|  |
| --- |
| PROG8870 - Spring 2025 - Section 1 |
| Assignment3: Deploying AWS EC2 Instance and RDS Instance using CloudFormation |
| Instructor: Prof. Vikas Vattikonda |

|  |
| --- |
| By Twinkle Mishra -8894858  Due Date: 22-7-2025 |

# *Github Repo:* [*https://github.com/TwinkleM97/A3-IAC-8894858*](https://github.com/TwinkleM97/A3-IAC-8894858)

# Objective

The main objective of this A3 was to learn how to use AWS CloudFormation to build and manage cloud infrastructure using Infrastructure as Code (IaC). The goal was to automate the setup of three major components on AWS:

* Networking:

I created a custom Virtual Private Cloud (VPC) with both public and private subnets. I also configured an Internet Gateway and added a Route Table so that the public subnet could access the internet.

* EC2 Instance:

I've launched a Linux based EC2 virtual machine inside the public subnet. I used a key pair for SSH access, allowed traffic on port 22, and passed in parameters for the AMI and instance type.

* RDS Database:

I deployed a MySQL-compatible RDS instance inside the private subnet. For testing, I made it publicly accessible and created a database called assignment3. I also allowed traffic on port 3306.

Each component was created using a separate YAML-based CloudFormation template. I deployed them one by one using the AWS Console and verified everything using both the Console and AWS CLI.

* **Step 1: Deploy Networking Stack**

aws cloudformation create-stack \

--stack-name assignment3-networking-twinklemishra \

--template-body file://templates/networking-template.yaml \

--capabilities CAPABILITY\_NAMED\_IAM \

--region us-east-1

* **Step 2: Deploy EC2 Stack**

aws cloudformation create-stack \

--stack-name assignment3-ec2-twinklemishra \

--template-body file://templates/ec2-template.yaml \

--parameters ParameterKey=KeyName,ParameterValue=assignment3-keypair \

ParameterKey=AMI,ParameterValue=ami-xxxxxxxxxxxxx \

ParameterKey=InstanceType,ParameterValue=t2.micro \

--capabilities CAPABILITY\_NAMED\_IAM \

--region us-east-1

* **Step 3: Deploy RDS Stack**

aws cloudformation create-stack \

--stack-name assignment3-rds-twinklemishra \

--template-body file://templates/rds-template.yaml \

--parameters ParameterKey=DBUsername,ParameterValue=admin \

ParameterKey=DBPassword,ParameterValue=StrongPassword123 \

--capabilities CAPABILITY\_NAMED\_IAM \

--region us-east-1

* **Step 4: CLI Verification**

*# Get EC2 Instance Outputs*

aws cloudformation describe-stacks \

--stack-name assignment3-ec2-twinklemishra \

--region us-east-1 \

--query "Stacks[0].Outputs"

*# Get RDS Endpoint*

aws cloudformation describe-stacks \

--stack-name assignment3-rds-twinklemishra \

--region us-east-1 \

--query "Stacks[0].Outputs"

*# EC2 instance check*

aws ec2 describe-instances --region us-east-1

*# RDS instance check*

aws rds describe-db-instances --region us-east-1

* **Step 5: Cleanup**

*# Delete stacks*

aws cloudformation delete-stack --stack-name assignment3-ec2-twinklemishra --region us-east-1

aws cloudformation delete-stack --stack-name assignment3-rds-twinklemishra --region us-east-1

aws cloudformation delete-stack --stack-name assignment3-networking-twinklemishra --region us-east-1

*# Confirm deletion*

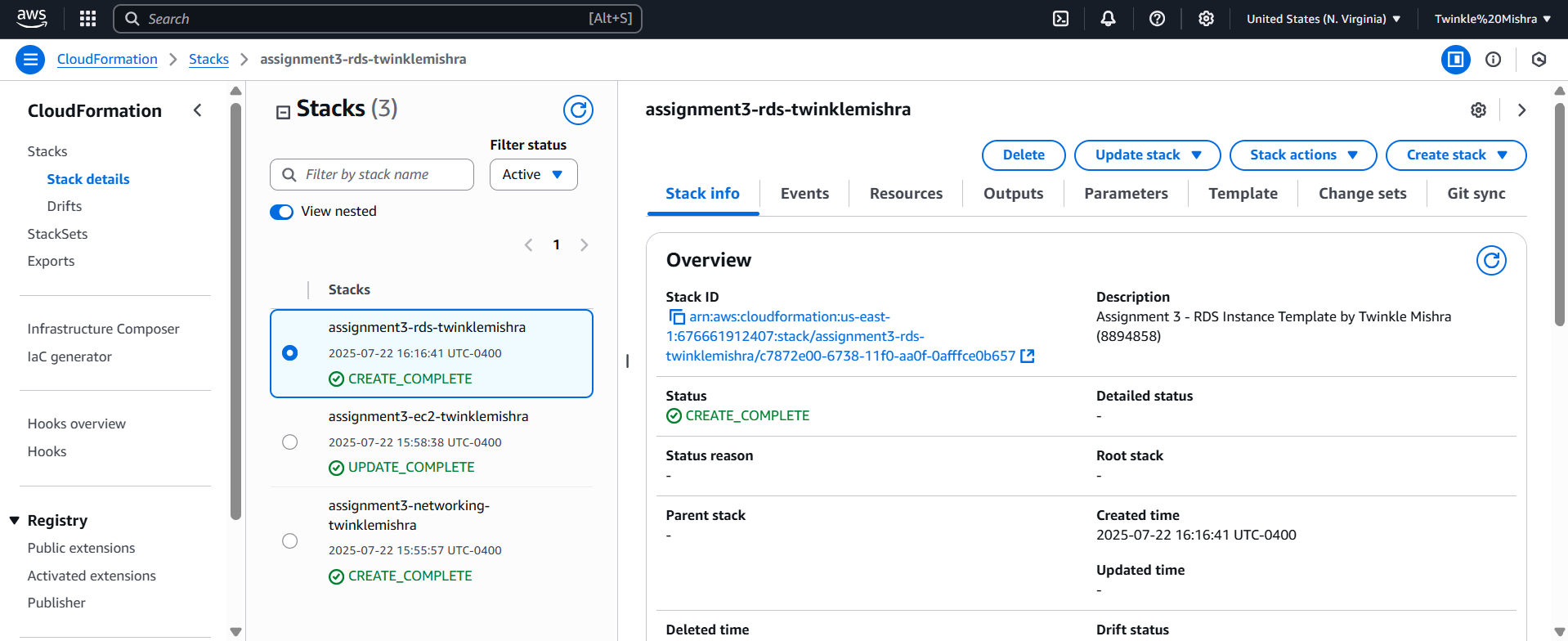
aws ec2 describe-instances --region us-east-1

aws rds describe-db-instances --region us-east-1

aws ec2 describe-vpcs --region us-east-1

# Screenshots

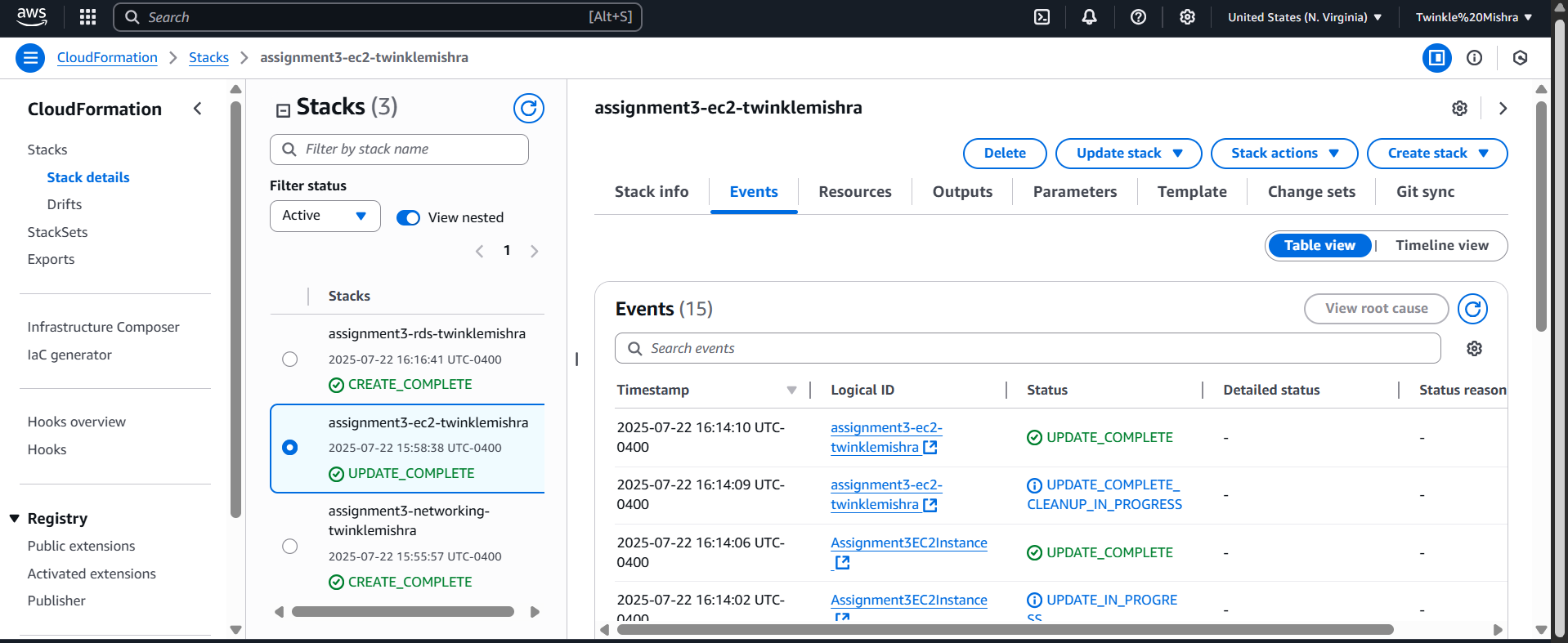
## CloudFormation-Stack Status Overview



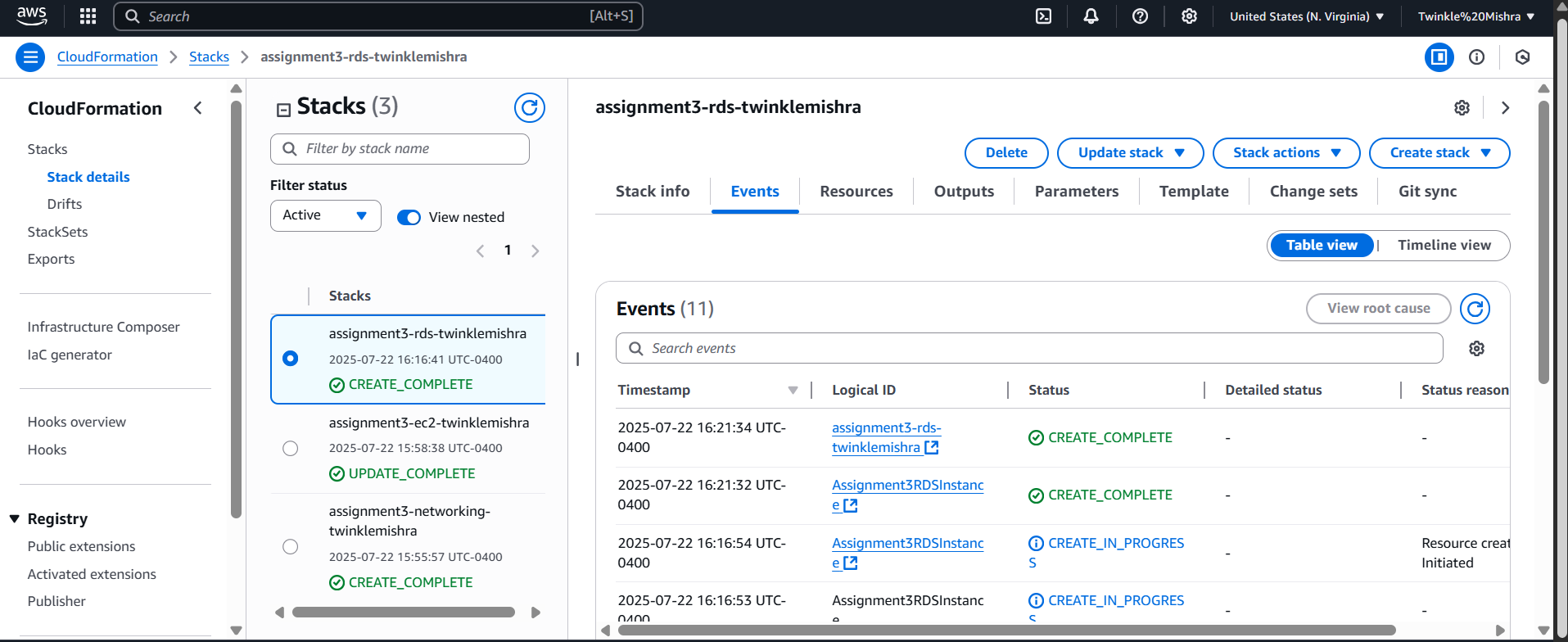
## EC2 Instance Public IP from Stack Output



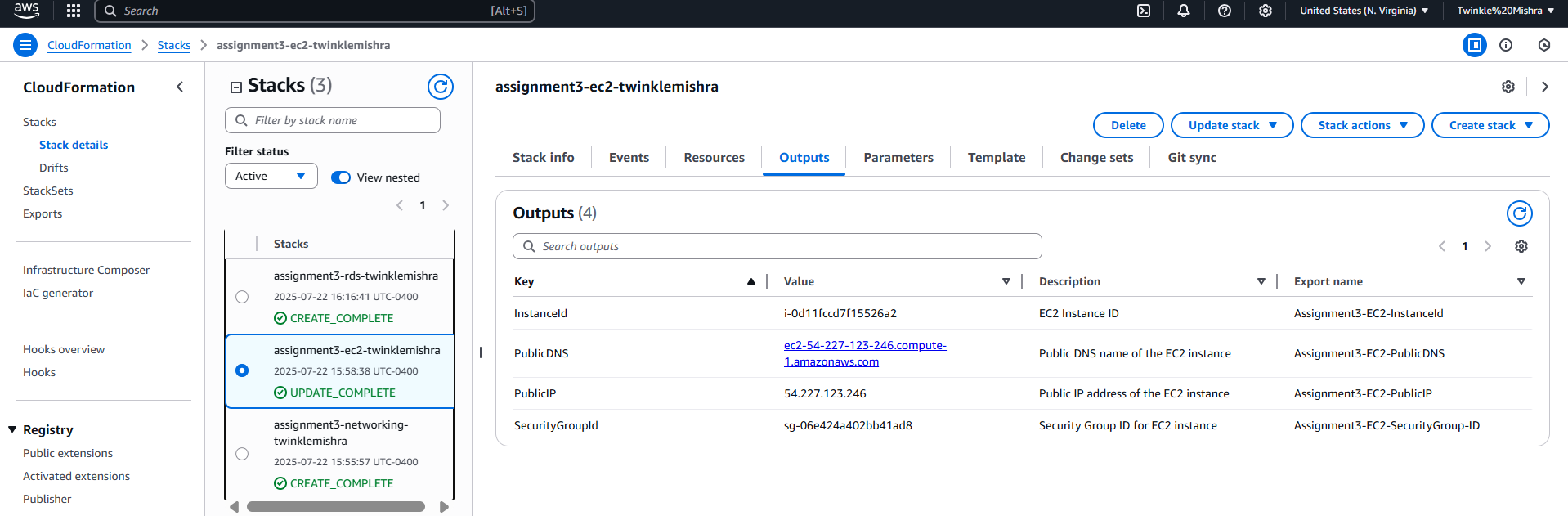
## EC2 Stack Creation Events



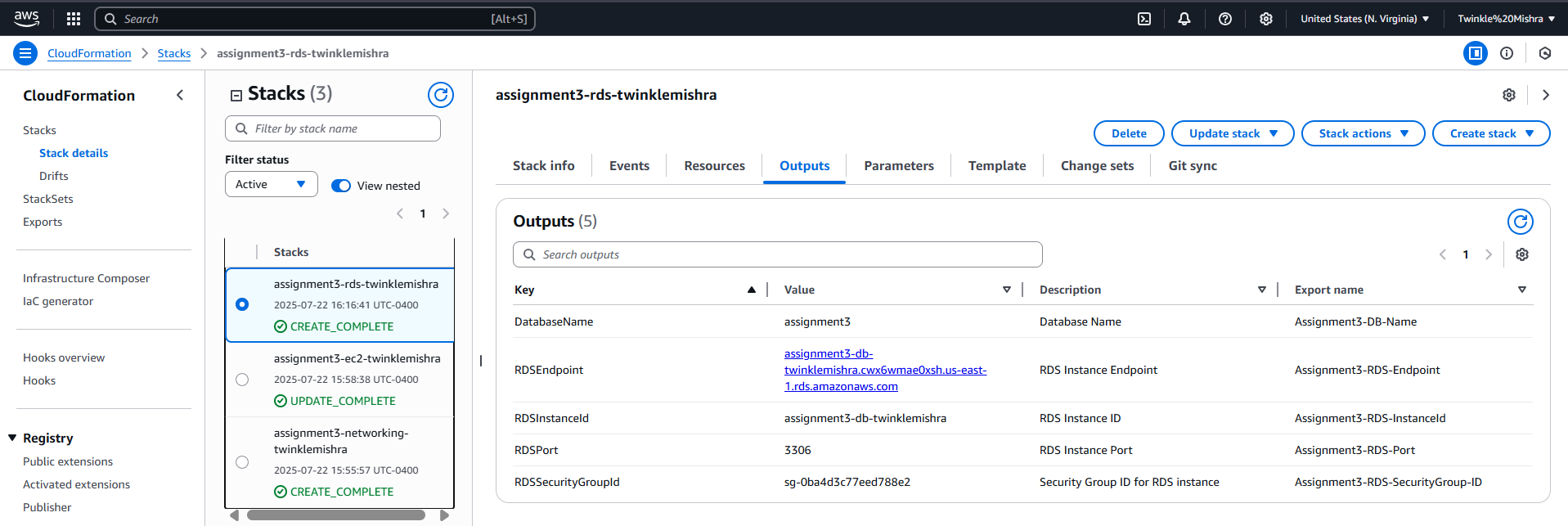
## RDS Stack Creation Events



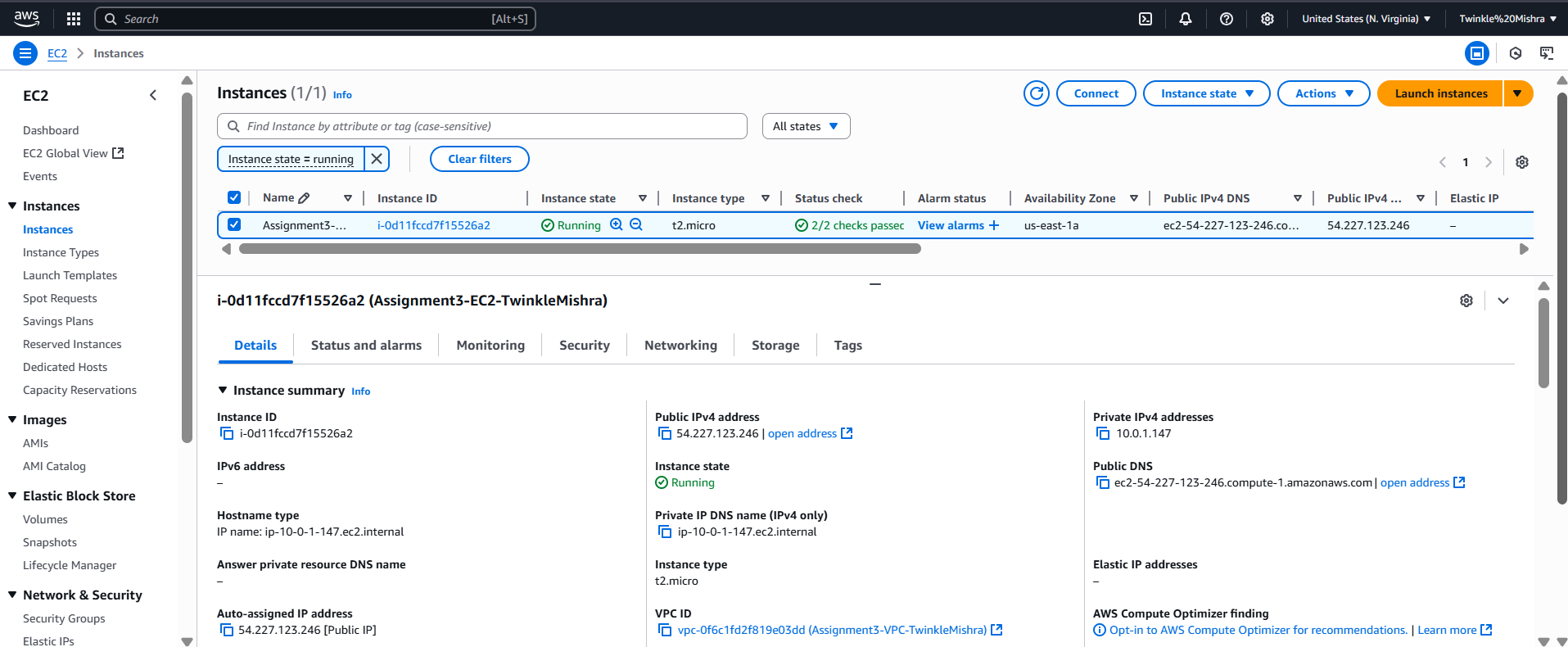
## EC2 Stack Outputs



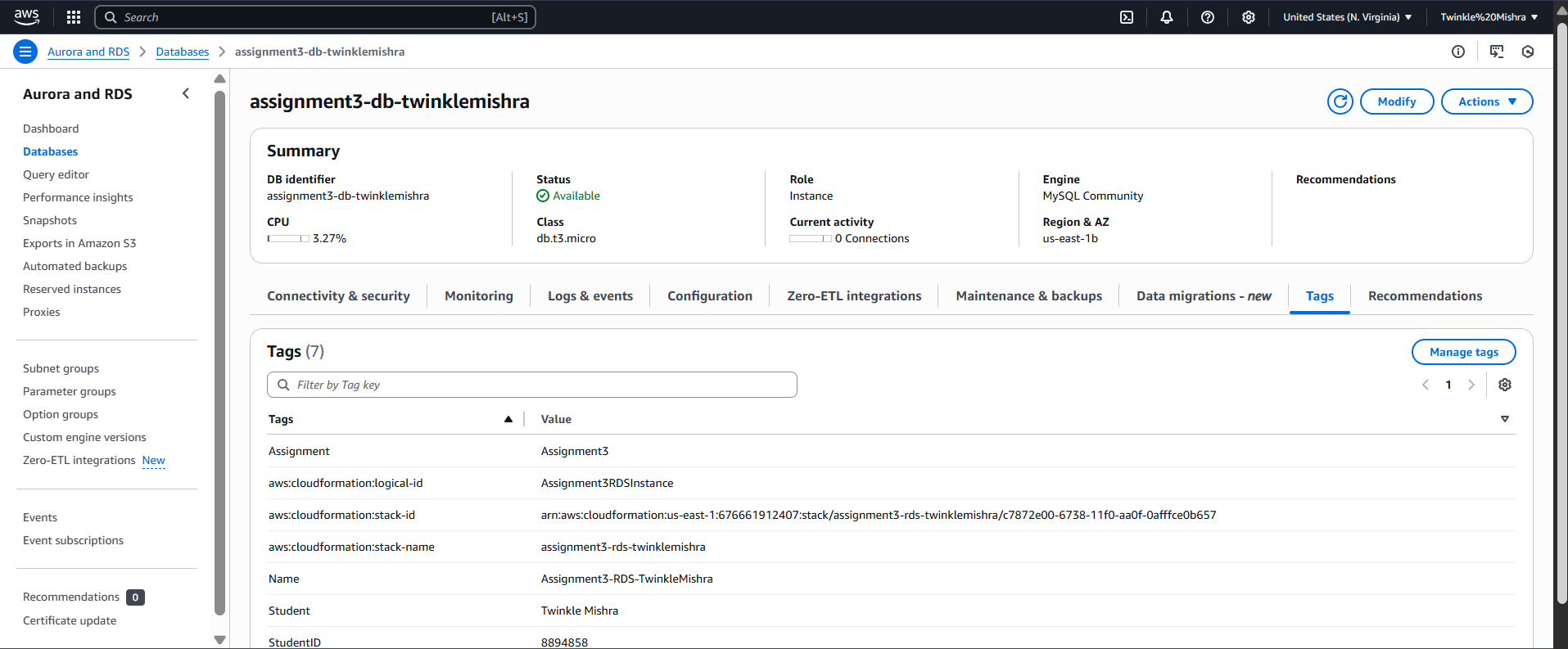
## RDS Stack Outputs



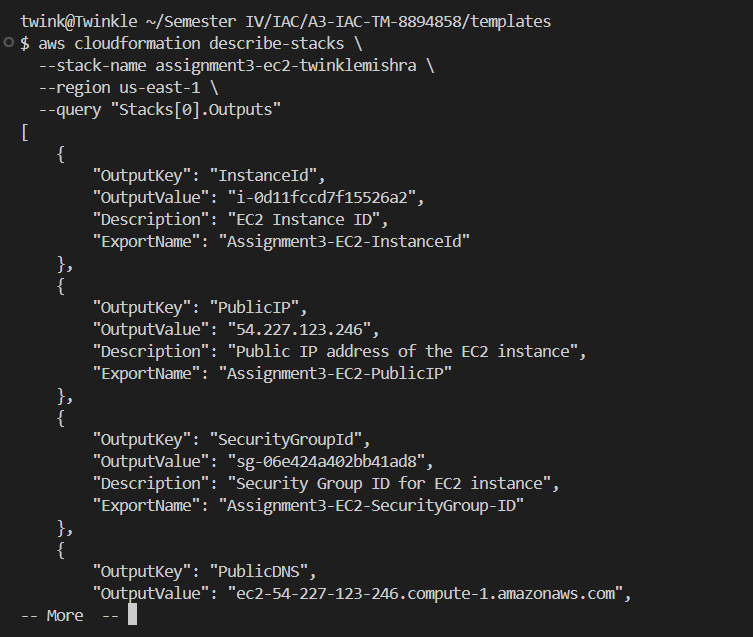
## Running EC2 Instance in AWS Console



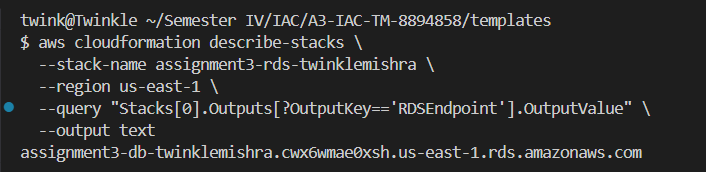
## Running RDS Instance in AWS Console



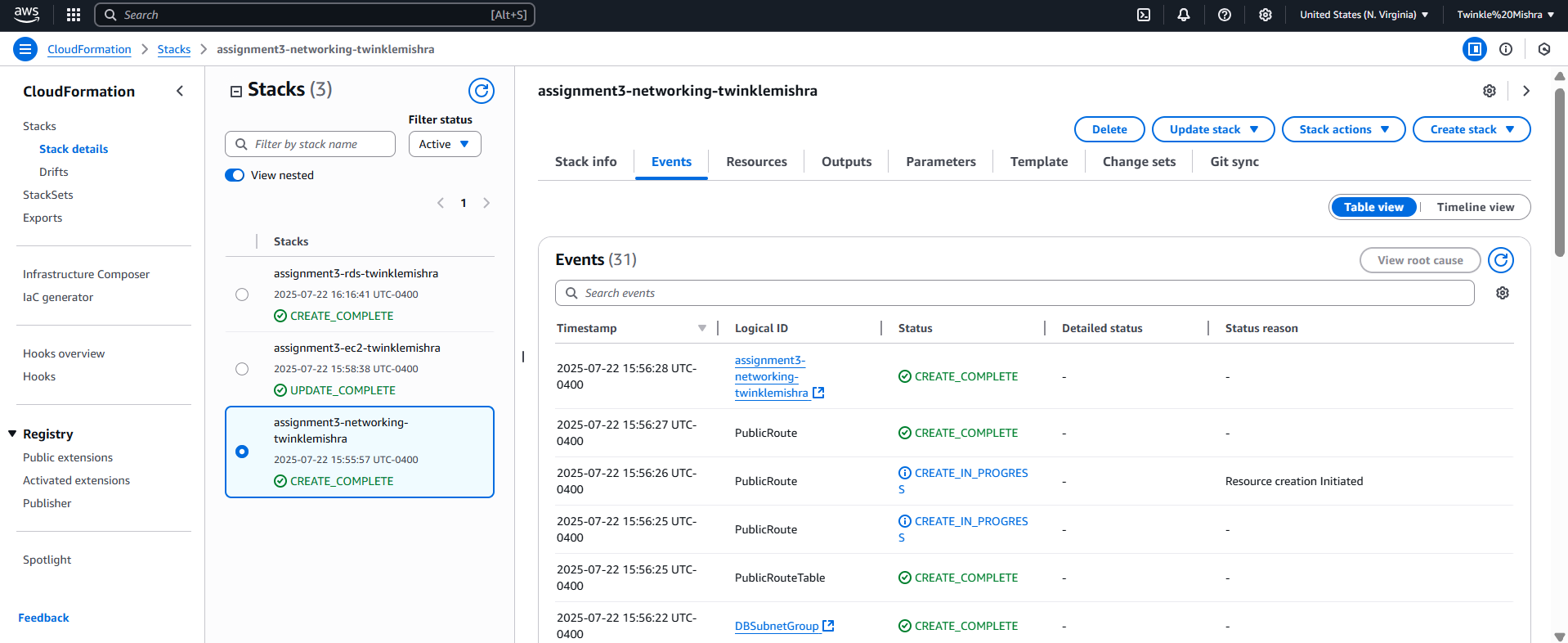
## CLI Output: Describe EC2 Instance



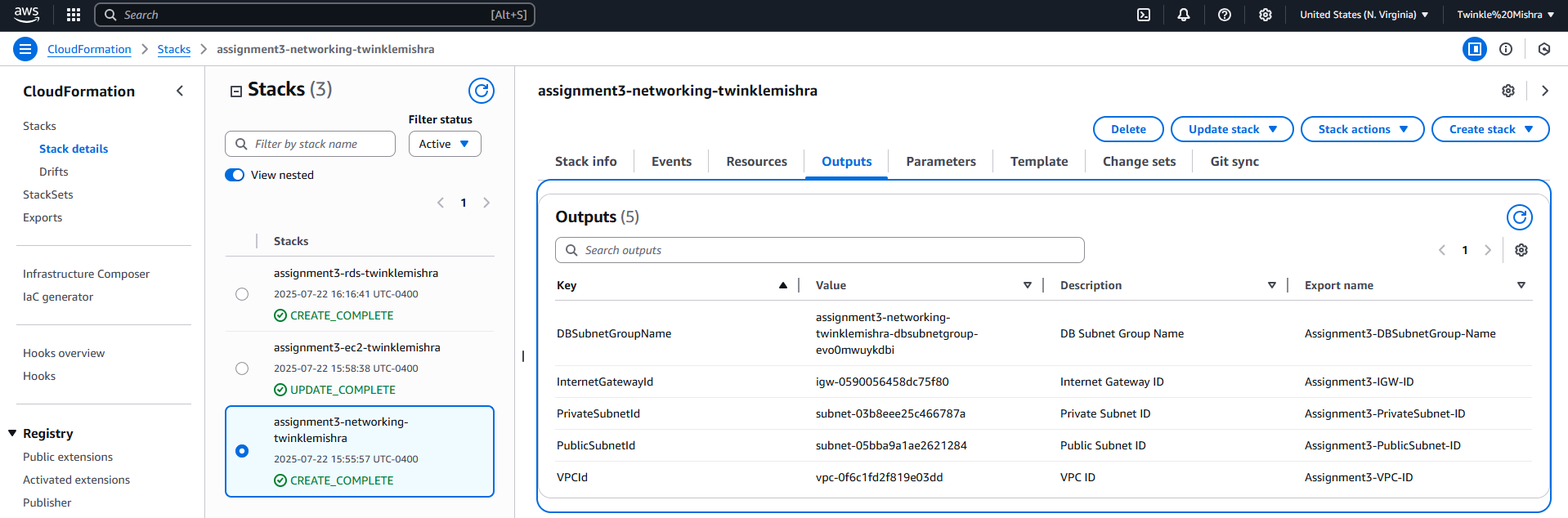
## CLI Output: Describe RDS Instance



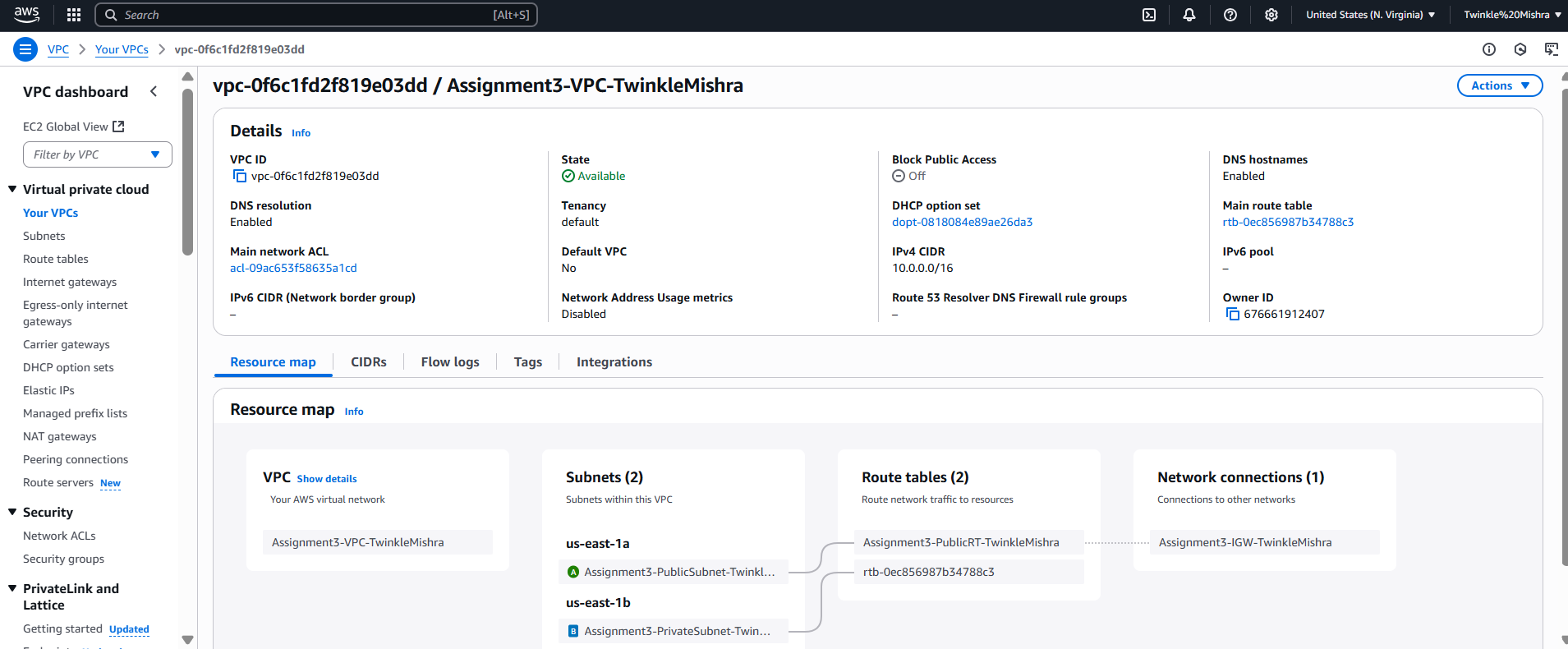
## Networking Stack Events



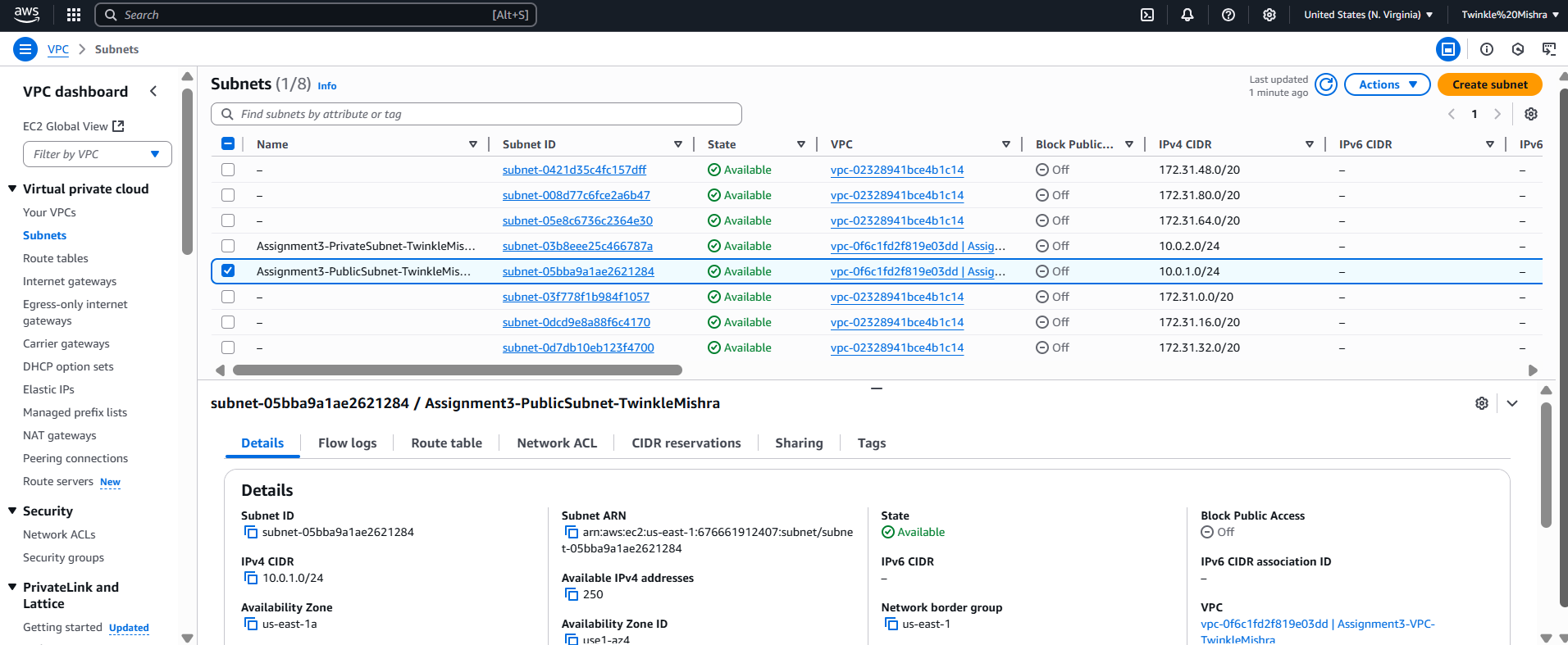
## Networking Stack Outputs



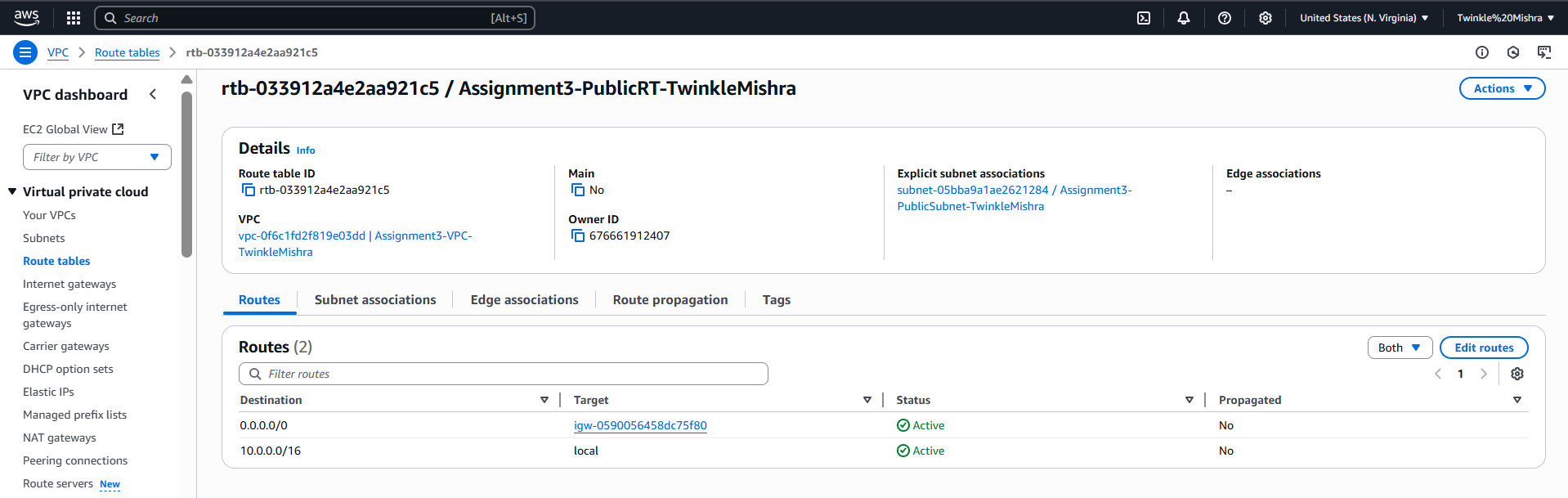
## VPC Created via CloudFormation



## Public and Private Subnets



## Public Route Table Associated



## Networking-Stack Status in CloudFormation

