-subnet-1	subnet-0414f9ca3f651b473	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:

Find subnet association

Edit subnet associations

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

No subnets without explicit associations

All your subnets are associated with a route table.

ECSTaskExecutionRole

Amazon Elastic Container Service

Namespaces

Task definitions

Account settings

Install AWS Copilot

Amazon ECR

Repositories

AWS Batch

Documentation

Discover products

Subscriptions

Tell us what you think

azmi-ecs-cluster

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

Tasks

Infrastructure

Metrics

Scheduled tasks

Configuration

Tags

Tasks (8)

Filter desired status

Filter launch type

Manage tags

Stop

Run new task

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
d1db634840544a27af258...	Running	Running	azmi-tas...	Unknown	2 minutes ago	ecs-svc/11994203858...	1 minute ago	service:azmi-ecs-service	-	FARGATE
3049888985df4be0873b...	Deactivating	Stopped	azmi-tas...	Unknown	10 minutes ago	ecs-svc/11994203858...	9 minutes ago	service:azmi-ecs-service	-	FARGATE
a91e0e9b0bf343f6c0c04...	Stopped	Stopped	azmi-tas...	Unknown	20 minutes ago	ecs-svc/11994203858...	19 minutes ago	service:azmi-ecs-service	-	FARGATE
b6cea4ec60f46a8a52bfa...	Stopped	Stopped	azmi-tas...	Unknown	19 minutes ago	ecs-svc/11994203858...	19 minutes ago	service:azmi-ecs-service	-	FARGATE
b813d08fcd3b47138551...	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	-	FARGATE

ECSTaskDefinition

Amazon Elastic Container Service

Clusters

Namespaces

Task definitions

Account settings

Install AWS Copilot

Amazon ECR

Repositories

AWS Batch

Documentation

Discover products

Subscriptions

Tell us what you think

azmi-task-definition:1

Deploy Actions Create new revision

Overview

ARN

Task role

Fault injection

Status

Task execution role

Time created

Operating system/Architecture

App environment

Network mode

Containers

JSON

Task placement

Volumes (0)

Requires attributes

Tags

Task size

Task CPU

Task CPU maximum allocation for containers

Task memory

Task memory maximum allocation for container memory reservation

Containers

Container name

Image

Private registry

Essential

CPU

Memory hard/soft limit

GPU

ECSService

Amazon Elastic Container Service

Clusters

Namespaces

Task definitions

Account settings

Install AWS Copilot

Amazon ECR

Repositories

AWS Batch

Documentation

Discover products

Subscriptions

Tell us what you think

azmi-ecs-service

Last updated

Update service Delete service

Service overview

Status

Tasks (2 Desired)

Task definition: revision

Deployment status

Health and metrics

Tasks

Logs

Deployments

Events

Configuration and networking

Service auto scaling

Tags

Status

Service name

Service ARN

Deployments current state

Created at

Health check grace period

Load balancer health

Load balancer

Load balancer type

Listeners

Target group

Targets

Health

Alarm recommendations

CPU utilization

Memory utilization

ECS Cluster

Amazon Elastic Container Service

Clusters

Namespaces

Task definitions

Account settings

Install AWS Copilot

Amazon ECR

Repositories

AWS Batch

Documentation

Discover products

Subscriptions

Tell us what you think

azmi-ecs-cluster

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)

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-east-1:211125401262:cluster/azmi-ecs-cluster

Active

REPLICA

21 minutes ago

2/2 Tasks running

in progress

View

azmi-task-definition:1

FAR...

CodePipelineRole

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Root access management

Access reports

Access Analyzer

Archive rules

Analizers

Settings

Credential report

Organization activity

Service control policies

Resource control policies

IAM Identity Center

AWS Organizations

azmi-codepipeline-role

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

Trust relationships

Tags

Last Accessed

Revoke sessions

Permissions policies (1)

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

Generate policy

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

CodeBuildRole

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

Users

Roles

Policies

Identity providers

Account settings

Root access management

Access reports

Access Analyzer

Archive rules

Analysers

Settings

Credential report

Organization activity

Service control policies

Resource control policies

IAM Identity Center

AWS Organizations

azmi-codebuild-role

Info

Delete

Edit

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

Trust relationships

Tags

Last Accessed

Revoke sessions

Permissions policies (1)

Info

You can attach up to 10 managed policies.

Search

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](#)

Generate policy

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

CloudWatchPipelineFailureAlarm

CloudWatch

Alarms

finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD

Alarms (3)

Search

Alarm state: Any

Alarm type: Any

Actions status: Any

Hide Auto Scaling alarms

finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD

Metric alarm

Insufficient data

scaleoutpolicy

Metric alarm

Insufficient data

scaleinpolicy

Metric alarm

Insufficient data

finaltest-CloudWatchPipelineFailureAlarm-0Y8khxvHG4WD

View

Investigate

Actions

Explore related

1

0

16:00

16:15

16:30

16:45

17:00

17:15

17:30

17:45

18:00

18:15

18:30

18:45

PipelineExecutionFailed

Click timeline to see the state change at the selected time.

In alarm

OK

Insufficient data

Disabled actions

Details

Tags

Actions

History

Parent alarms

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

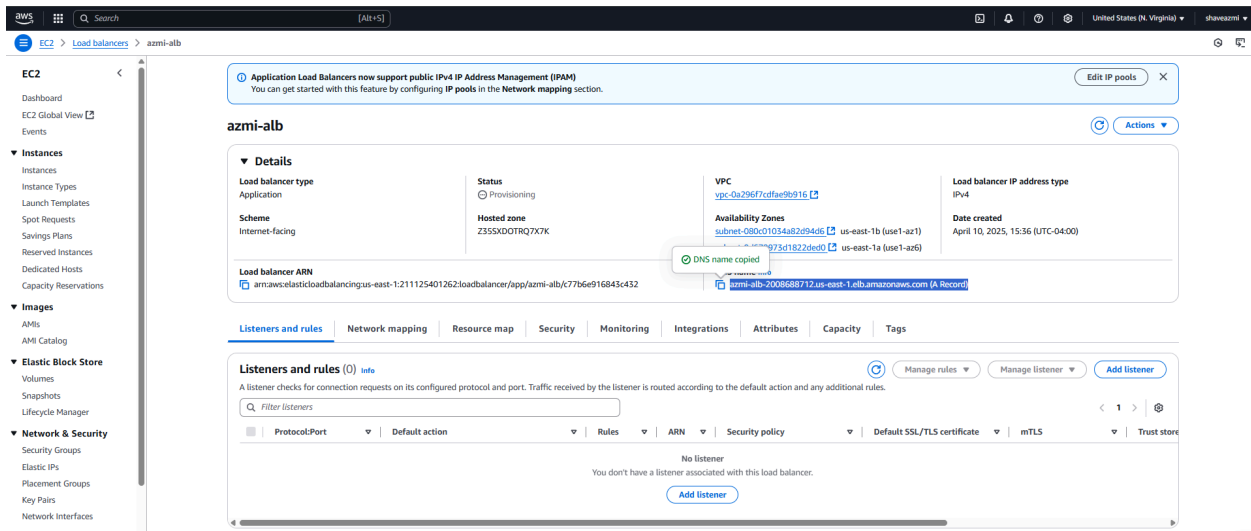
ARN

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

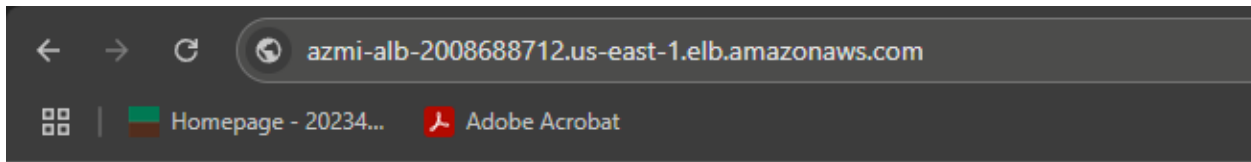
View EventBridge rule

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ApplicationLoadBalancer

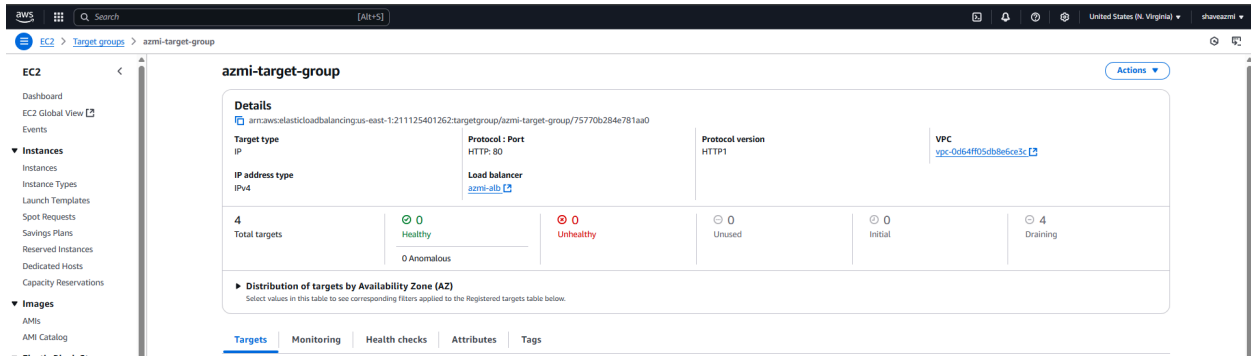


Website loading



Azmi Hello Cloud!

ALBTargetGroup



ALBSecurityGroup

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

sg-0eba7c3bdf6084265 - finaltest-ALBSecurityGroup-aEVNyOgDrJjd

Actions

Details

Security group name

finaltest-ALBSecurityGroup-aEVNyOgDrJjd

Security group ID

sg-0eba7c3bdf6084265

Description

azmi-alb-sg

VPC ID

vpc-0d64ff05dbb6e3c

Owner

211125401262

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

Search

<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sg-r085c1294dcf62d29	IPv4	HTTP	TCP	80	0.0.0.0/0

ALBListener

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

Load balancers

azmi-alb

HTTP:80 listener

Actions

Details

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

Rules

Attributes

Tags

Listener rules (1)

Rule limits

Actions

Add rule

Filter rules

<input type="checkbox"/>	Name tag	Priority	Conditions (If)	Actions (Then)	ARN	Tags
<input type="checkbox"/>	Default	Last (default)	If no other rule applies	Forward to target group	azmi-target-group (1 (100%))	Target group stickiness: Off

Stack Parameter

CloudFormation

Stacks

Stack details

StackSets

Exports

Infrastructure Composer

laC generator

Hooks overview

Hooks

Registry

Public extensions

Activated extensions

Publisher

Spotlight

Feedback

Stacks (2)

Filter status

Active

View nested

Stacks

azmi

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

CREATE_IN_PROGRESS

finaltest

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

DELETE_IN_PROGRESS

Stack info

Events

Resources

Outputs

Parameters

Template

Change sets

Git sync

Delete

Update

Stack actions

Create stack

Parameters (5)

Q Search

Key

Value

Resolved value

CodeStarConnectionArn

arn:aws:codestar-connections:us-east-1:211125401262:connection/0df2e0a6-d6ee-438c-a029-5b51ba723b91

-

ECRRepositoryName

azmi-finaltest

-

GitHubBranch

main

-

GitHubOwner

shavez94

-

GitHubRepo

Azmi-Finaltest1209

-

Stack Running

CloudFormation

Stacks

Stack details

StackSets

Exports

Infrastructure Composer

laC generator

Hooks overview

Hooks

Registry

Public extensions

Activated extensions

Publisher

Spotlight

Feedback

Stacks (1)

Filter status

Active

View nested

Stacks

finaltest

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

CREATE_IN_PROGRESS

Stack info

Events

Resources

Outputs

Parameters

Template

Change sets

Git sync

Delete

Update

Stack actions

Create stack

Timeline view

Table view

Latest deployment timeline - new

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

Q Search events

Resources

Time

AzmiCodePipeline

AzmiECSService

AzmiALBListener

AzmiApplicationLoadBalancer

AzmiCodeBuildProject

AzmiECSTaskDefinition

AzmiALBTargetGroup

AzmiPrivateSubnet1

AzmiVPCGatewayAttachment

AzmiPrivateSubnet2

AzmiPublicSubnet2

AzmiALBSecurityGroup

AzmiPublicSubnet1

AzmiECSCluster

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

finaltest

Delete Update Stack actions

Stack info Events Resources Outputs Parameters Template Change sets Git sync

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

Developer Tools > CodePipeline > Pipelines > azmi-pipeline

azmi-pipeline

Pipeline Executions Settings Tags Stage

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