PROG 8870 - Cloud Architectures and Infrastructure as Code

Deploying AWS Infrastructure with Terraform and CloudFormation

Project Report

Prepared by : Nikhil Shankar Chirakkal Sivasankaran **Student ID:** 9026254

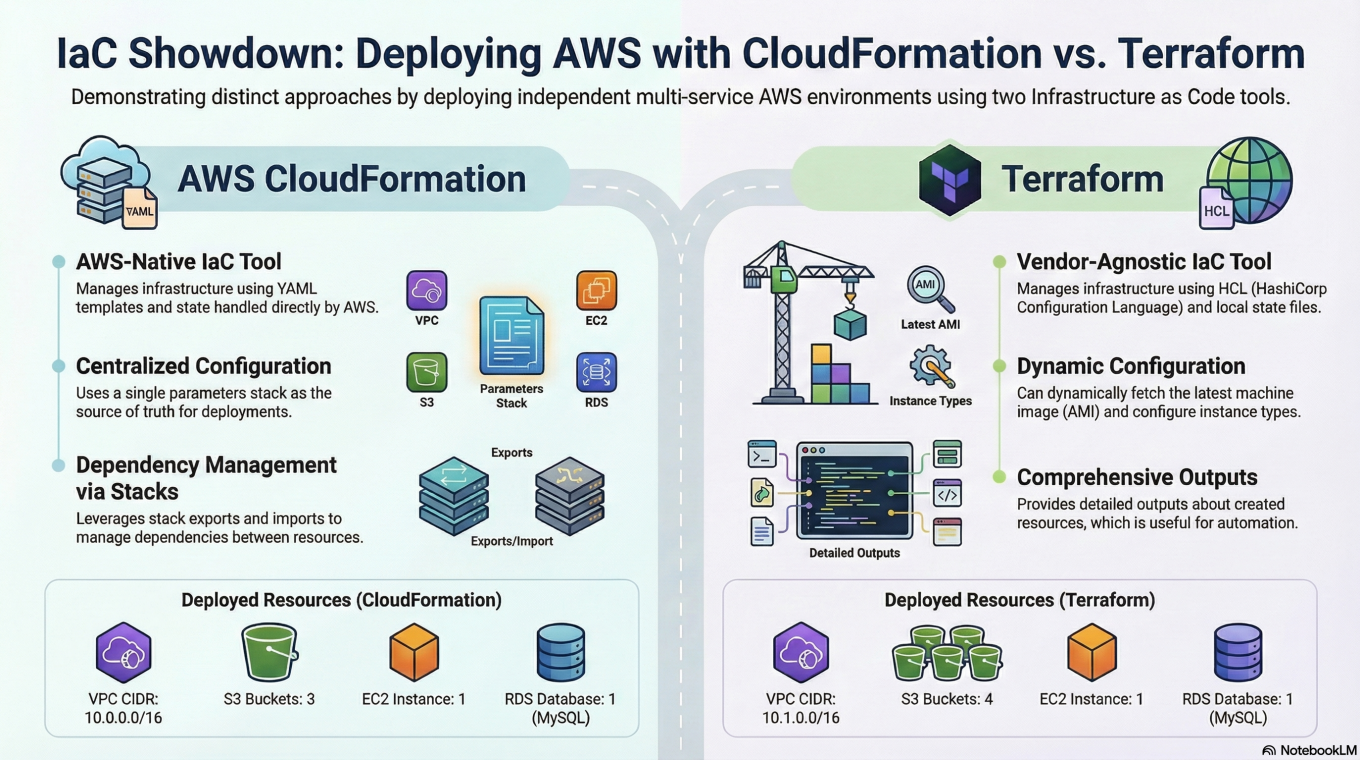
Overview

Deployed identical AWS infrastructure using CloudFormation and Terraform:

- **CloudFormation:** `cloudinfra-finalproject-cf-9026254-` (VPC: 10.0.0.0/16)
- **Terraform:** `cloudinfra-finalproject-terra-9026254-` (VPC: 10.1.0.0/16)

Resources: S3 buckets, VPC + subnets, EC2 instance, RDS MySQL database

Architecture



Network Layout:

- 1 Public subnet (EC2)
- 2 Private subnets in different AZs (RDS)
- Internet Gateway for public access
- Security groups for EC2 (SSH) and RDS (MySQL)

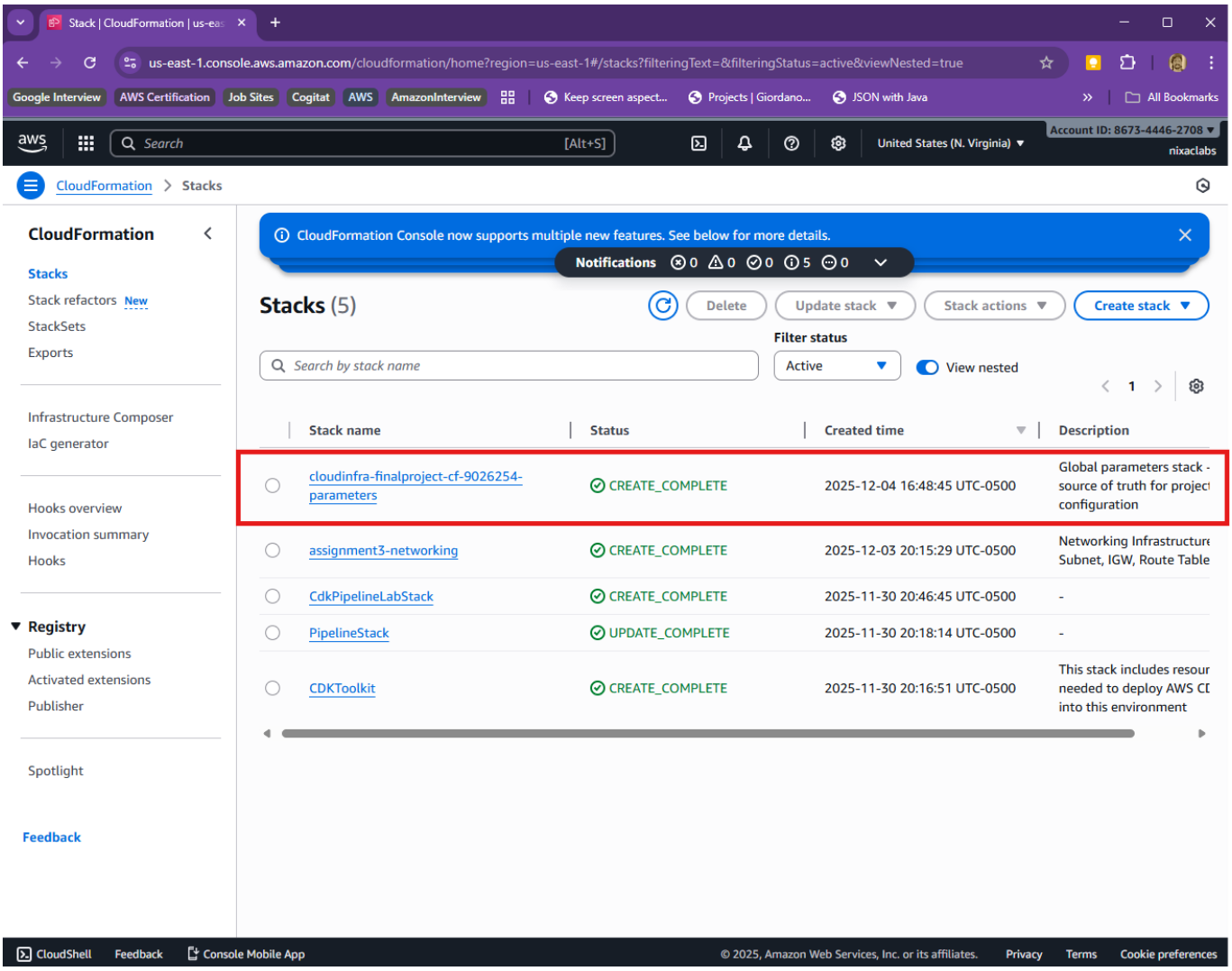
Code Structure

CloudFormation

```
cloudformation/
├── parameters.yaml      # Global config (single source of truth)
├── s3-buckets.yaml     # Storage
├── network.yaml        # VPC/subnets
├── ec2-instance.yaml   # Compute
└── rds-instance.yaml   # Database
```

Key Feature: Cross-Stack References

```yaml
# parameters.yaml exports prefix
Outputs:
  ProjectPrefix:
    Export:
      Name: 'GlobalProjectPrefix'

# Other stacks import it
Parameters:
  ProjectPrefix:
    Default: !ImportValue GlobalProjectPrefix
```

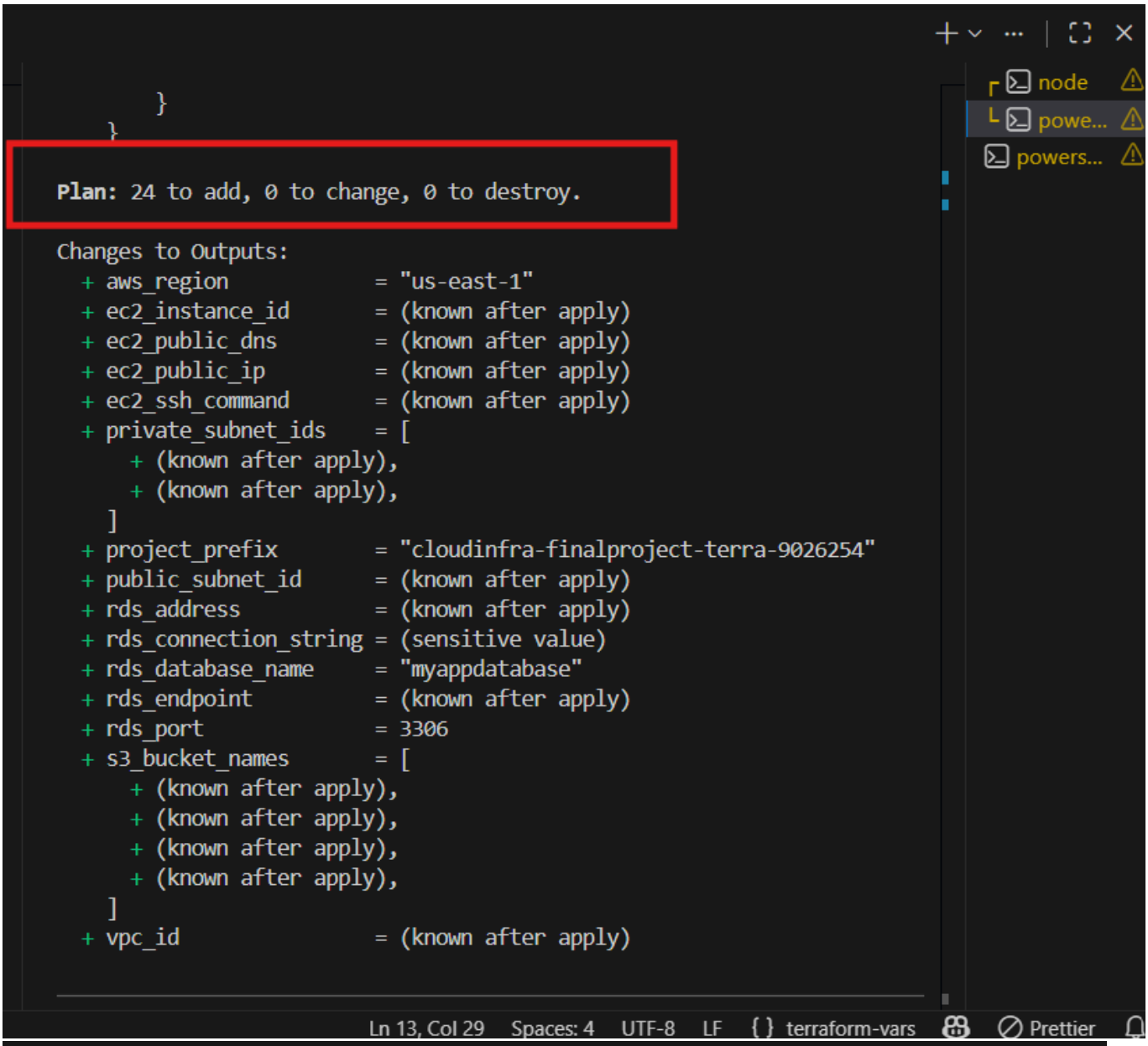


Terraform

```
terraform/
├── variables.tf    # Variable definitions
├── terraform.tfvars # Values (not in Git)
├── s3.tf           # Storage
├── vpc.tf          # VPC/subnets
├── ec2.tf          # Compute
├── rds.tf          # Database
└── outputs.tf      # Connection details
```

Key Feature: Dynamic AMI

```
data "aws_ami" "amazon_linux_2" {
  most_recent = true
  owners      = ["amazon"]
  filter {
    name   = "name"
    values = ["amzn2-ami-hvm-*-x86_64-gp2"]
  }
}
```

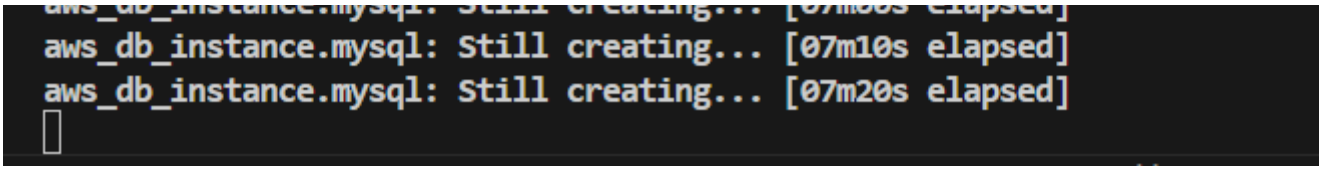


```
        }
    }

Plan: 24 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + aws_region            = "us-east-1"
  + ec2_instance_id       = (known after apply)
  + ec2_public_dns        = (known after apply)
  + ec2_public_ip         = (known after apply)
  + ec2_ssh_command       = (known after apply)
  + private_subnet_ids    = [
      + (known after apply),
      + (known after apply),
    ]
  + project_prefix        = "cloudinfra-finalproject-terra-9026254"
  + public_subnet_id      = (known after apply)
  + rds_address           = (known after apply)
  + rds_connection_string = (sensitive value)
  + rds_database_name     = "myappdatabase"
  + rds_endpoint          = (known after apply)
  + rds_port              = 3306
  + s3_bucket_names       = [
      + (known after apply),
      + (known after apply),
      + (known after apply),
      + (known after apply),
    ]
  + vpc_id                = (known after apply)
```

node
powe...
powers...

Ln 13, Col 29 Spaces: 4 UTF-8 LF { } terraform-vars Prettier

```
aws_db_instance.mysql: Still creating... [04m20s elapsed]
aws_db_instance.mysql: Still creating... [04m30s elapsed]
aws_db_instance.mysql: Still creating... [04m40s elapsed]
aws_db_instance.mysql: Still creating... [04m50s elapsed]
aws_db_instance.mysql: Still creating... [05m00s elapsed]
aws_db_instance.mysql: Still creating... [05m10s elapsed]
aws_db_instance.mysql: Still creating... [05m20s elapsed]
aws_db_instance.mysql: Still creating... [05m30s elapsed]
aws_db_instance.mysql: Still creating... [05m40s elapsed]
aws_db_instance.mysql: Still creating... [05m50s elapsed]
aws_db_instance.mysql: Still creating... [06m00s elapsed]
aws_db_instance.mysql: Still creating... [06m10s elapsed]
aws_db_instance.mysql: Still creating... [06m20s elapsed]
aws_db_instance.mysql: Still creating... [06m30s elapsed]
aws_db_instance.mysql: Still creating... [06m40s elapsed]
aws_db_instance.mysql: Still creating... [06m50s elapsed]
aws_db_instance.mysql: Still creating... [07m00s elapsed]
```



Modularity & Best Practices

CloudFormation

- **5 separate stacks:** Parameters → S3 → Network → EC2 → RDS
- **Exports/Imports:** Stack outputs used by other stacks
- **Benefits:** Update one stack independently, clear dependencies

Terraform

- **File separation:** Each service in separate `.tf` file
- **Centralized variables:** Single source in `variables.tf`
- **Benefits:** Easy to find/modify, single deployment command

Both

☑ Consistent naming with configurable prefix ☑ No hardcoded values ☑ Security: RDS in private subnets, S3 public access blocked ☑ Sensitive data excluded from Git

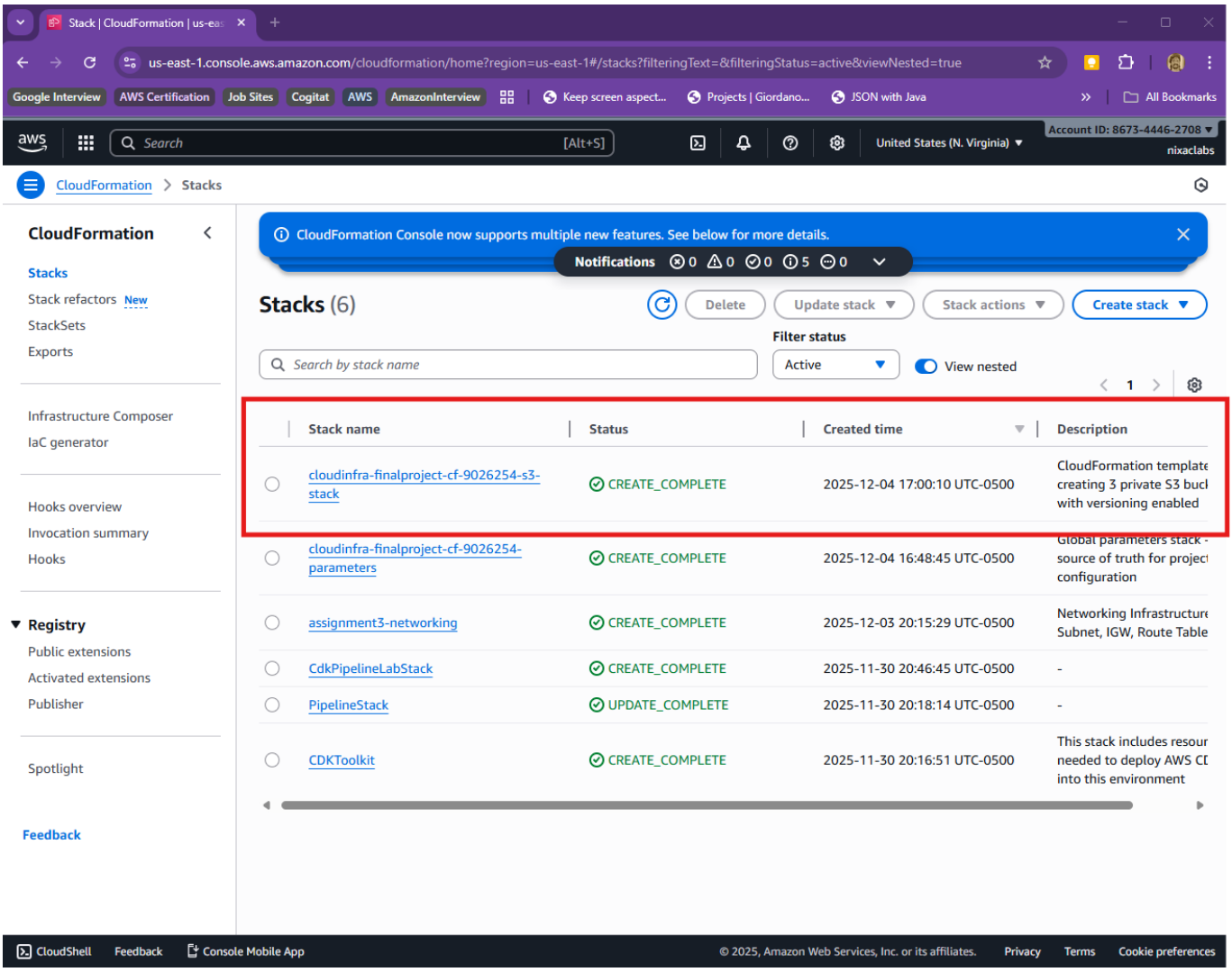
---

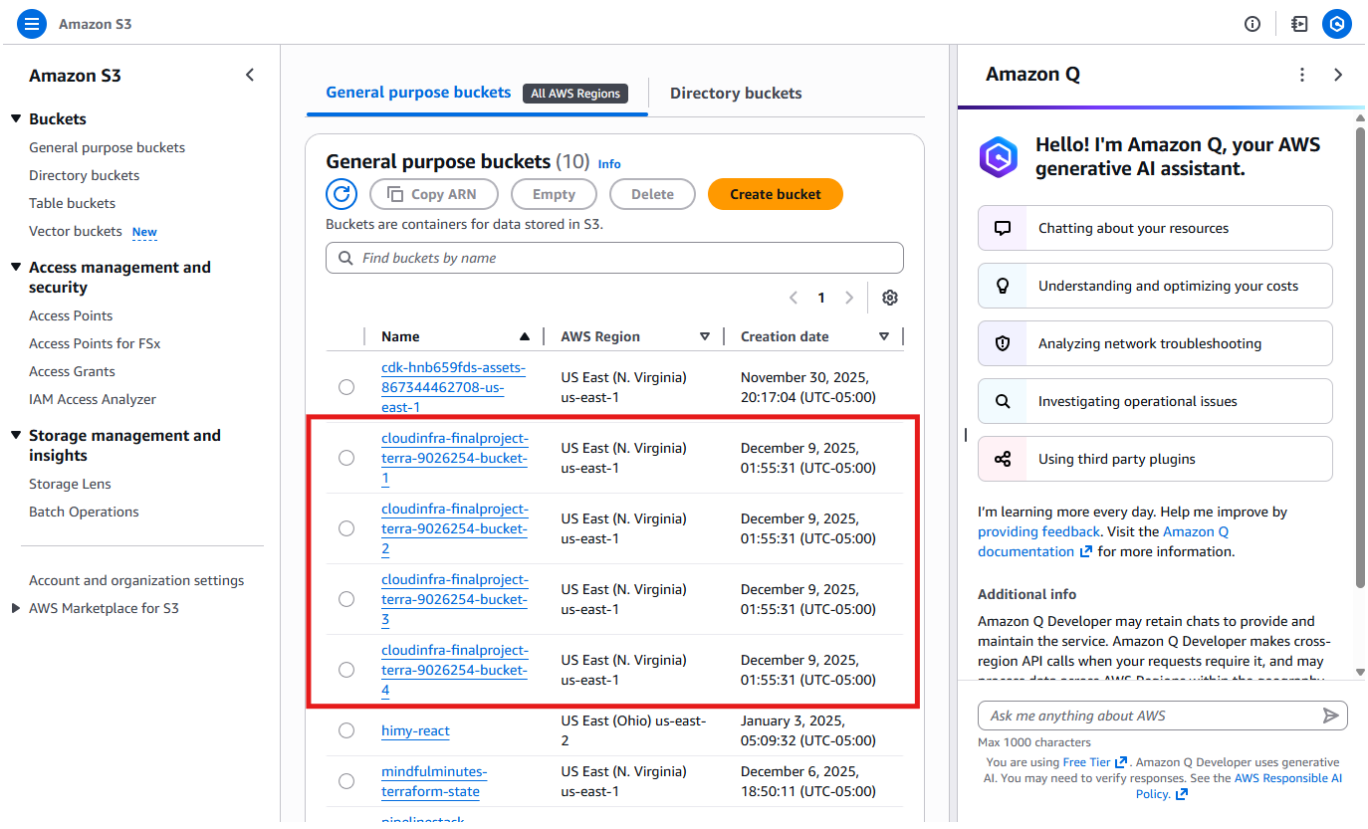
Deployed Resources

S3 Buckets

CloudFormation: 3 buckets | **Terraform:** 4 buckets

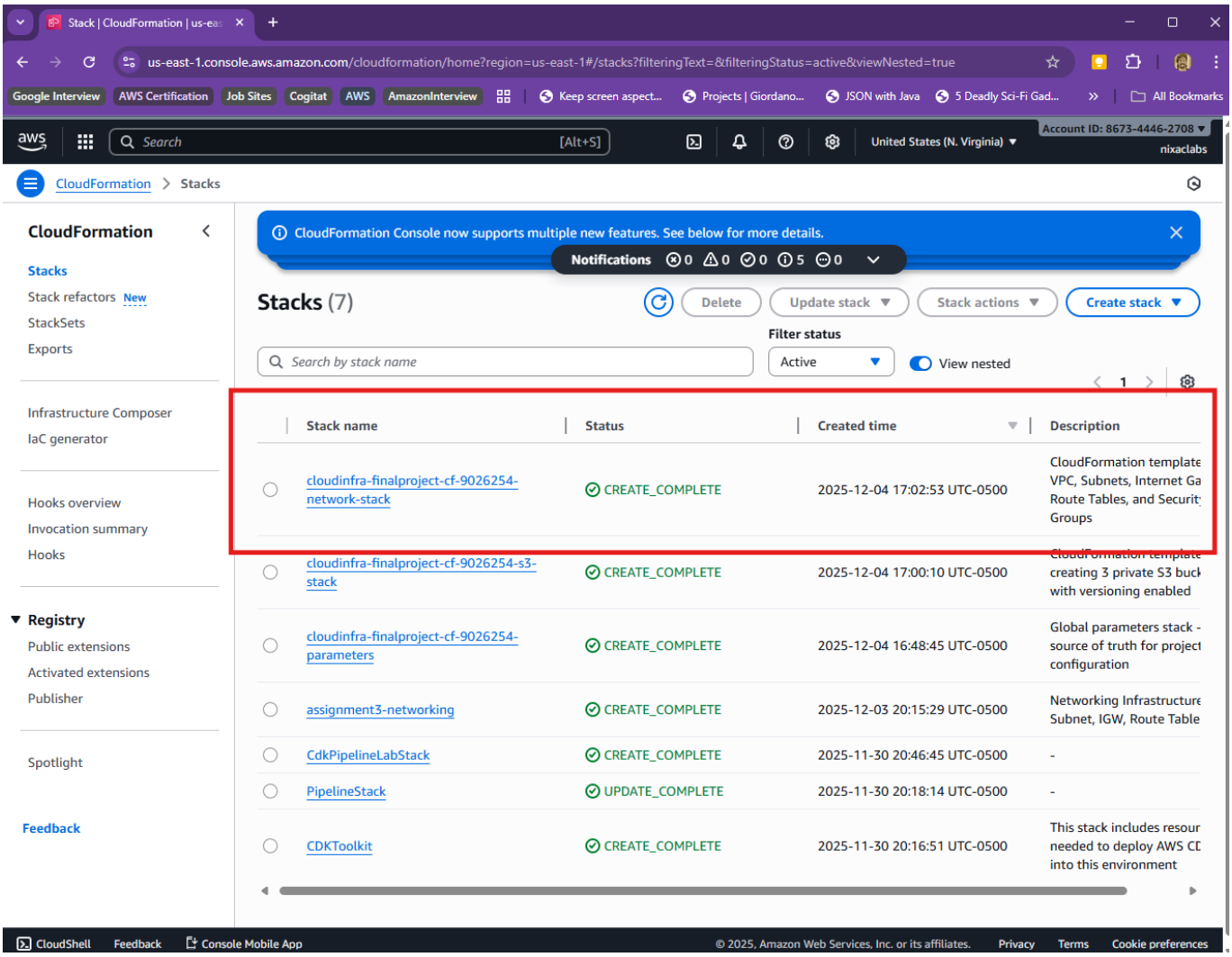
- Versioning enabled
- Public access blocked





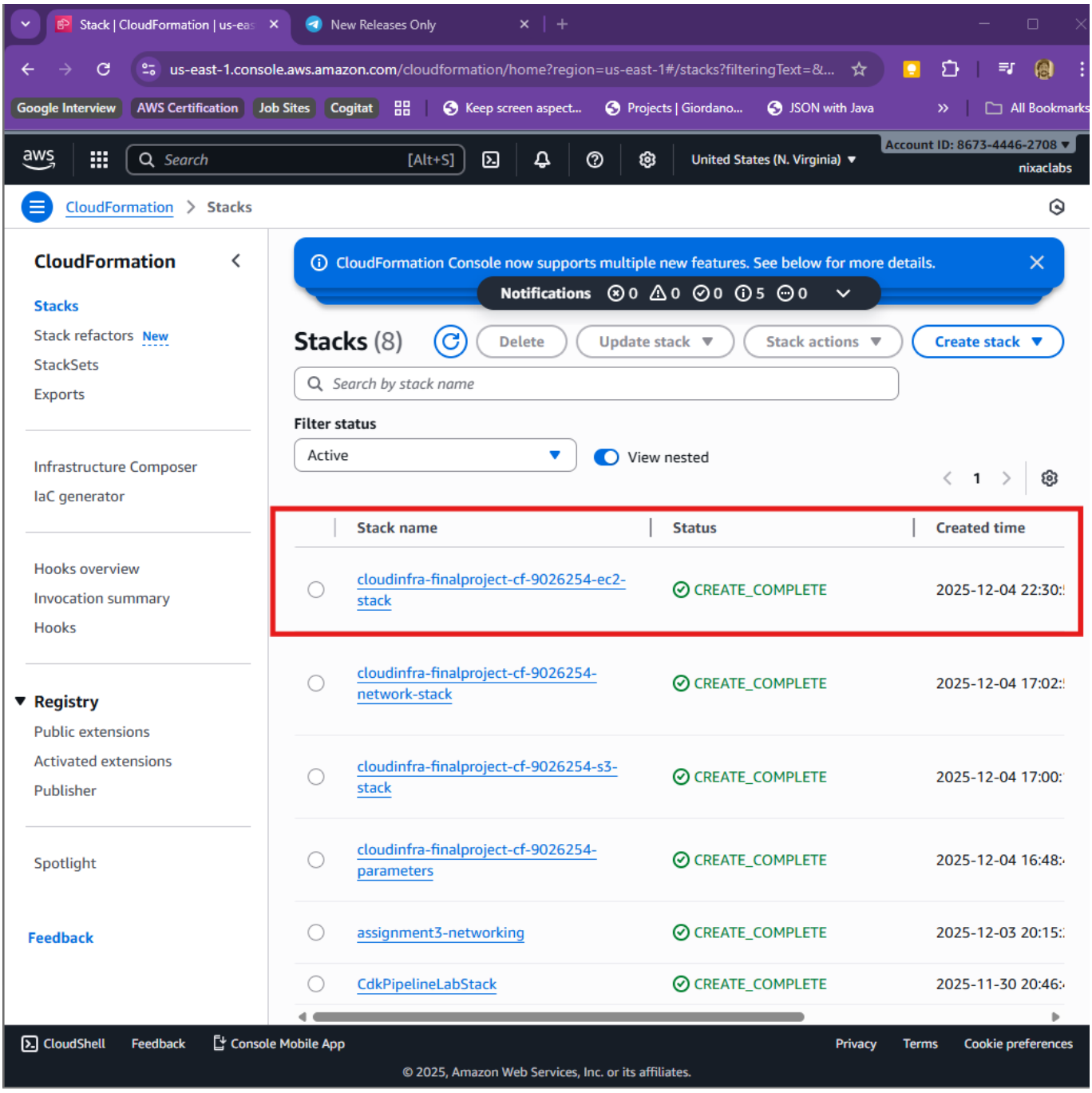
VPC & Networking

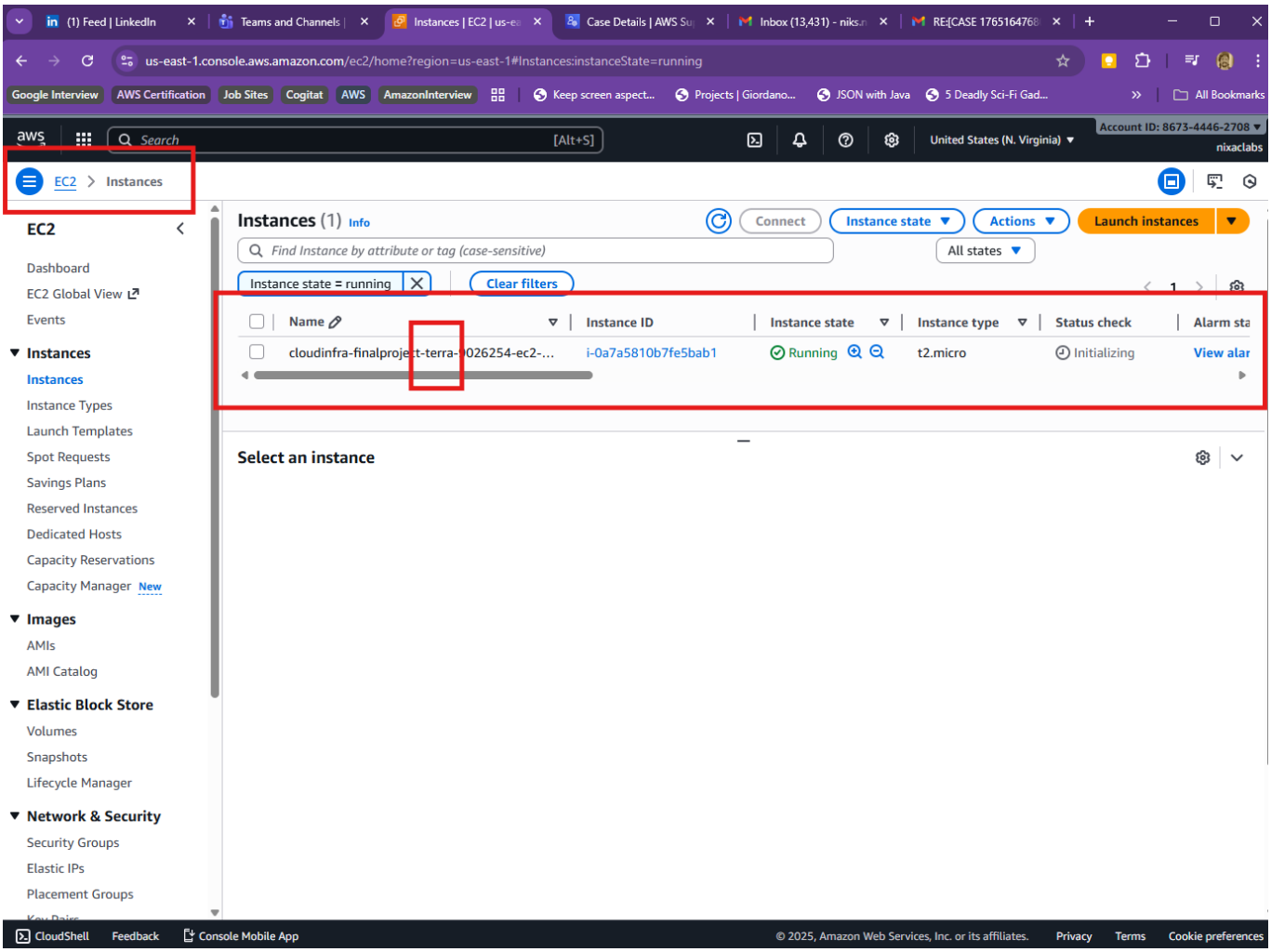
- VPC with Internet Gateway
- Public subnet for EC2
- 2 private subnets (different AZs) for RDS



EC2 Instance

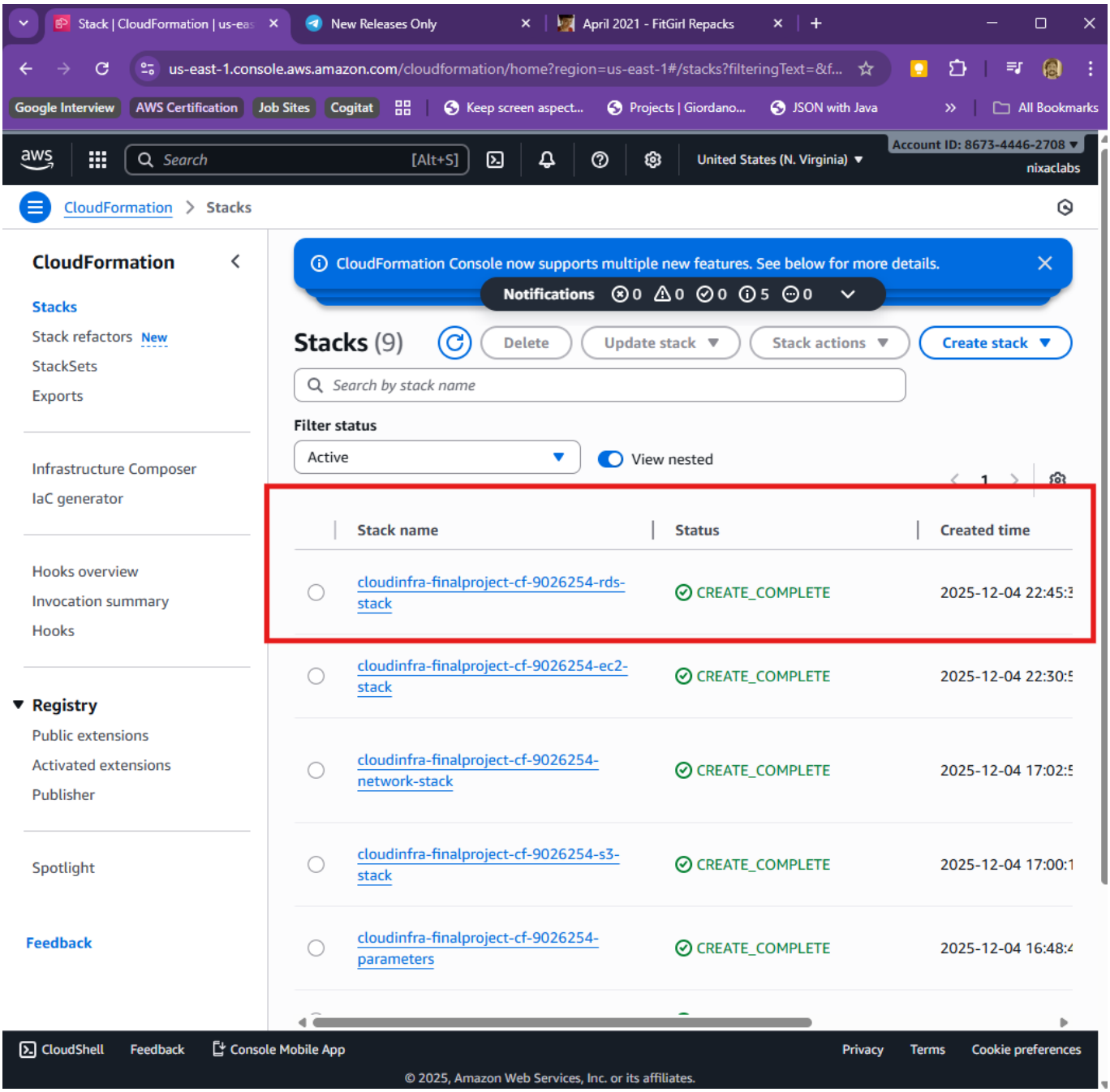
- Type: t2.micro (Free Tier)
- AMI: Amazon Linux 2
- Public subnet with public IP
- SSH access (port 22)

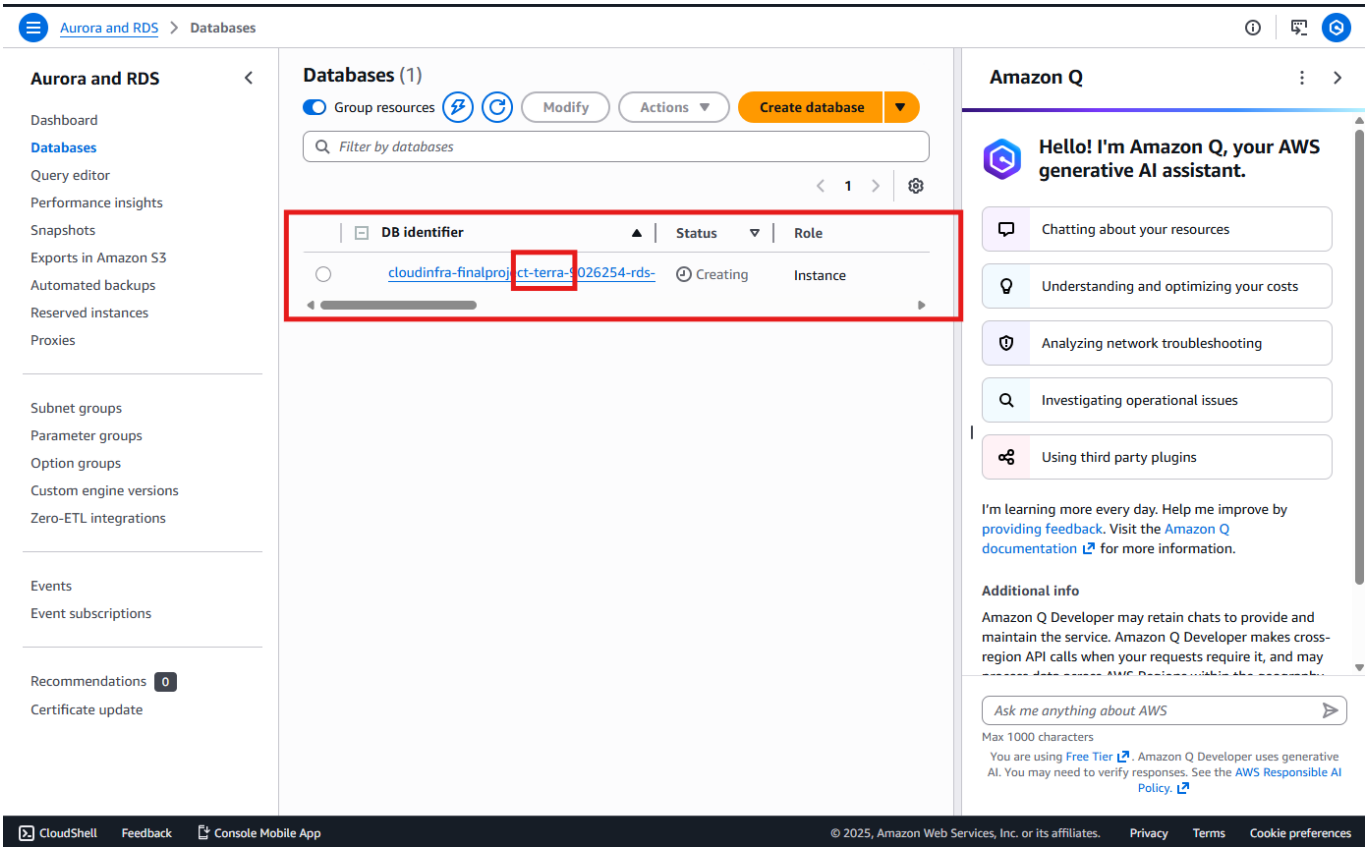

Stack | CloudFormation | us-ea
New Releases Only
us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks?filteringText=&...
Google Interview
AWS Certification
Job Sites
Cogitat
Keep screen aspect...
Projects | Giordano...
JSON with Java
All Bookmarks
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 8673-4446-2708
nixaclabs
CloudFormation > Stacks
CloudFormation
Stacks
Stack refactors New
StackSets
Exports
Infrastructure Composer
IaC generator
Hooks overview
Invocation summary
Hooks
Registry
Public extensions
Activated extensions
Publisher
Spotlight
Feedback
CloudFormation Console now supports multiple new features. See below for more details.
Notifications 0 0 0 5 0
Stacks (8)
Delete
Update stack
Stack actions
Create stack
Search by stack name
Filter status
Active
View nested
1
Stack name
Status
Created time
cloudinfra-finalproject-cf-9026254-ec2-stack
CREATE_COMPLETE
2025-12-04 22:30:
cloudinfra-finalproject-cf-9026254-network-stack
CREATE_COMPLETE
2025-12-04 17:02:
cloudinfra-finalproject-cf-9026254-s3-stack
CREATE_COMPLETE
2025-12-04 17:00:
cloudinfra-finalproject-cf-9026254-parameters
CREATE_COMPLETE
2025-12-04 16:48:
assignment3-networking
CREATE_COMPLETE
2025-12-03 20:15:
CdkPipelineLabStack
CREATE_COMPLETE
2025-11-30 20:46:
CloudShell
Feedback
Console Mobile App
Privacy
Terms
Cookie preferences
© 2025, Amazon Web Services, Inc. or its affiliates.



RDS Database

- Engine: MySQL 8.0
- Instance: db.t3.micro (Free Tier)
- Private subnets only (not publicly accessible)


Stack | CloudFormation | us-ea
New Releases Only
April 2021 - FitGirl Repacks
us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks?filteringText=&f...
Google Interview
AWS Certification
Job Sites
Cogitat
Keep screen aspect...
Projects | Giordano...
JSON with Java
All Bookmarks
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 8673-4446-2708
nixaclabs
CloudFormation
Stacks
CloudFormation
Stacks
Stack refactors New
StackSets
Exports
Infrastructure Composer
IaC generator
Hooks overview
Invocation summary
Hooks
Registry
Public extensions
Activated extensions
Publisher
Spotlight
Feedback
CloudFormation Console now supports multiple new features. See below for more details.
Notifications 0 0 0 5 0
Stacks (9)
Delete
Update stack
Stack actions
Create stack
Search by stack name
Filter status
Active
View nested
Stack name
Status
Created time
cloudinfra-finalproject-cf-9026254-rds-stack
CREATE_COMPLETE
2025-12-04 22:45:3
cloudinfra-finalproject-cf-9026254-ec2-stack
CREATE_COMPLETE
2025-12-04 22:30:5
cloudinfra-finalproject-cf-9026254-network-stack
CREATE_COMPLETE
2025-12-04 17:02:5
cloudinfra-finalproject-cf-9026254-s3-stack
CREATE_COMPLETE
2025-12-04 17:00:1
cloudinfra-finalproject-cf-9026254-parameters
CREATE_COMPLETE
2025-12-04 16:48:4
CloudShell
Feedback
Console Mobile App
Privacy
Terms
Cookie preferences
© 2025, Amazon Web Services, Inc. or its affiliates.



Deployment

CloudFormation (Sequential):

```
aws cloudformation create-stack --stack-name ...-params-stack --template-body
file://parameters.yaml
aws cloudformation create-stack --stack-name ...-s3-stack --template-body
file://s3-buckets.yaml
aws cloudformation create-stack --stack-name ...-network-stack --template-body
file://network.yaml
aws cloudformation create-stack --stack-name ...-ec2-stack --template-body
file://ec2-instance.yaml
aws cloudformation create-stack --stack-name ...-rds-stack --template-body
file://rds-instance.yaml
```

Terraform (Single Command):

```
terraform init
terraform plan
terraform apply
```

Challenges & Solutions

Challenge	Solution
Stack dependency order	Created parameters stack first, used Exports/Imports
RDS needs 2+ AZs	Created 2 private subnets in different AZs
Key pair must pre-exist	Created in AWS Console before deployment
Sensitive data security	Used .gitignore for tfvars, NoEcho for CloudFormation

---

Key Learnings

Technical:

- IaC modularity patterns
- AWS service dependencies
- State management (CloudFormation: AWS-managed, Terraform: local file)

Best Practices:

- Separate concerns (one service per file/stack)
- Centralized configuration
- Security-first design
- Version control with sensitive data protection

Tool Comparison:

- **CloudFormation:** AWS-native, auto state management, sequential stacks
- **Terraform:** Multi-cloud, HCL syntax, single deployment

Conclusion

Successfully deployed identical infrastructure with two IaC tools, demonstrating:

- Modular, maintainable code
- Security best practices
- Production-ready templates

Both tools achieve the same result with different approaches. Choice depends on use case: CloudFormation for AWS-only, Terraform for multi-cloud flexibility.

Project Repository: [GitHub URL] **Student ID:** 9026254