Module 8: Case Study - 1

Assignment Submitted By:-Hitesh Chauhan
Course Offered: -Advanced Cloud Computing and Devops
Assignment By: -Intellipaat
Trainer: -Puneet Gavri
Date Of Submission: -19/11/2024

Problem Statement:

You work for XYZ Corporation. Your corporation wants to launch a new web-based application. The development team has prepared the code but it is not tested yet. The development team needs the system admins to build a web server to test the code but the system admins are not available.

Tasks To Be Performed:

1. Web tier: Launch an instance in a public subnet and that instance should allow HTTP and SSH from the internet.
2. Application tier: Launch an instance in a private subnet of the web tier and it should allow only SSH from the public subnet of Web Tier-3.
3. DB tier: Launch an RDS MYSQL instance in a private subnet and it should allow connection on port 3306 only from the private subnet of Application Tier-4.
4. Setup a Route 53 hosted zone and direct traffic to the EC2 instance.

You have been also asked to propose a solution so that:

1. Development team can test their code without having to involve the system admins and can invest their time in testing the code rather than provisioning, configuring and updating the resources needed to test the code.
2. Make sure when the development team deletes the stack, RDS DB instances should not be deleted.

ANSWER:

1. Web tier: Launch an instance in a public subnet and that instance should allow HTTP and SSH from the internet

```yaml
WebTierInstance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: t2.micro
    ImageId: !Ref AMI
    SubnetId: !Ref PublicSubnet1
    SecurityGroupIds:
      - !Ref WebTierSecurityGroup

WebTierSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow HTTP and SSH from the internet
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0
```

2. Application tier: Launch an instance in a private subnet of the web tier and it should allow only SSH from the public subnet of Web Tier-3

```yaml
ApplicationTierInstance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: t2.micro
    ImageId: !Ref AMI
    SubnetId: !Ref PrivateSubnet1
    SecurityGroupIds:
      - !Ref ApplicationTierSecurityGroup

ApplicationTierSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow SSH from Web Tier
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        SourceSecurityGroupId: !Ref WebTierSecurityGroup
```

3. DB tier: Launch an RDS MYSQL instance in a private subnet and it should allow connection on port 3306 only from the private subnet of Application Tier-4.

```yaml
InstanceSecurityGroupDataBase:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow access to the database
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 3306
        ToPort: 3306
        SourceSecurityGroupId: !Ref ApplicationTierSecurityGroup

RDSDBSubnetGroup:
  Type: AWS::RDS::DBSubnetGroup
  Properties:
    DBSubnetGroupDescription: Subnet Group for RDS
    SubnetIds:
      - !Ref PrivateSubnet1
      - !Ref PrivateSubnet2
```

```yaml
RDSDBInstance:
  Type: AWS::RDS::DBInstance
  Properties:
    AllocatedStorage: 20
    AutoMinorVersionUpgrade: true
    DBInstanceClass: db.t3.micro
    DBInstanceIdentifier: !Ref DBName
    DBName: !Ref DBName
    Engine: MySQL
    EngineVersion: "8.0.33"
    MasterUsername: !Ref DBUsername
    MasterUserPassword: !Ref DBPassword
    PubliclyAccessible: false
    StorageType: gp3
    MultiAZ: false
    Port: "3306"
    StorageEncrypted: false
    CopyTagsToSnapshot: true
    EnableIAMDatabaseAuthentication: false
    VPCSecurityGroups:
      - !Ref InstanceSecurityGroupDataBase
    DBSubnetGroupName: !Ref RDSDBSubnetGroup
    DeletionProtection: false
    MaxAllocatedStorage: 1000
    Tags:
      - Key: Name
        Value: DBAssessment
      - Key: createdBy
        Value: Hitesh Chauhan
      - Key: Project
        Value: CloudFormationAssignment01
      - Key: Environment
        Value: Prod
```

4. Setup a Route 53 hosted zone and direct traffic to the EC2 instance.

```yaml
  HostedZone01:
    Type: AWS::Route53::HostedZone
    Properties:
      HostedZoneConfig:
        Comment: ""
      Name: xyzcorporation.com
  MyDNSRecord:
    Type: AWS::Route53::RecordSet
    Properties:
      HostedZoneId: !Ref HostedZone01
      Name: webtier3.xyzcorporation.com
      Type: A
      TTL: 300
      ResourceRecords:
        - !GetAtt WebTierInstance.PublicIp
```

Full Code:

```yaml
Parameters:
  EnvironmentName:
    Description: An environment name that is prefixed to resource names
    Type: String

  #db
  DBInstanceClass:
    Description: RDS instance class
    Type: String
    Default: db.t2.micro
    AllowedValues:
      - db.t2.micro
      - db.t2.small
    ConstraintDescription: Must be a valid RDS instance class.

  DBName:
    Description: The name of the database
    Type: String
    Default: mydatabase

  DBUsername:
    Description: The database admin account username
    Type: String
    Default: admin

  DBPassword:
    Description: The database admin account password
    Type: String
    NoEcho: true

  #route53
  DomainName:
    Description: The domain name for the Route 53 hosted zone
    Type: String
    Default: xyzcorporation.com
```

```
  HostedZone:
    Type: String
    Description: The DNS name of an existing Amazon Route 53 hosted zone

  AMI:
    Type: String
    Description: AMI ID for the web and application tier instances
    Default: ami-063d43db0594b521b
```

```
Resources:
  VPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 10.26.0.0/16
      EnableDnsSupport: true
      EnableDnsHostnames: true
      Tags:
        - Key: Name
          Value: !Ref EnvironmentName

  InternetGateway:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: !Ref EnvironmentName

  InternetGatewayAttachment:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      InternetGatewayId: !Ref InternetGateway
      VpcId: !Ref VPC
```

```yaml
PublicSubnet1:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref VPC
    AvailabilityZone: !Select [0, !GetAZs ""]
    CidrBlock: 10.26.0.0/24
    MapPublicIpOnLaunch: true
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Public Subnet (AZ1)

PrivateSubnet1:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref VPC
    CidrBlock: 10.26.1.0/24
    AvailabilityZone: us-east-1a
    MapPublicIpOnLaunch: false
    Tags:
      - Key: Name
        Value: Private

PrivateSubnet2:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref VPC
    CidrBlock: 10.26.2.0/24
    AvailabilityZone: us-east-1b
    MapPublicIpOnLaunch: false
    Tags:
      - Key: Name
        Value: Private2
```

```yaml
PublicRouteTable:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Public Routes

DefaultPublicRoute:
  Type: AWS::EC2::Route
  DependsOn: InternetGatewayAttachment
  Properties:
    RouteTableId: !Ref PublicRouteTable
    DestinationCidrBlock: 0.0.0.0/0
    GatewayId: !Ref InternetGateway

PublicSubnet1RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref PublicRouteTable
    SubnetId: !Ref PublicSubnet1

PrivateRouteTable1:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Private Routes (AZ1)
```

```yaml
WebTierInstance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: t2.micro
    ImageId: !Ref AMI
    SubnetId: !Ref PublicSubnet1
    SecurityGroupIds:
      - !Ref WebTierSecurityGroup

WebTierSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow HTTP and SSH from the internet
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0
```

```yaml
ApplicationTierInstance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: t2.micro
    ImageId: !Ref AMI
    SubnetId: !Ref PrivateSubnet1
    SecurityGroupIds:
      - !Ref ApplicationTierSecurityGroup

ApplicationTierSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow SSH from Web Tier
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        SourceSecurityGroupId: !Ref WebTierSecurityGroup

InstanceSecurityGroupDataBase:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow access to the database
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 3306
        ToPort: 3306
        SourceSecurityGroupId: !Ref ApplicationTierSecurityGroup
```

```yaml
RDSDBSubnetGroup:
  Type: AWS::RDS::DBSubnetGroup
  Properties:
    DBSubnetGroupDescription: Subnet Group for RDS
    SubnetIds:
      - !Ref PrivateSubnet1
      - !Ref PrivateSubnet2

RDSDBInstance:
  Type: AWS::RDS::DBInstance
  Properties:
    AllocatedStorage: 20
    AutoMinorVersionUpgrade: true
    DBInstanceClass: db.t3.micro
    DBInstanceIdentifier: !Ref DBName
    DBName: !Ref DBName
    Engine: MySQL
    EngineVersion: "8.0.33"
    MasterUsername: !Ref DBUsername
    MasterUserPassword: !Ref DBPassword
    PubliclyAccessible: false
    StorageType: gp3
    MultiAZ: false
    Port: "3306"
    StorageEncrypted: false
    CopyTagsToSnapshot: true
    EnableIAMDatabaseAuthentication: false
    VPCSecurityGroups:
      - !Ref InstanceSecurityGroupDataBase
    DBSubnetGroupName: !Ref RDSDBSubnetGroup
    DeletionProtection: false
    MaxAllocatedStorage: 1000
```

```
      MaxAllocatedStorage: 1000
      Tags:
        - Key: Name
          Value: DBAssessment
        - Key: createdBy
          Value: Hitesh Chauhan
        - Key: Project
          Value: CloudFormationAssignment01
        - Key: Environment
          Value: Prod

  HostedZone01:
    Type: AWS::Route53::HostedZone
    Properties:
      HostedZoneConfig:
        Comment: ""
      Name: xyzcorporation.com
  MyDNSRecord:
    Type: AWS::Route53::RecordSet
    Properties:
      HostedZoneId: !Ref HostedZone01
      Name: webtier3.xyzcorporation.com
      Type: A
      TTL: 300
      ResourceRecords:
        - !GetAtt WebTierInstance.PublicIp
```

This is the full code of this task performed.

Parameters:

EnvironmentName:

Description: An environment name that is prefixed to resource names

Type: String

#db

DBInstanceClass:

Description: RDS instance class

Type: String

Default: db.t2.micro

AllowedValues:

- db.t2.micro

- db.t2.small

ConstraintDescription: Must be a valid RDS instance class.

DBName:

Description: The name of the database

Type: String

Default: mydatabase

DBUsername:

Description: The database admin account username

Type: String

Default: admin

DBPassword:

Description: The database admin account password

Type: String

NoEcho: true

#route53

DomainName:

Description: The domain name for the Route 53 hosted zone

Type: String

Default: xyzcorporation.com

HostedZone:

Type: String

Description: The DNS name of an existing Amazon Route 53 hosted zone

AMI:

Type: String

Description: AMI ID for the web and application tier instances

Default: ami-063d43db0594b521b

Mappings:

RegionMap:

us-east-1:

AMI: ami-7f418316

us-west-1:

AMI: ami-951945d0

us-west-2:

AMI: ami-16fd7026

eu-west-1:

AMI: ami-24506250

sa-east-1:

AMI: ami-3e3be423

ap-southeast-1:

AMI: ami-74dda626

ap-southeast-2:

AMI: ami-b3990e89

ap-northeast-1:

AMI: ami-dcfa4edd

```
Resources:
  VPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 10.26.0.0/16
      EnableDnsSupport: true
      EnableDnsHostnames: true
      Tags:
        - Key: Name
          Value: !Ref EnvironmentName

  InternetGateway:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: !Ref EnvironmentName

  InternetGatewayAttachment:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      InternetGatewayId: !Ref InternetGateway
      VpcId: !Ref VPC

  PublicSubnet1:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref VPC
```

```
    AvailabilityZone: !Select [0, !GetAZs ""]
    CidrBlock: 10.26.0.0/24
    MapPublicIpOnLaunch: true
    Tags:
     - Key: Name
       Value: !Sub ${EnvironmentName} Public Subnet (AZ1)

  PrivateSubnet1:
   Type: AWS::EC2::Subnet
   Properties:
    VpcId: !Ref VPC
    CidrBlock: 10.26.1.0/24
    AvailabilityZone: us-east-1a
    MapPublicIpOnLaunch: false
    Tags:
     - Key: Name
       Value: Private

  PrivateSubnet2:
   Type: AWS::EC2::Subnet
   Properties:
    VpcId: !Ref VPC
    CidrBlock: 10.26.2.0/24
    AvailabilityZone: us-east-1b
    MapPublicIpOnLaunch: false
    Tags:
     - Key: Name
       Value: Private2
```

```
PublicRouteTable:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Public Routes

DefaultPublicRoute:
  Type: AWS::EC2::Route
  DependsOn: InternetGatewayAttachment
  Properties:
    RouteTableId: !Ref PublicRouteTable
    DestinationCidrBlock: 0.0.0.0/0
    GatewayId: !Ref InternetGateway

PublicSubnet1RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref PublicRouteTable
    SubnetId: !Ref PublicSubnet1

PrivateRouteTable1:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Private Routes (AZ1)
```

```
WebTierInstance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: t2.micro
    ImageId: !Ref AMI
    SubnetId: !Ref PublicSubnet1
    SecurityGroupIds:
      - !Ref WebTierSecurityGroup

WebTierSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow HTTP and SSH from the internet
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0

ApplicationTierInstance:
  Type: AWS::EC2::Instance
  Properties:
```

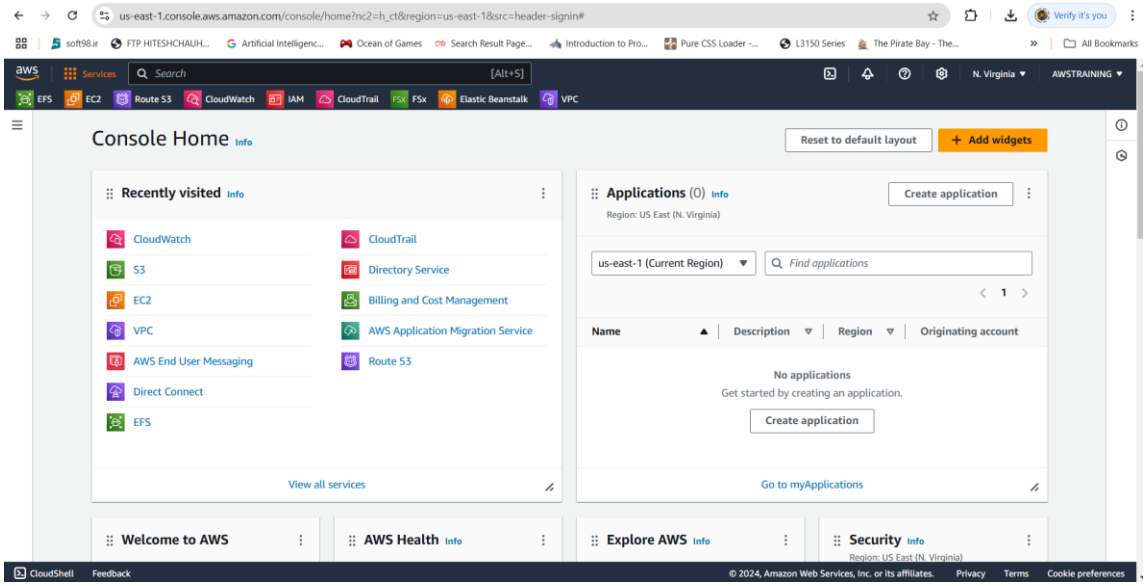
```
    InstanceType: t2.micro
    ImageId: !Ref AMI
    SubnetId: !Ref PrivateSubnet1
    SecurityGroupIds:
      - !Ref ApplicationTierSecurityGroup

  ApplicationTierSecurityGroup:
    Type: AWS::EC2::SecurityGroup
    Properties:
      GroupDescription: Allow SSH from Web Tier
      VpcId: !Ref VPC
      SecurityGroupIngress:
        - IpProtocol: tcp
          FromPort: 22
          ToPort: 22
          SourceSecurityGroupId: !Ref WebTierSecurityGroup

  InstanceSecurityGroupDataBase:
    Type: AWS::EC2::SecurityGroup
    Properties:
      GroupDescription: Allow access to the database
      VpcId: !Ref VPC
      SecurityGroupIngress:
        - IpProtocol: tcp
          FromPort: 3306
          ToPort: 3306
          SourceSecurityGroupId: !Ref ApplicationTierSecurityGroup
```

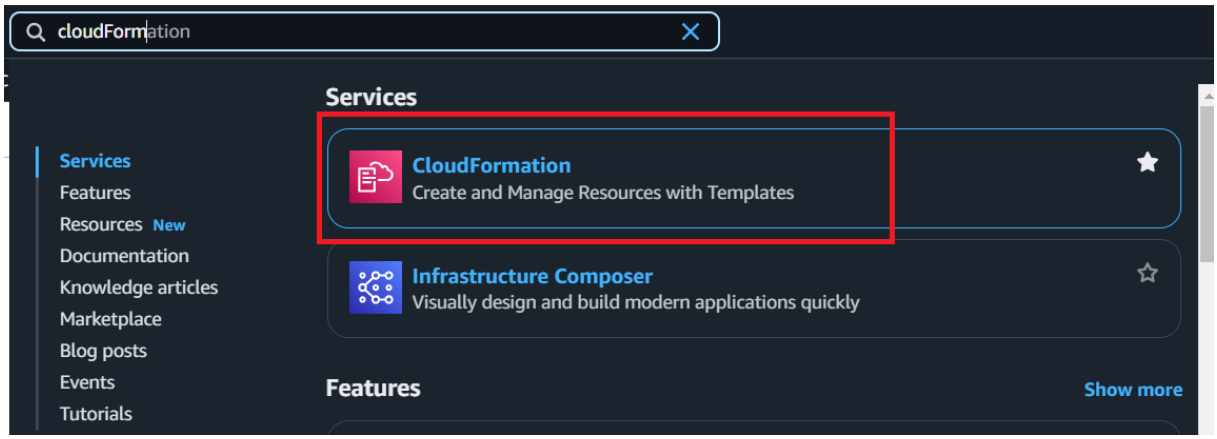
```
RDSDBSubnetGroup:
  Type: AWS::RDS::DBSubnetGroup
  Properties:
    DBSubnetGroupDescription: Subnet Group for RDS
    SubnetIds:
      - !Ref PrivateSubnet1
      - !Ref PrivateSubnet2

RDSDBInstance:
  Type: AWS::RDS::DBInstance
  Properties:
    AllocatedStorage: 20
    AutoMinorVersionUpgrade: true
    DBInstanceClass: db.t3.micro
    DBInstanceIdentifier: !Ref DBName
    DBName: !Ref DBName
    Engine: MySQL
    EngineVersion: "8.0.33"
    MasterUsername: !Ref DBUsername
    MasterUserPassword: !Ref DBPassword
    PubliclyAccessible: false
    StorageType: gp3
    MultiAZ: false
    Port: "3306"
    StorageEncrypted: false
    CopyTagsToSnapshot: true
    EnableIAMDatabaseAuthentication: false
    VPCSecurityGroups:
      - !Ref InstanceSecurityGroupDataBase
```

```
    DBSubnetGroupName: !Ref RDSDBSubnetGroup
    DeletionProtection: false
    MaxAllocatedStorage: 1000
    Tags:
     - Key: Name
       Value: DBAssessment
     - Key: createdBy
       Value: Hitesh Chauhan
     - Key: Project
       Value: CloudFormationAssignment01
     - Key: Environment
       Value: Prod

  HostedZone01:
   Type: AWS::Route53::HostedZone
   Properties:
    HostedZoneConfig:
     Comment: ""
    Name: xyzcorporation.com
  MyDNSRecord:
   Type: AWS::Route53::RecordSet
   Properties:
    HostedZoneId: !Ref HostedZone01
    Name: webtier3.xyzcorporation.com
    Type: A
    TTL: 300
    ResourceRecords:
     - !GetAtt WebTierInstance.PublicIp
```

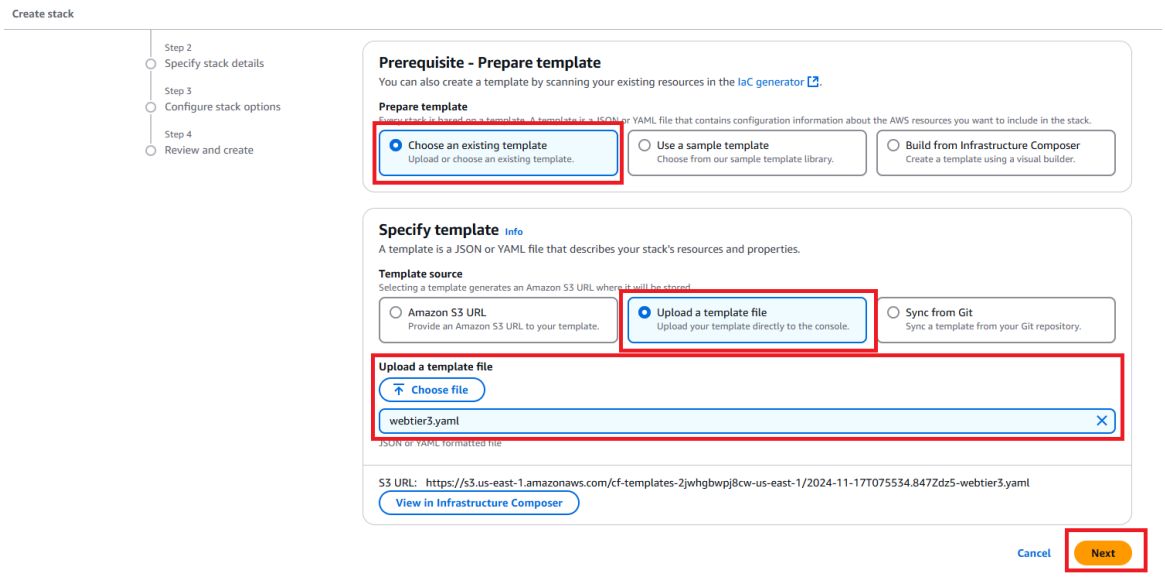
Then After Complete the code and need to upload this code to cloud formation so this code will create the infrastructure with the help **.yml** file

Login to AWS Management Console then Search Cloud Formation





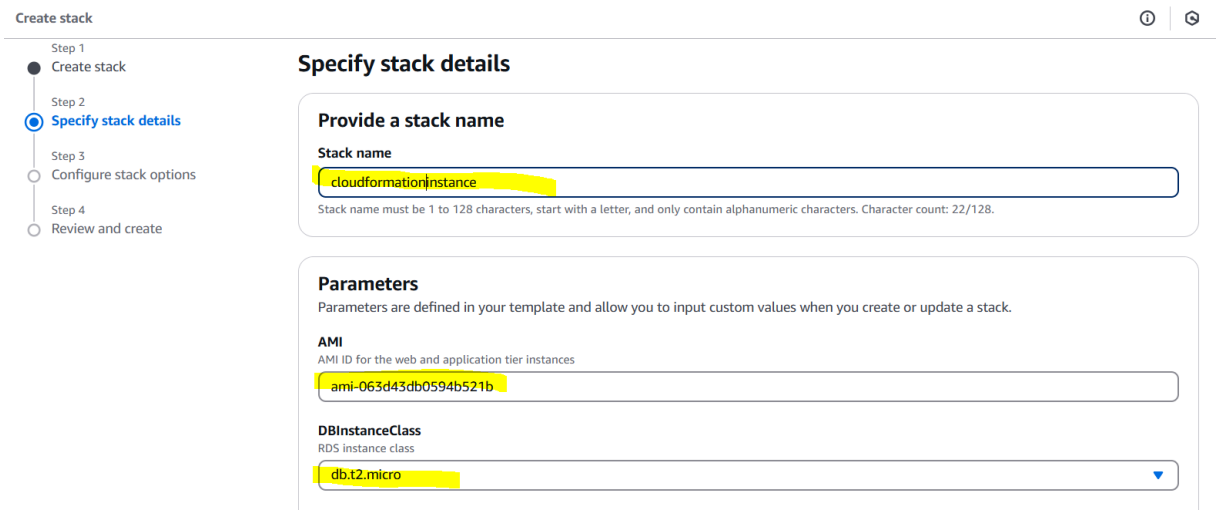
Here we need to upload the template as we created so in this case we need to choose existing template then specify template there is 3 option but we need to choose upload the template. This template through create the infrastructure.

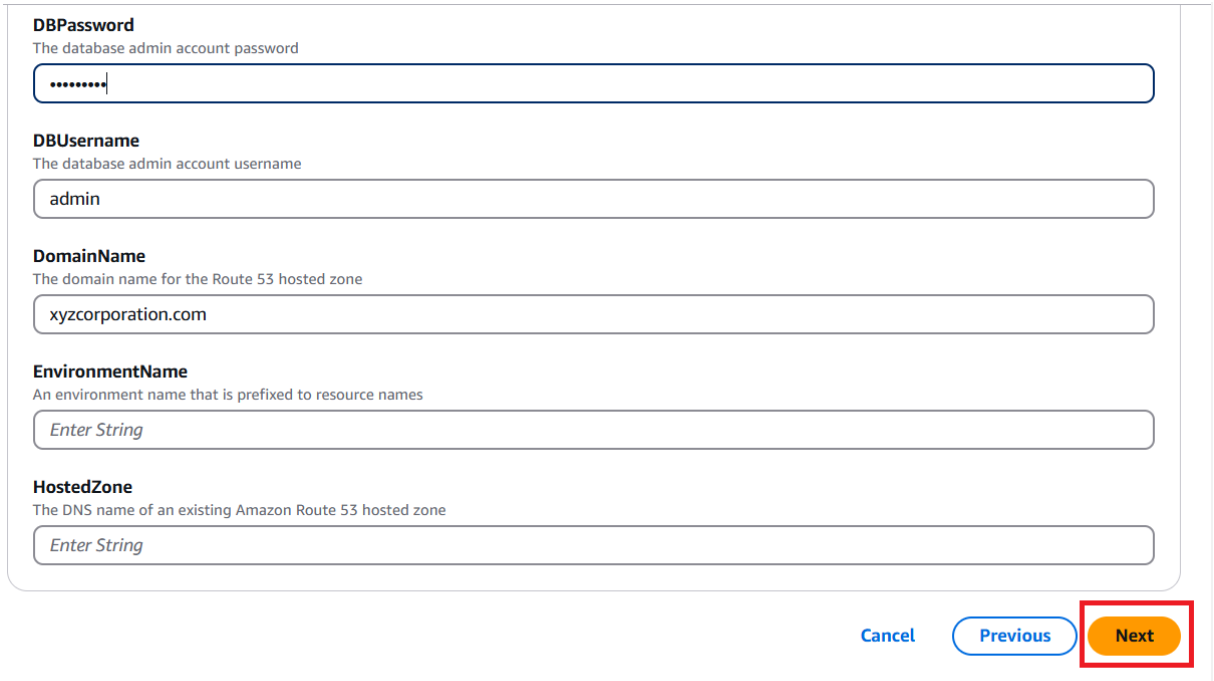


After upload the template next step will be provide a stack name

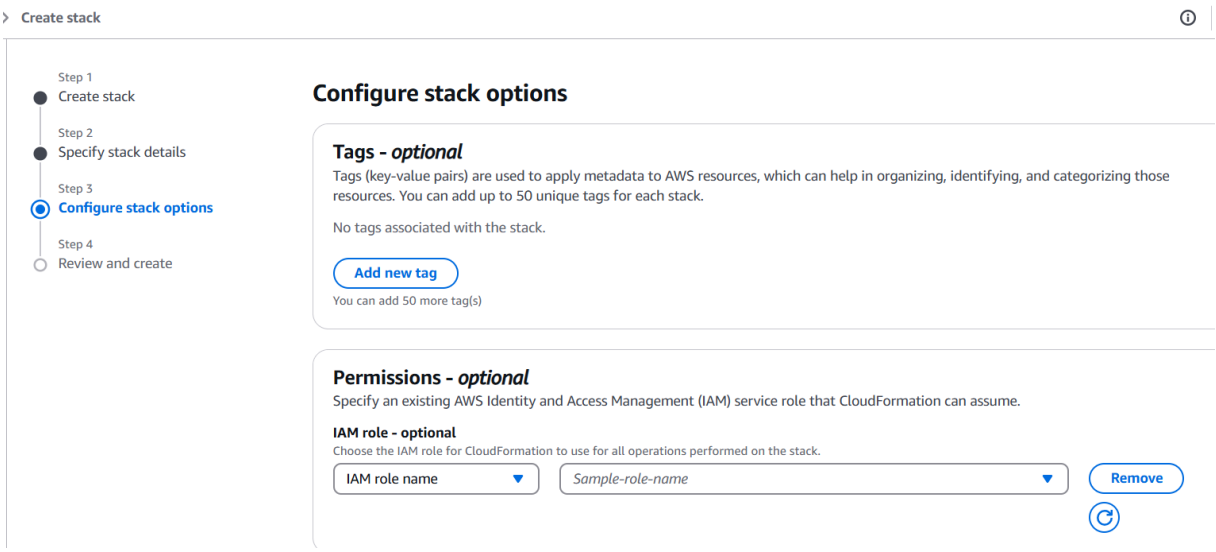
Stack Name :cloudformationinstance without spacing.

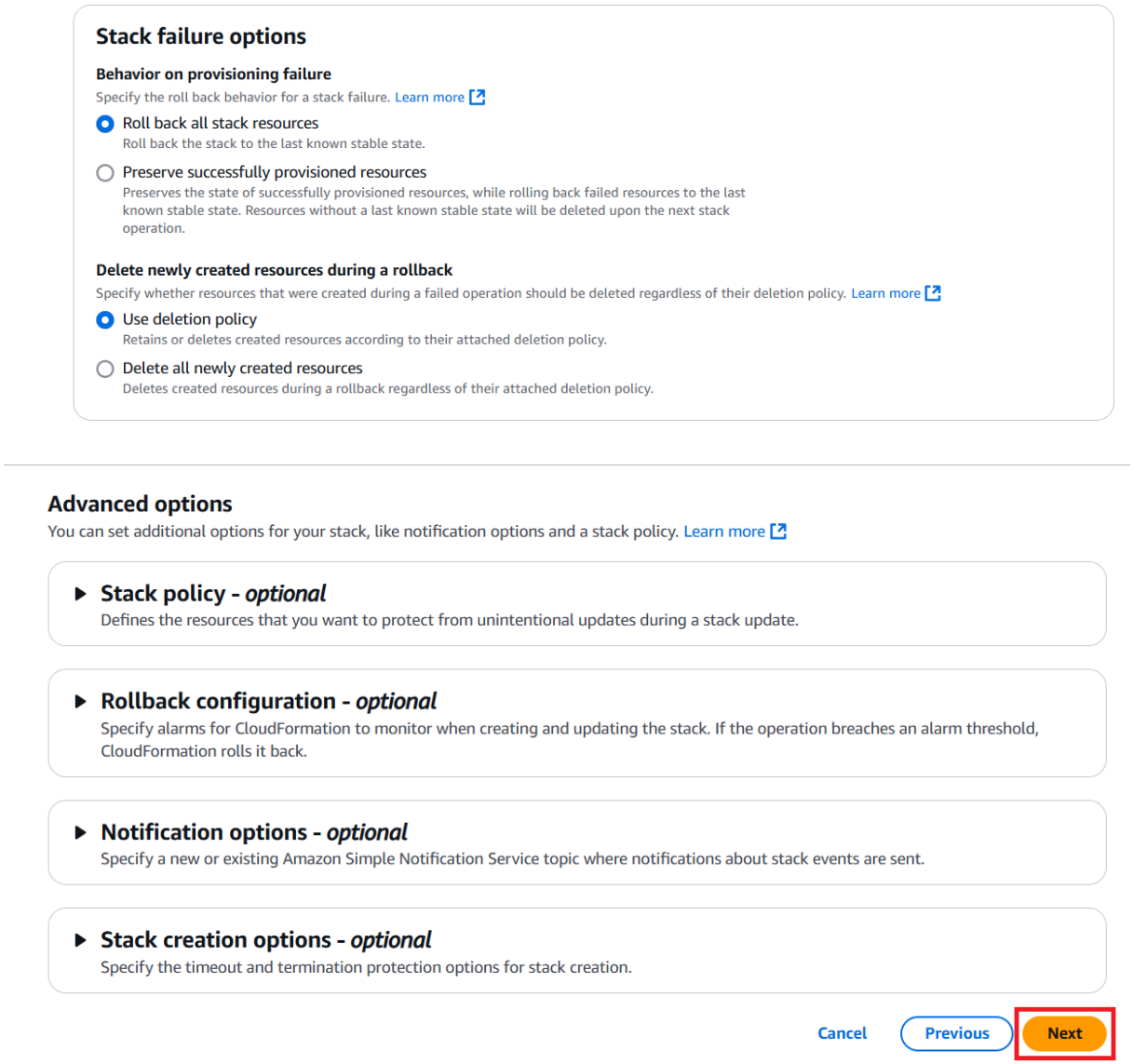
Then As per template we have set the parameters like AMI,DBInstanceClass,DBPassword,DBUsername,DomainName,EnvironmentName and HostedZone.



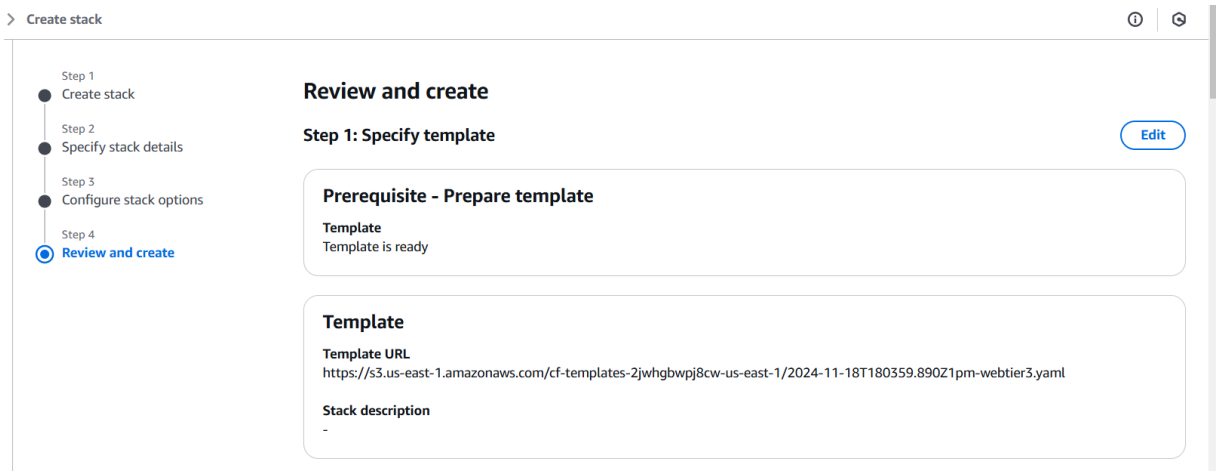


After Click Next You will see Configure Stack option we set a default setting in this.





Then After Click next.we need to review the all the setting as per templates



Step 2: Specify stack details

Edit

Provide a stack name

Stack name
cloudformationinstance

Parameters (8)

Search

1

Key	Value
AMI	ami-063d43db0594b521b
DBInstanceClass	db.t2.micro
DBName	mydatabase
DBPassword	*********
DBUsername	admin
DomainName	xyzcorporation.com
EnvironmentName	-
HostedZone	-

tep 3: Configure stack options

Edit

Tags

Search tags

1

Key	Value

No tags

There are no tags defined for this stack

Step 3: Configure stack options

Edit

Tags

Search tags

1

Key	Value

No tags

There are no tags defined for this stack

Permissions

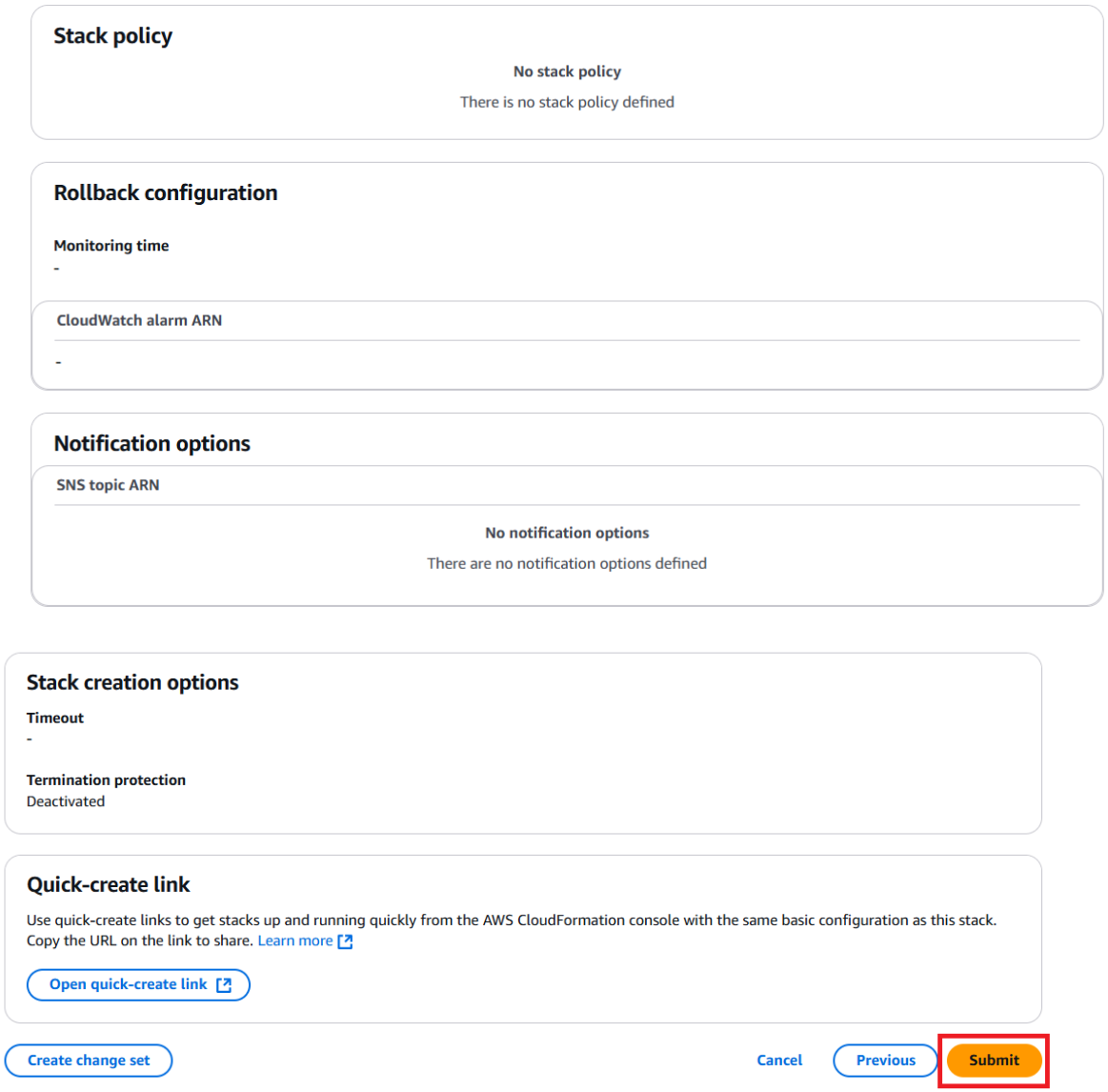
No permissions

There is no IAM role associated with this stack

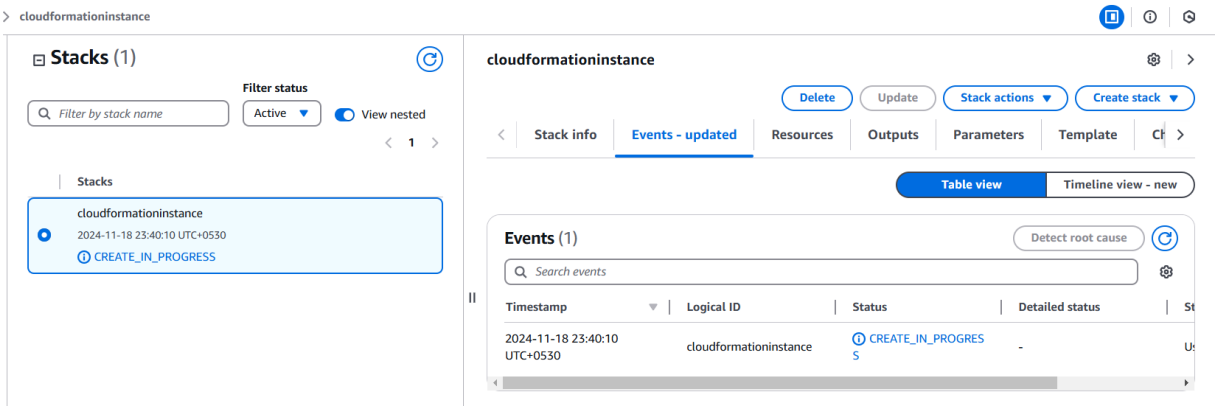
Stack failure options

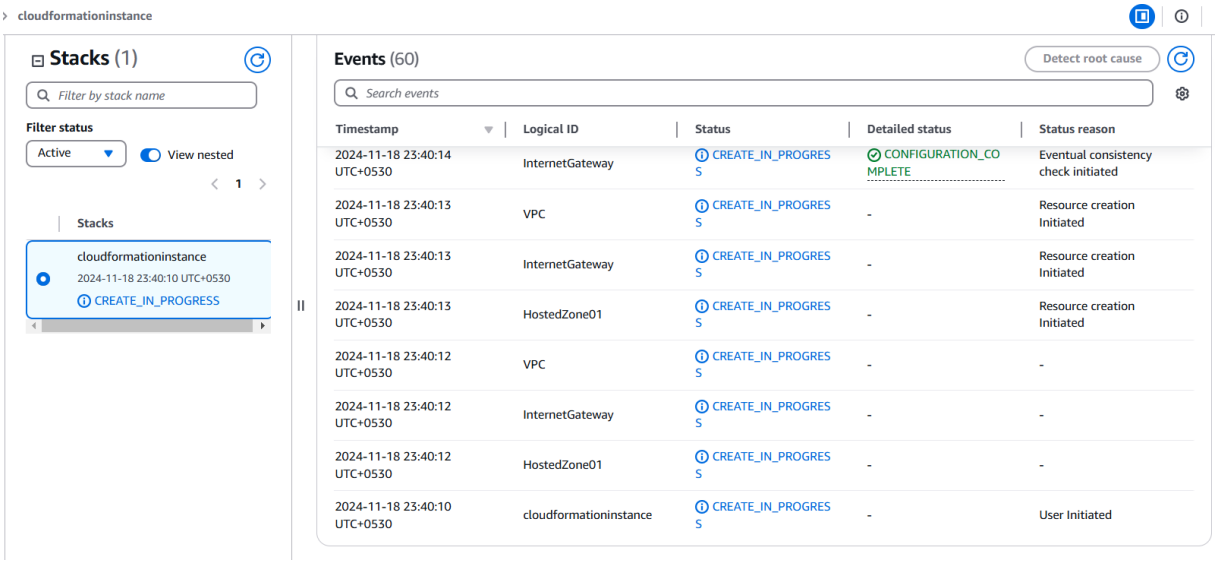
Rollback on failure
Activated

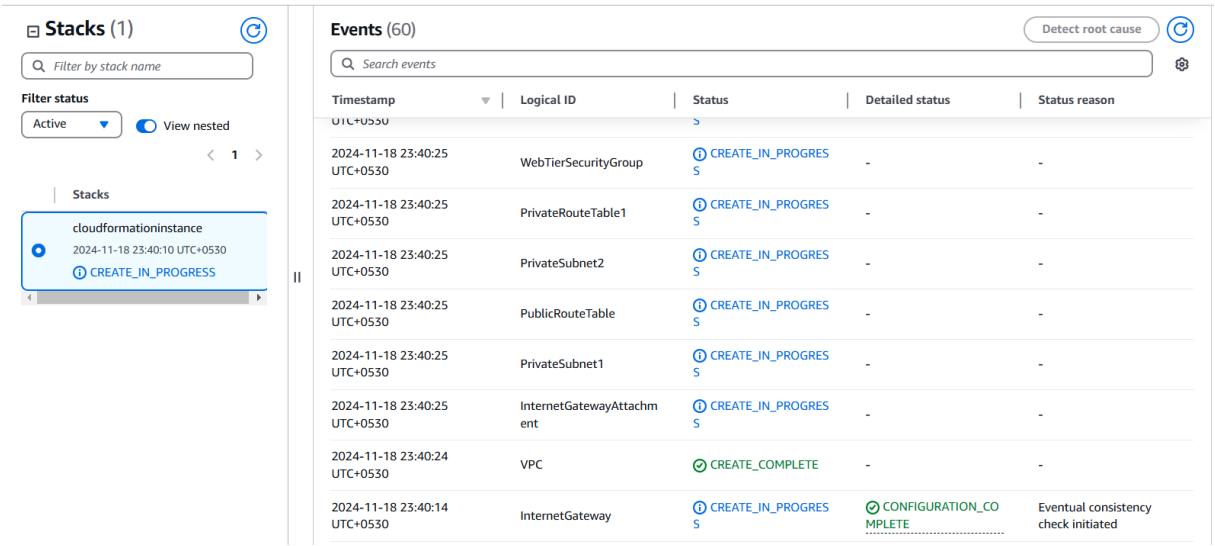
Delete newly created resources during a rollback
Deactivated

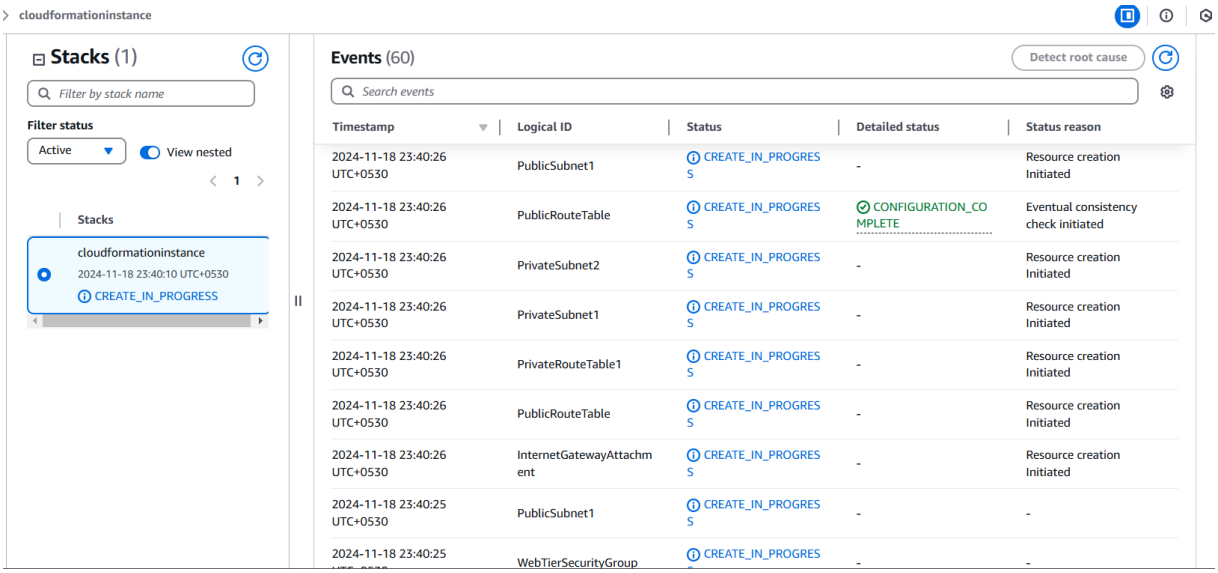


After Review completed we need to submit the code to cloud formation.cloud formation will helps you create the insfrastructure.




cloudformationinstance
Stacks (1)
Filter by stack name
Filter status
Active
View nested
1
Stacks
cloudformationinstance
2024-11-18 23:40:10 UTC+0530
CREATE_IN_PROGRESS
Events (60)
Detect root cause
Search events
Timestamp
Logical ID
Status
Detailed status
Status reason
2024-11-18 23:40:14 UTC+0530 InternetGateway CREATE_IN_PROGRESS CONFIGURATION_COMPLETE Eventual consistency check initiated
2024-11-18 23:40:13 UTC+0530 VPC CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:13 UTC+0530 InternetGateway CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:13 UTC+0530 HostedZone01 CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:12 UTC+0530 VPC CREATE_IN_PROGRESS - -
2024-11-18 23:40:12 UTC+0530 InternetGateway CREATE_IN_PROGRESS - -
2024-11-18 23:40:12 UTC+0530 HostedZone01 CREATE_IN_PROGRESS - -
2024-11-18 23:40:10 UTC+0530 cloudformationinstance CREATE_IN_PROGRESS - User Initiated


Stacks (1)
Filter by stack name
Filter status
Active
View nested
1
Stacks
cloudformationinstance
2024-11-18 23:40:10 UTC+0530
CREATE_IN_PROGRESS
Events (60)
Detect root cause
Search events
Timestamp
Logical ID
Status
Detailed status
Status reason
2024-11-18 23:40:25 UTC+0530 WebTierSecurityGroup CREATE_IN_PROGRESS - -
2024-11-18 23:40:25 UTC+0530 PrivateRouteTable1 CREATE_IN_PROGRESS - -
2024-11-18 23:40:25 UTC+0530 PrivateSubnet2 CREATE_IN_PROGRESS - -
2024-11-18 23:40:25 UTC+0530 PublicRouteTable CREATE_IN_PROGRESS - -
2024-11-18 23:40:25 UTC+0530 PrivateSubnet1 CREATE_IN_PROGRESS - -
2024-11-18 23:40:25 UTC+0530 InternetGatewayAttachment CREATE_IN_PROGRESS - -
2024-11-18 23:40:24 UTC+0530 VPC CREATE_COMPLETE - -
2024-11-18 23:40:14 UTC+0530 InternetGateway CREATE_IN_PROGRESS CONFIGURATION_COMPLETE Eventual consistency check initiated


cloudformationinstance
Stacks (1)
Filter by stack name
Filter status
Active
View nested
1
Stacks
cloudformationinstance
2024-11-18 23:40:10 UTC+0530
CREATE_IN_PROGRESS
Events (60)
Detect root cause
Search events
Timestamp
Logical ID
Status
Detailed status
Status reason
2024-11-18 23:40:26 UTC+0530 PublicSubnet1 CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:26 UTC+0530 PublicRouteTable CREATE_IN_PROGRESS CONFIGURATION_COMPLETE Eventual consistency check initiated
2024-11-18 23:40:26 UTC+0530 PrivateSubnet2 CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:26 UTC+0530 PrivateSubnet1 CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:26 UTC+0530 PrivateRouteTable1 CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:26 UTC+0530 PublicRouteTable CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:26 UTC+0530 InternetGatewayAttachment CREATE_IN_PROGRESS - Resource creation Initiated
2024-11-18 23:40:25 UTC+0530 PublicSubnet1 CREATE_IN_PROGRESS - -
2024-11-18 23:40:25 WebTierSecurityGroup CREATE_IN_PROGRES - -

2024-11-18 23:40:29 UTC+0530	PublicSubnet1	CREATE_COMPLETE	-	-
2024-11-18 23:40:29 UTC+0530	InternetGateway	CREATE_COMPLETE	-	-
2024-11-18 23:40:28 UTC+0530	PrivateSubnet1	CREATE_COMPLETE	-	-
2024-11-18 23:40:28 UTC+0530	PrivateSubnet2	CREATE_COMPLETE	-	-
2024-11-18 23:40:27 UTC+0530	InternetGatewayAttachment	CREATE_COMPLETE	-	-
2024-11-18 23:40:27 UTC+0530	WebTierSecurityGroup	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:27 UTC+0530	PrivateRouteTable1	CREATE_IN_PROGRESS	CONFIGURATION_COMPLETE	Eventual consistency check initiated
2024-11-18 23:40:26 UTC+0530	PublicSubnet1	CREATE_IN_PROGRESS	-	Resource creation Initiated

2024-11-18 23:40:34 UTC+0530	ApplicationTierSecurityGroup	CREATE_IN_PROGRESS	-	-
2024-11-18 23:40:34 UTC+0530	WebTierSecurityGroup	CREATE_COMPLETE	-	-
2024-11-18 23:40:32 UTC+0530	RDSDBSubnetGroup	CREATE_COMPLETE	-	-
2024-11-18 23:40:32 UTC+0530	RDSDBSubnetGroup	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:31 UTC+0530	PublicSubnet1RouteTableAssociation	CREATE_COMPLETE	-	-
2024-11-18 23:40:31 UTC+0530	PublicSubnet1RouteTableAssociation	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:30 UTC+0530	PublicSubnet1RouteTableAssociation	CREATE_IN_PROGRESS	-	-
2024-11-18 23:40:30 UTC+0530	RDSDBSubnetGroup	CREATE_IN_PROGRESS	-	-

cloudformationinstance

Stacks (1)

Filter by stack name

Filter status

Active | View nested

1

Stacks

cloudformationinstance
2024-11-18 23:40:10 UTC+0530
CREATE_IN_PROGRESS

Events (60)

Detect root cause

Search events

Timestamp	Logical ID	Status	Detailed status	Status reason
2024-11-18 23:40:38 UTC+0530	DefaultPublicRoute	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:38 UTC+0530	WebTierInstance	CREATE_IN_PROGRESS	CONFIGURATION_COMPLETE	Eventual consistency check initiated
2024-11-18 23:40:37 UTC+0530	PrivateRouteTable1	CREATE_COMPLETE	-	-
2024-11-18 23:40:37 UTC+0530	DefaultPublicRoute	CREATE_IN_PROGRESS	-	-
2024-11-18 23:40:36 UTC+0530	ApplicationTierSecurityGroup	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:36 UTC+0530	PublicRouteTable	CREATE_COMPLETE	-	-
2024-11-18 23:40:36 UTC+0530	WebTierInstance	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:34 UTC+0530	WebTierInstance	CREATE_IN_PROGRESS	-	-

2024-11-18 23:40:52 UTC+0530	InstanceSecurityGroupDataBase	CREATE_COMPLETE	-	-
2024-11-18 23:40:47 UTC+0530	WebTierInstance	CREATE_COMPLETE	-	-
2024-11-18 23:40:46 UTC+0530	InstanceSecurityGroupDataBase	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:45 UTC+0530	ApplicationTierInstance	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:43 UTC+0530	InstanceSecurityGroupDataBase	CREATE_IN_PROGRESS	-	-
2024-11-18 23:40:43 UTC+0530	ApplicationTierInstance	CREATE_IN_PROGRESS	-	-
2024-11-18 23:40:42 UTC+0530	ApplicationTierSecurityGroup	CREATE_COMPLETE	-	-
2024-11-18 23:40:38 UTC+0530	DefaultPublicRoute	CREATE_COMPLETE	-	-

2024-11-18 23:41:26 UTC+0530	MyDNSRecord	CREATE_COMPLETE	-	-
2024-11-18 23:40:56 UTC+0530	ApplicationTierInstance	CREATE_COMPLETE	-	-
2024-11-18 23:40:55 UTC+0530	MyDNSRecord	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:55 UTC+0530	RDSDBInstance	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-11-18 23:40:55 UTC+0530	MyDNSRecord	CREATE_IN_PROGRESS	-	-
2024-11-18 23:40:54 UTC+0530	HostedZone01	CREATE_COMPLETE	-	-
2024-11-18 23:40:52 UTC+0530	RDSDBInstance	CREATE_IN_PROGRESS	-	-

Resources (19)

Search resources

1

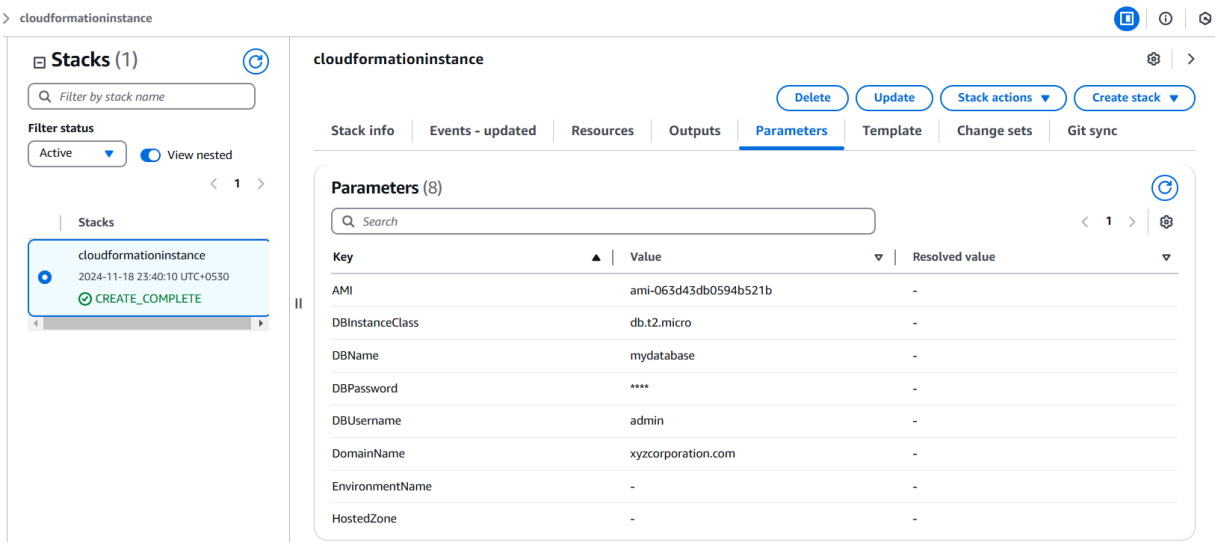
Logical ID	Physical ID	Type	Status	Module
PublicRouteTable	rtb-0c8d4b4d93abf4cea	AWS::EC2::RouteTable	CREATE_COMPLETE	-
PublicSubnet1	subnet-055241059f4e1df63	AWS::EC2::Subnet	CREATE_COMPLETE	-
PublicSubnet1RouteTableAssociation	rtbassoc-001f9823c4b650db5	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE	-
RDSDBInstance	mydatabase	AWS::RDS::DBInstance	CREATE_IN_PROGRESS	-
RDSDBSubnetGroup	cloudformationinstance-rdsdbsubnetgroup-6lexkpw9rorp	AWS::RDS::DBSubnetGroup	CREATE_COMPLETE	-
VPC	vpc-03bffb17009438a4a	AWS::EC2::VPC	CREATE_COMPLETE	-
WebTierInstance	i-0aa4654286d23cfd7	AWS::EC2::Instance	CREATE_COMPLETE	-
WebTierSecurityGroup	sg-0c42205f6ee8d8590	AWS::EC2::SecurityGroup	CREATE_COMPLETE	-

Resources (19)

Search resources

Logical ID	Physical ID	Type	Status	Module
DefaultPublicRoute	0c8d4b4d93abf4cea|0.0.0.0/0	AWS::EC2::Route	CREATE_COMPLETE	-
HostedZone01	Z09315002SDBCJQRM3B5R	AWS::Route53::HostedZone	CREATE_COMPLETE	-
InstanceSecurityGroupDataBase	sg-027729a683b465233	AWS::EC2::SecurityGroup	CREATE_COMPLETE	-
InternetGateway	igw-095e118dc3b17a132	AWS::EC2::InternetGateway	CREATE_COMPLETE	-
InternetGatewayAttachment	IGW|vpc-03bffb17009438a4a	AWS::EC2::VPCGatewayAttachment	CREATE_COMPLETE	-
MyDNSRecord	webtier3.xyzcorporation.com	AWS::Route53::RecordSet	CREATE_COMPLETE	-
PrivateRouteTable1	rtb-0a1bcd50e619b5133	AWS::EC2::RouteTable	CREATE_COMPLETE	-
PrivateSubnet1	subnet-098abfb3b316a9a14	AWS::EC2::Subnet	CREATE_COMPLETE	-
PrivateSubnet2	subnet-03e60bc4188364da7	AWS::EC2::Subnet	CREATE_COMPLETE	-
ApplicationTierInstance	i-0223bf0fd951e94fa	AWS::EC2::Instance	CREATE_COMPLETE	-
ApplicationTierSecurityGroup	sg-0f1302a16548373a4	AWS::EC2::SecurityGroup	CREATE_COMPLETE	-

This Is the Parameter section as we set before the templates

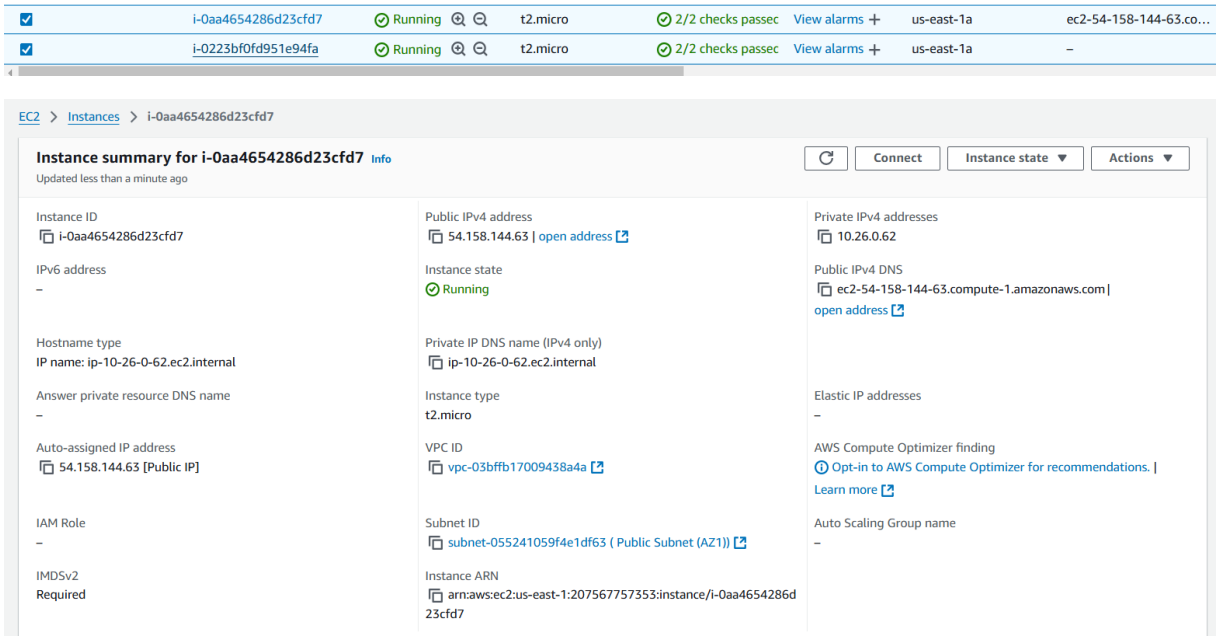


After Complete this process we need to all things verify.

This is the instance created

1.public instance

2.private instance



☑	i-0aa4654286d23cfd7	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	ec2-54-158-144-63.co…
☑	i-0223bf0fd951e94fa	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	–

EC2 > Instances > i-0aa4654286d23cfd7

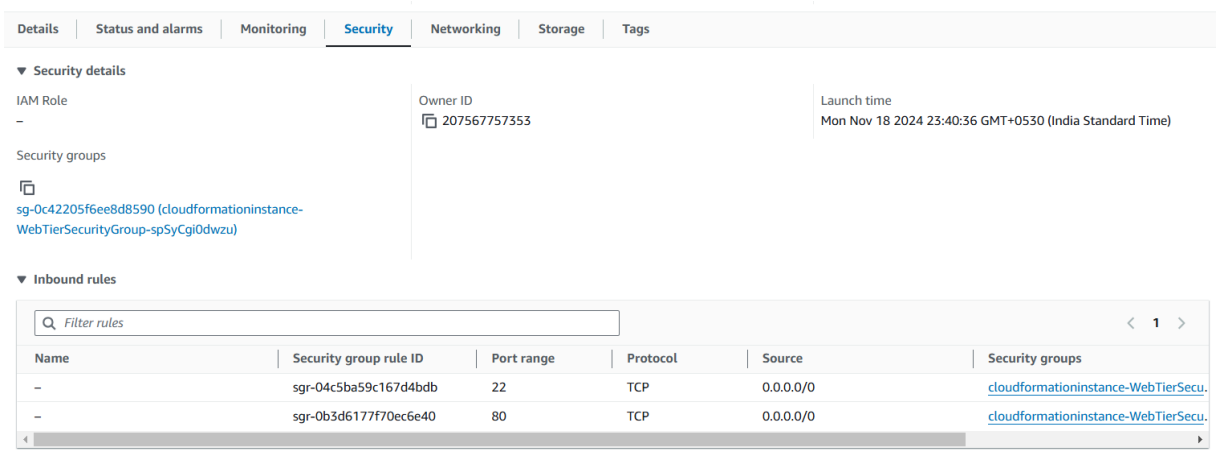
Instance summary for i-0aa4654286d23cfd7 Info

Updated less than a minute ago

Connect | Instance state | Actions

Instance ID i-0aa4654286d23cfd7	Public IPv4 address 54.158.144.63 | open address	Private IPv4 addresses 10.26.0.62
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-54-158-144-63.compute-1.amazonaws.com | open address
Hostname type IP name: ip-10-26-0-62.ec2.internal	Private IP DNS name (IPv4 only) ip-10-26-0-62.ec2.internal	
Answer private resource DNS name –	Instance type t2.micro	Elastic IP addresses –
Auto-assigned IP address 54.158.144.63 [Public IP]	VPC ID vpc-03bffb17009438a4a	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-055241059f4e1df63 (Public Subnet (AZ1))	Auto Scaling Group name –
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:207567757353:instance/i-0aa4654286d23cfd7	

This is the security group as per templates



Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

▼ Security details

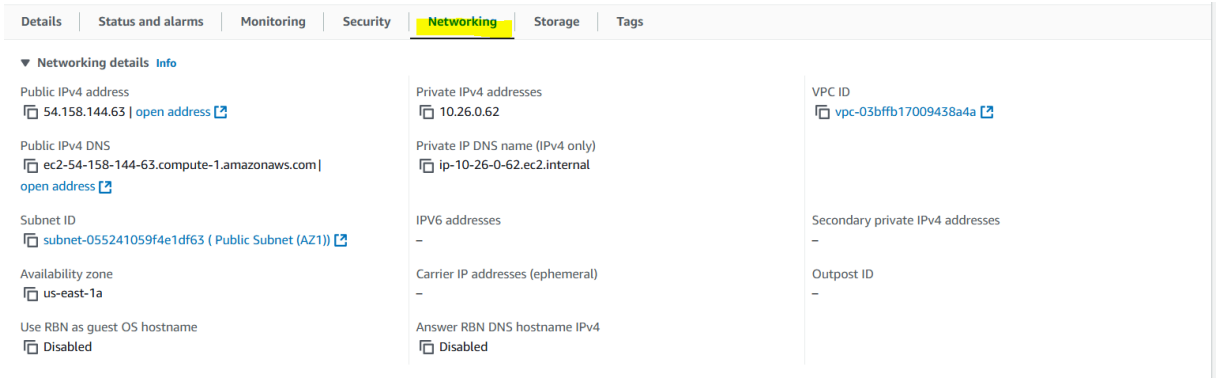
IAM Role –	Owner ID 207567757353	Launch time Mon Nov 18 2024 23:40:36 GMT+0530 (India Standard Time)
Security groups sg-0c42205f6ee8d8590 (cloudformationinstance-WebTierSecurityGroup-spSyCgi0dwzu)		

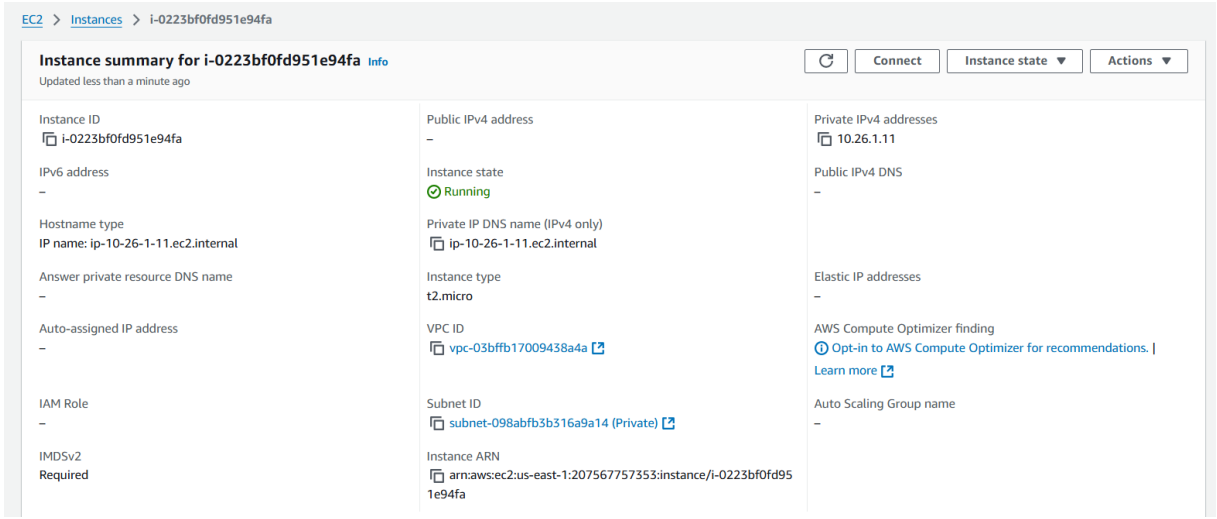
▼ Inbound rules

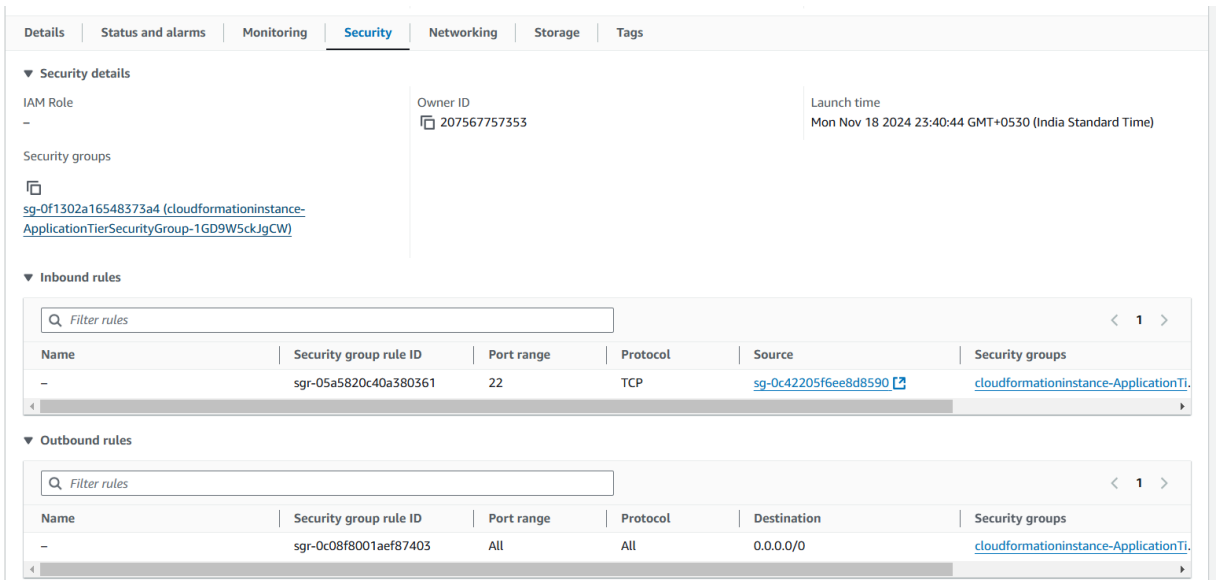
Filter rules

1

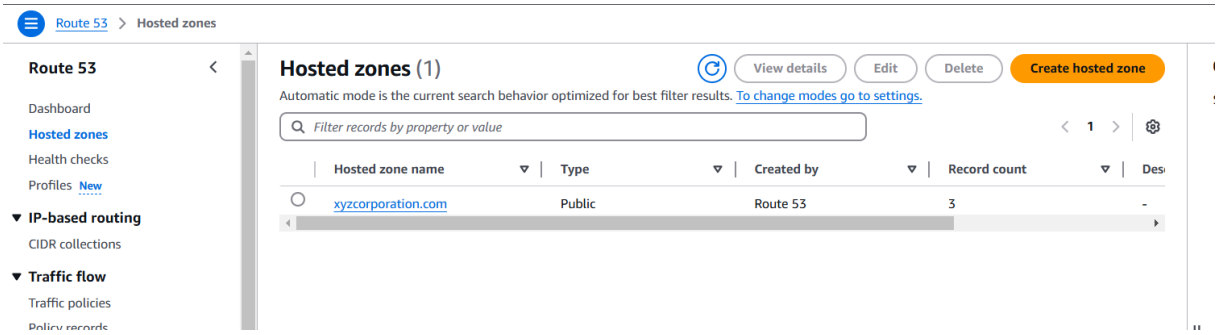
Name	Security group rule ID	Port range	Protocol	Source	Security groups
–	sgr-04c5ba59c167d4bdb	22	TCP	0.0.0.0/0	cloudformationinstance-WebTierSecu…
–	sgr-0b3d6177f70ec6e40	80	TCP	0.0.0.0/0	cloudformationinstance-WebTierSecu…


Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Networking details Info
Public IPv4 address
54.158.144.63 | open address
Private IPv4 addresses
10.26.0.62
VPC ID
vpc-03bffb17009438a4a
Public IPv4 DNS
ec2-54-158-144-63.compute-1.amazonaws.com | open address
Private IP DNS name (IPv4 only)
ip-10-26-0-62.ec2.internal
Subnet ID
subnet-055241059f4e1df63 (Public Subnet (AZ1))
IPV6 addresses
–
Secondary private IPv4 addresses
–
Availability zone
us-east-1a
Carrier IP addresses (ephemeral)
–
Outpost ID
–
Use RBN as guest OS hostname
Disabled
Answer RBN DNS hostname IPv4
Disabled

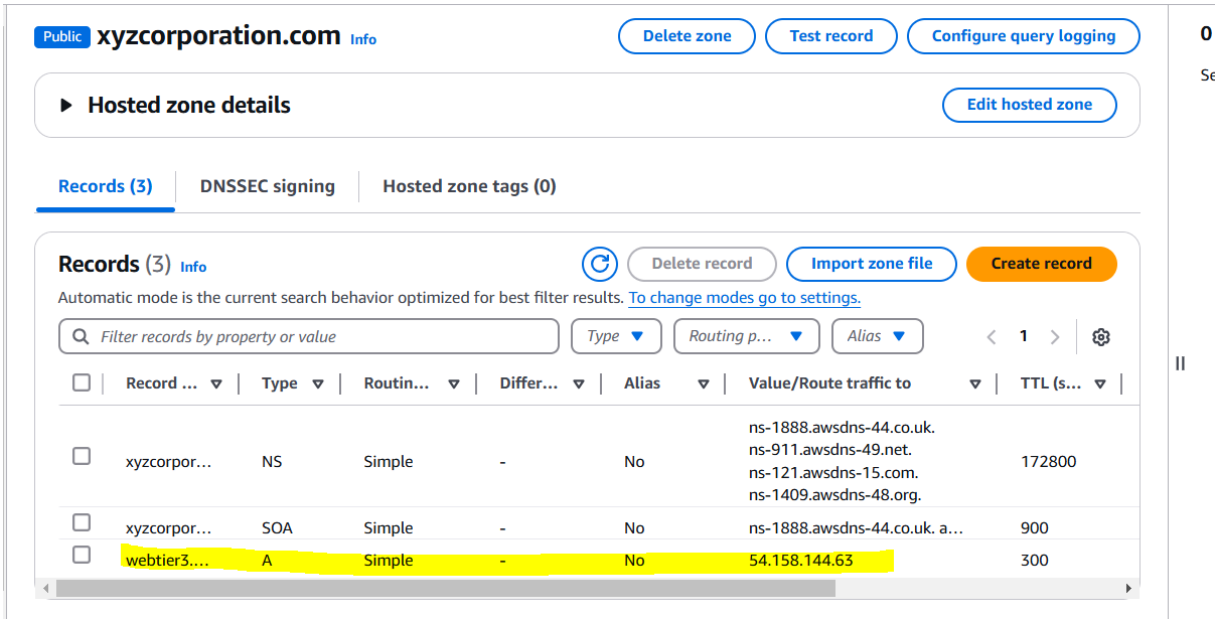

EC2
Instances
i-0223bf0fd951e94fa
Instance summary for i-0223bf0fd951e94fa Info
Updated less than a minute ago
Connect
Instance state
Actions
Instance ID
i-0223bf0fd951e94fa
Public IPv4 address
–
Private IPv4 addresses
10.26.1.11
IPv6 address
–
Instance state
Running
Public IPv4 DNS
–
Hostname type
IP name: ip-10-26-1-11.ec2.internal
Private IP DNS name (IPv4 only)
ip-10-26-1-11.ec2.internal
Answer private resource DNS name
–
Instance type
t2.micro
Elastic IP addresses
–
Auto-assigned IP address
–
VPC ID
vpc-03bffb17009438a4a
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role
–
Subnet ID
subnet-098abfb3b316a9a14 (Private)
Auto Scaling Group name
–
IMDSv2
Required
Instance ARN
arn:aws:ec2:us-east-1:207567757353:instance/i-0223bf0fd951e94fa


Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Security details
IAM Role
–
Owner ID
207567757353
Launch time
Mon Nov 18 2024 23:40:44 GMT+0530 (India Standard Time)
Security groups
sg-0f1302a16548373a4 (cloudformationinstance-ApplicationTierSecurityGroup-1GD9W5ckJgCW)
Inbound rules
Filter rules
1
Name
Security group rule ID
Port range
Protocol
Source
Security groups
–
sgr-05a5820c40a380361
22
TCP
sg-0c42205f6ee8d8590
cloudformationinstance-ApplicationTi.
Outbound rules
Filter rules
1
Name
Security group rule ID
Port range
Protocol
Destination
Security groups
–
sgr-0c08f8001aef87403
All
All
0.0.0.0/0
cloudformationinstance-ApplicationTi.

This is the Hosted Zone



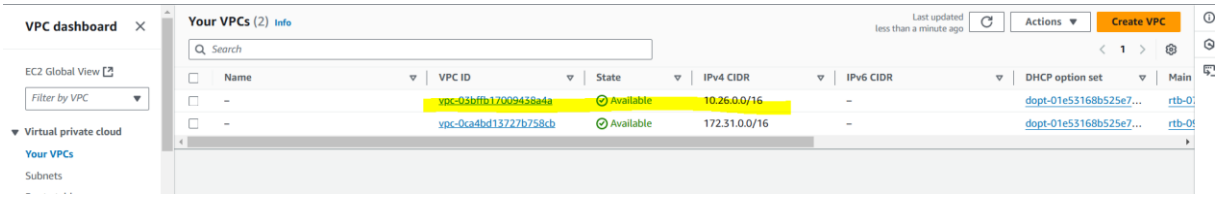
In this Hosted zone we have created A record as per code.

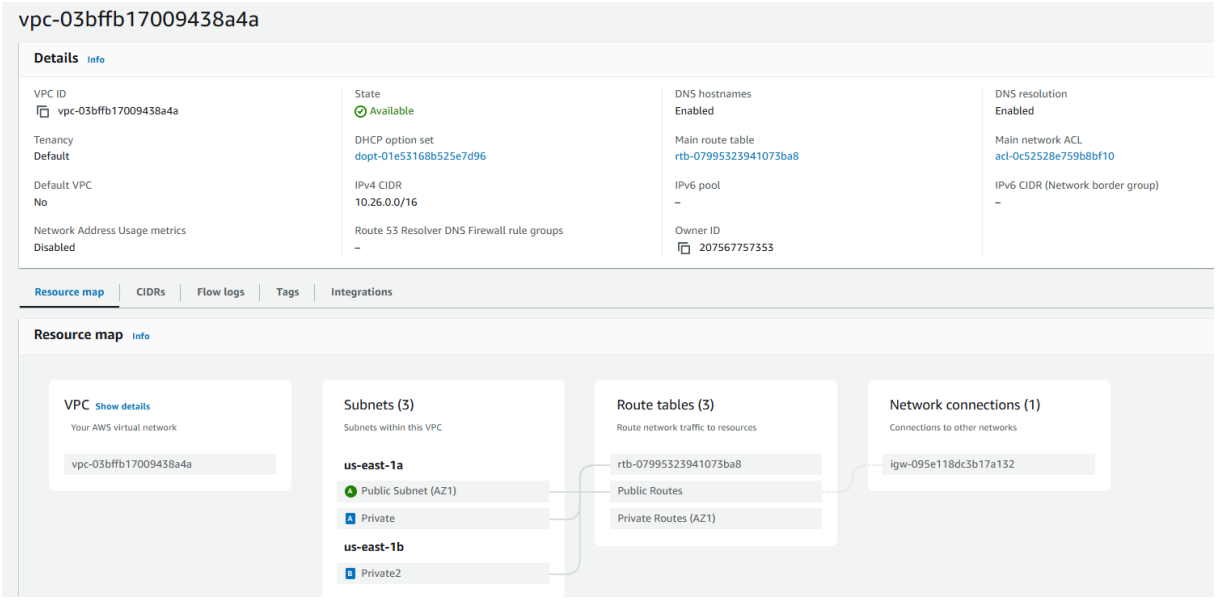


This is the VPC

1.Public

2.Private



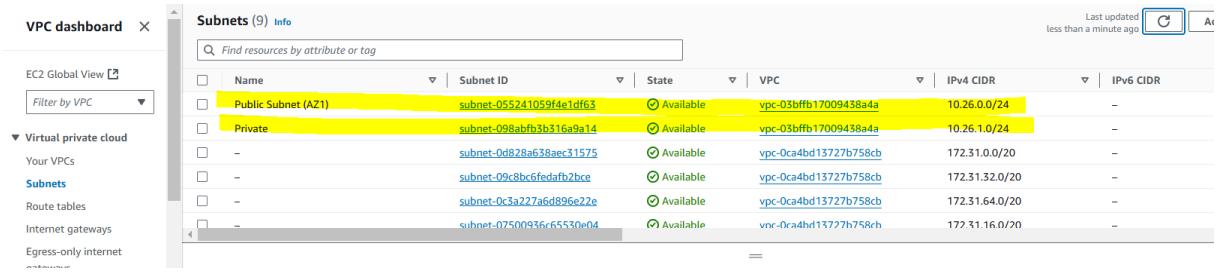


This is the subnet creation

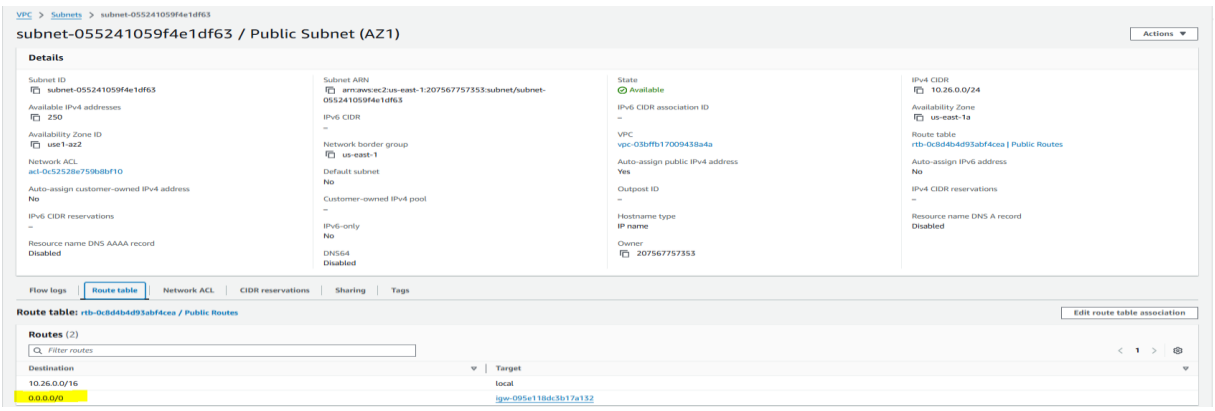
1.Public

2.Private

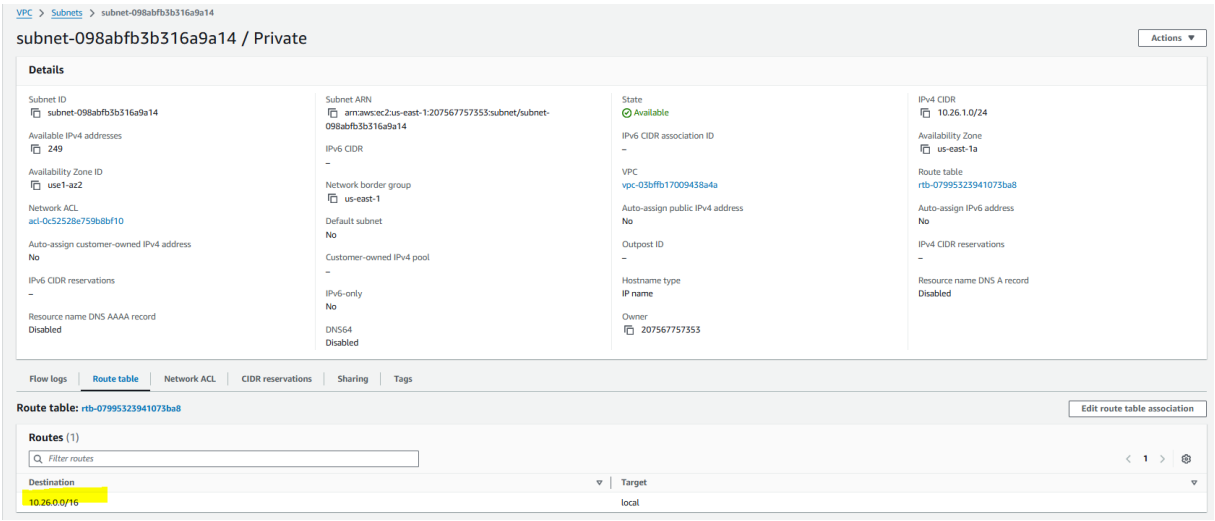
As per the task.



This is public subnet in internet.



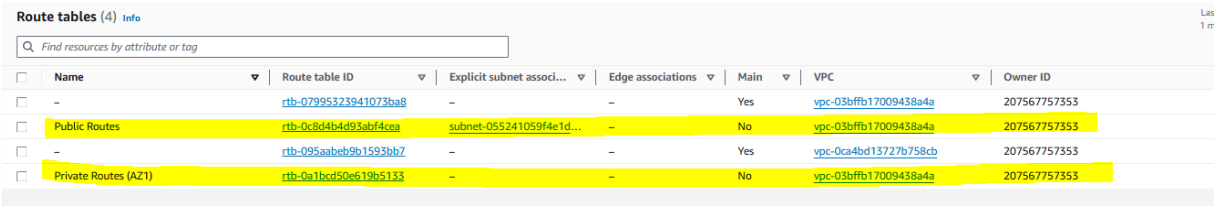
This is the private subnet without internet.

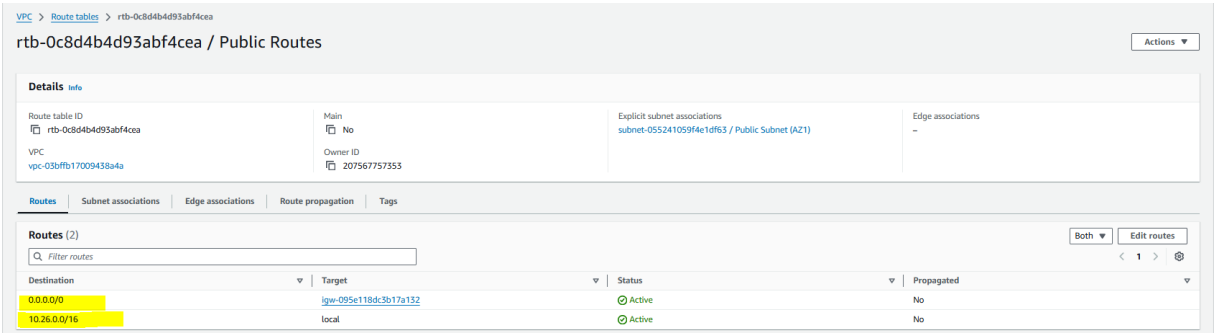


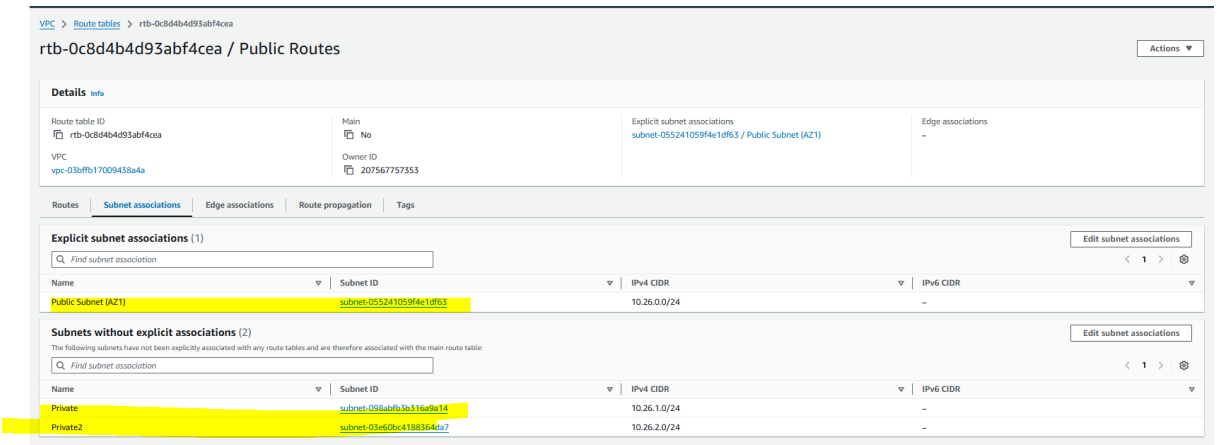
Route table

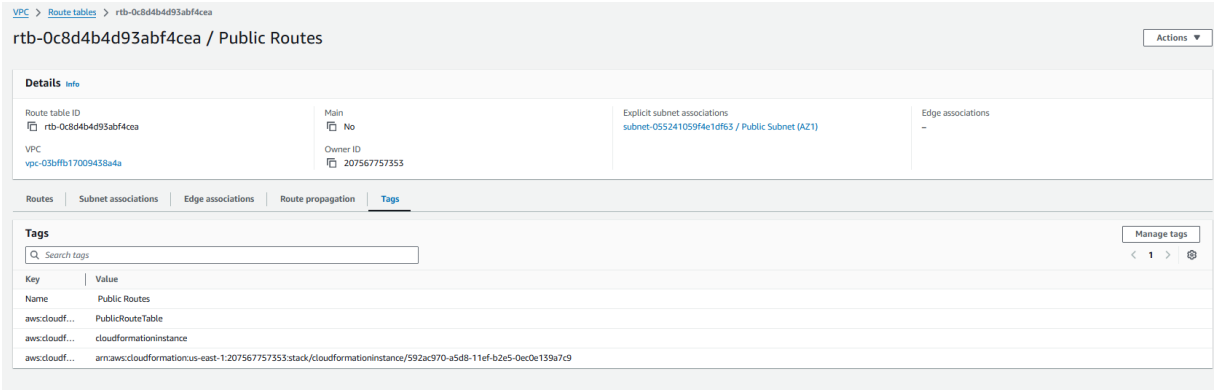
1.Public

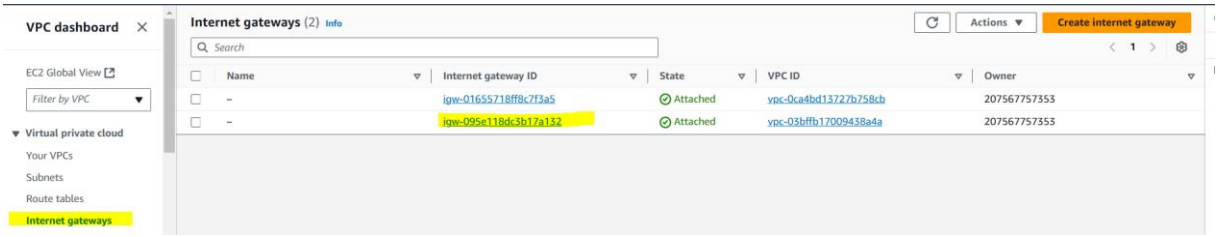
2.Private.

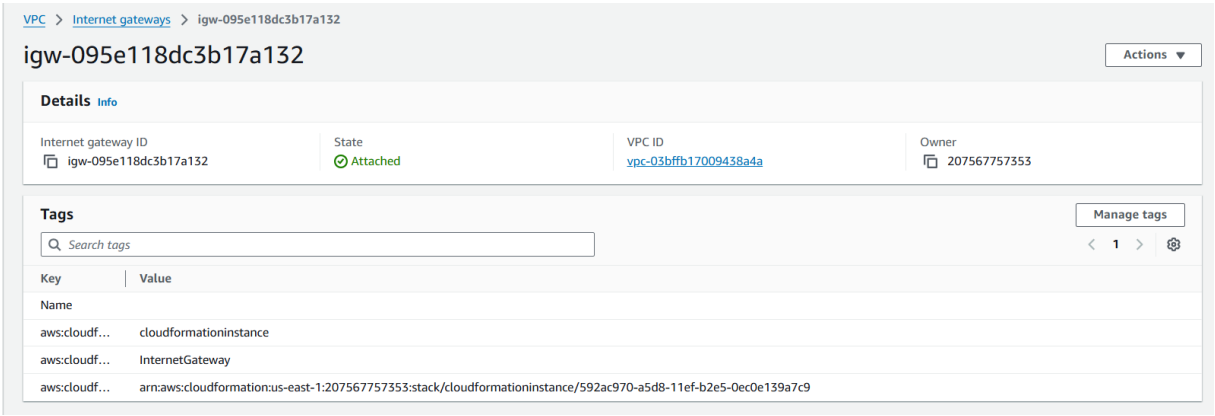


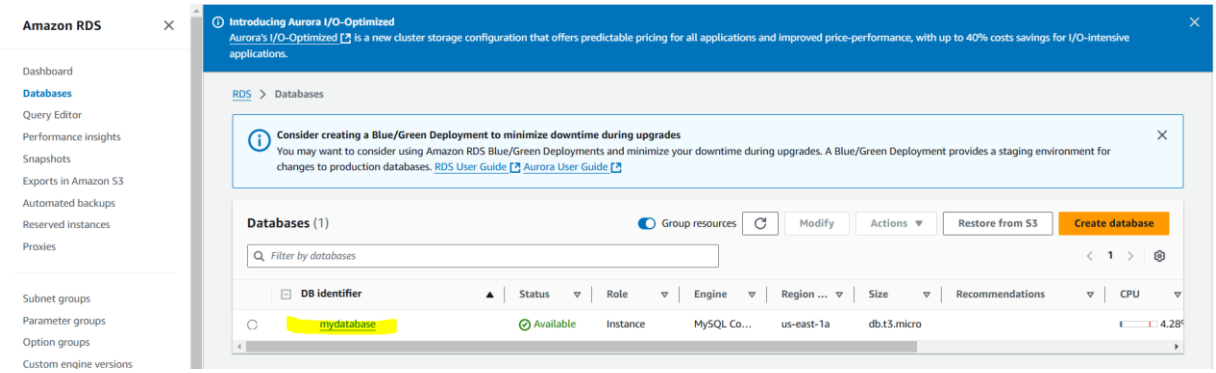





VPC > Route tables > rtb-0c8d4b4d93abf4cea
rtb-0c8d4b4d93abf4cea / Public Routes
Actions
Details Info
Route table ID
rtb-0c8d4b4d93abf4cea
Main
No
Explicit subnet associations
subnet-055241059f4e1df63 / Public Subnet (AZ1)
Edge associations
–
VPC
vpc-03bffb17009438a4a
Owner ID
207567757353
Routes
Subnet associations
Edge associations
Route propagation
Tags
Tags
Search tags
Manage tags
Key
Value
Name
Public Routes
aws:cloudf…
PublicRouteTable
aws:cloudf…
cloudformationinstance
aws:cloudf…
arn:aws:cloudformation:us-east-1:207567757353:stack/cloudformationinstance/592ac970-a5d8-11ef-b2e5-0ec0e139a7c9


VPC dashboard
EC2 Global View
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Internet gateways (2) Info
Search
Actions
Create internet gateway
Name
Internet gateway ID
State
VPC ID
Owner
igw-01655718ff8c7f3a5
Attached
vpc-0ca4bd13727b758cb
207567757353
igw-095e118dc3b17a132
Attached
vpc-03bffb17009438a4a
207567757353


VPC > Internet gateways > igw-095e118dc3b17a132
igw-095e118dc3b17a132
Actions
Details Info
Internet gateway ID
igw-095e118dc3b17a132
State
Attached
VPC ID
vpc-03bffb17009438a4a
Owner
207567757353
Tags
Search tags
Manage tags
Key
Value
Name
aws:cloudf…
cloudformationinstance
aws:cloudf…
InternetGateway
aws:cloudf…
arn:aws:cloudformation:us-east-1:207567757353:stack/cloudformationinstance/592ac970-a5d8-11ef-b2e5-0ec0e139a7c9


Amazon 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
Introducing Aurora I/O-Optimized
Aurora's I/O-Optimized is a new cluster storage configuration that offers predictable pricing for all applications and improved price-performance, with up to 40% costs savings for I/O-intensive applications.
RDS > Databases
Consider creating a Blue/Green Deployment to minimize downtime during upgrades
You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. RDS User Guide Aurora User Guide
Databases (1)
Group resources
Modify
Actions
Restore from S3
Create database
Filter by databases
DB identifier
Status
Role
Engine
Region …
Size
Recommendations
CPU
mydatabase
Available
Instance
MySQL Co…
us-east-1a
db.t3.micro

Summary

DB identifier	Status	Role	Engine	Recommendations
mydatabase	Available	Instance	MySQL Community	
CPU	**Class**	**Current activity**	**Region & AZ**	
4.28%	db.t3.micro	0 Connections	us-east-1a	

Connectivity & security | Monitoring | Logs & events | Configuration | Zero-ETL integrations | Maintenance & backups | Tags | Recommendations

Connectivity & security

Endpoint & port

Endpoint
mydatabase.cjo22w4eqmmj.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1a

VPC
(vpc-03bffb17009438a4a)

Subnet group
cloudformationinstance-rdsdbsubnetgroup-6lexkpw9rorp

Subnets
subnet-098abfb3b316a9a14
subnet-03e60bc4188364da7

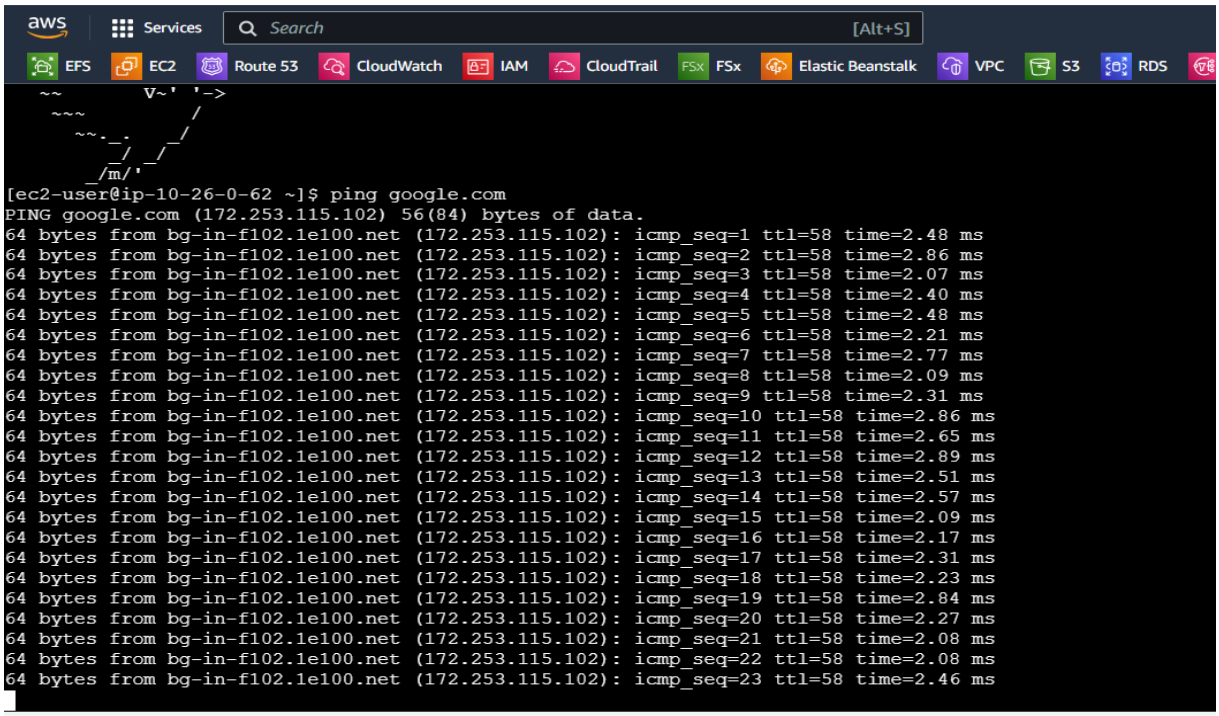
Security

VPC security groups
cloudformationinstance-InstanceSecurityGroupDataBase-oOTmlZbKprPD (sg-027729a683b465233)
Active

Publicly accessible
No

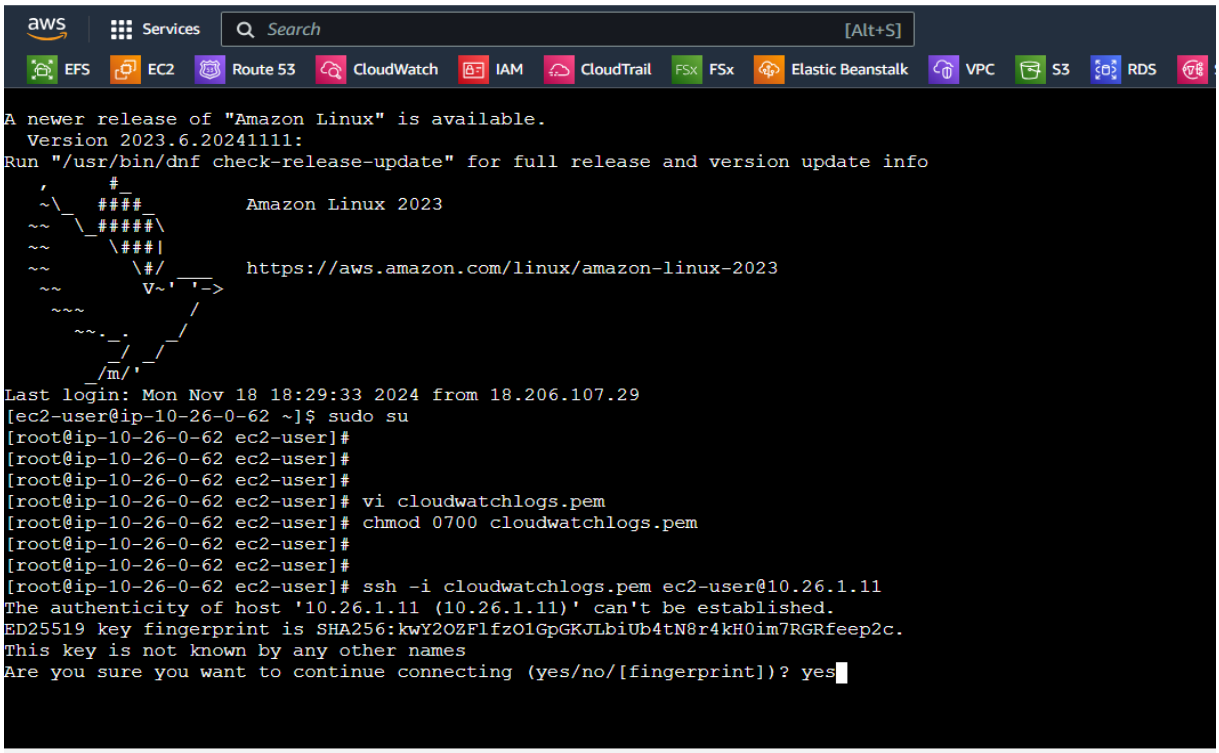
Certificate authority Info
rds-ca-rsa2048-g1

Certificate authority date



i-0aa4654286d23cfd7

PublicIPs: 54.158.144.63 PrivateIPs: 10.26.0.62



i-0aa4654286d23cfd7

PublicIPs: 54.158.144.63 PrivateIPs: 10.26.0.62