3-Tier Web Application Deployment using AWS CloudFormation Assignment

Name: ***Vikram***

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 has not been 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 that allows **HTTP (port 80)** and **SSH (port 22)** access from the internet.
2. **Application Tier:**
   Launch an instance in a private subnet of the web tier and allow only **SSH access** from the public subnet of the web tier.
3. **Database Tier:**
   Launch an **RDS MySQL instance** in a private subnet that allows connections on **port 3306** only from the private subnet of the application tier.
4. **Route 53 Setup:**
   Create a hosted zone in Route 53 and direct traffic to the web server instance.

Solution Requirement

1. The development team should be able to test their code **without depending on system admins**, focusing on testing rather than provisioning and configuring infrastructure.
2. When the development team deletes the CloudFormation stack, the **RDS instance must not be deleted** — it should be retained.

Implementation Steps

Step 1: Open CloudFormation Console

- Navigate to **AWS Management Console → CloudFormation → Create Stack → With new resources (standard)**
- Select **Template is ready → Upload a template file**
- Upload the provided YAML file.

```
AWSTemplateFormatVersion: '2010-09-09'
Description: >
  XYZ Corporation - 3 Tier Architecture (Web, App, DB) with Route53 DNS
  Web tier in public subnet, App tier in private subnet, DB tier (RDS MySQL) in private subnet.
  RDS instance retained after stack deletion.

Parameters:
  DomainName:
    Type: String
    Description: "Domain name for hosted zone (e.g., example.com)"
  KeyName:
    Type: AWS::EC2::KeyPair::KeyName
    Description: "Existing EC2 key pair for SSH login"
  DBUsername:
    Type: String
    Default: admin
    Description: "Database master username"
  DBPassword:
    Type: String
    NoEcho: true
    MinLength: 8
    Description: "Database master password"

Resources:

  # ------------------ VPC & NETWORKING ------------------
  VPC:
    Type: AWS::EC2::VPC
```

```
  Properties:
    CidrBlock: 10.0.0.0/16
    EnableDnsSupport: true
    EnableDnsHostnames: true
    Tags:
      - Key: Name
        Value: xyz-vpc

InternetGateway:
  Type: AWS::EC2::InternetGateway
  Properties:
    Tags:
      - Key: Name
        Value: xyz-igw

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

PublicSubnet:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref VPC
    AvailabilityZone: us-east-1a
    CidrBlock: 10.0.1.0/24
    MapPublicIpOnLaunch: true
    Tags:
      - Key: Name
        Value: xyz-public-subnet

AppPrivateSubnet:
  Type: AWS::EC2::Subnet
  Properties:
```

```
    VpcId: !Ref VPC
    AvailabilityZone: us-east-1a
    CidrBlock: 10.0.2.0/24
    Tags:
      - Key: Name
        Value: xyz-app-private-subnet

  DBPrivateSubnet:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref VPC
      AvailabilityZone: us-east-1b
      CidrBlock: 10.0.3.0/24
      Tags:
        - Key: Name
          Value: xyz-db-private-subnet

  PublicRouteTable:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: !Ref VPC
      Tags:
        - Key: Name
          Value: xyz-public-rt

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

  PublicSubnetAssociation:
    Type: AWS::EC2::SubnetRouteTableAssociation
```

```
  Properties:
    SubnetId: !Ref PublicSubnet
    RouteTableId: !Ref PublicRouteTable

# ------------------ SECURITY GROUPS ------------------
WebSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow HTTP and SSH from internet
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
    Tags:
      - Key: Name
        Value: xyz-web-sg

AppSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow SSH only from Web Tier
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        SourceSecurityGroupId: !Ref WebSecurityGroup
    Tags:
      - Key: Name
```

```
      Value: xyz-app-sg

  DBSecurityGroup:
    Type: AWS::EC2::SecurityGroup
    Properties:
      GroupDescription: Allow MySQL only from App Tier
      VpcId: !Ref VPC
      SecurityGroupIngress:
        - IpProtocol: tcp
          FromPort: 3306
          ToPort: 3306
          SourceSecurityGroupId: !Ref AppSecurityGroup
      Tags:
        - Key: Name
          Value: xyz-db-sg

  # ------------------ EC2 INSTANCES ------------------
  WebInstance:
    Type: AWS::EC2::Instance
    Properties:
      InstanceType: t3.micro
      KeyName: !Ref KeyName
      ImageId: ami-07860a2d7eb515d9a   # ✅ Updated AMI ID
      SubnetId: !Ref PublicSubnet
      SecurityGroupIds:
        - !Ref WebSecurityGroup
      Tags:
        - Key: Name
          Value: xyz-web-instance
      UserData:
        Fn::Base64: !Sub |
          #!/bin/bash
          yum update -y
          yum install -y httpd
          systemctl enable httpd
```

```
        systemctl start httpd
        echo "<h1>Welcome to XYZ Web Server (Web Tier)</h1>" > /var/www/html/index.html

  AppInstance:
    Type: AWS::EC2::Instance
    Properties:
      InstanceType: t3.micro
      KeyName: !Ref KeyName
      ImageId: ami-07860a2d7eb515d9a   # ✅ Updated AMI ID
      SubnetId: !Ref AppPrivateSubnet
      SecurityGroupIds:
        - !Ref AppSecurityGroup
      Tags:
        - Key: Name
          Value: xyz-app-instance

  # ------------------ RDS DATABASE ------------------
  DBSubnetGroup:
    Type: AWS::RDS::DBSubnetGroup
    Properties:
      DBSubnetGroupDescription: Subnet group for xyz DB
      SubnetIds:
        - !Ref AppPrivateSubnet
        - !Ref DBPrivateSubnet
      Tags:
        - Key: Name
          Value: xyz-db-subnet-group

  MySQLDB:
    Type: AWS::RDS::DBInstance
    DeletionPolicy: Retain
    Properties:
      DBInstanceIdentifier: xyz-mysql-db
      AllocatedStorage: 20
      DBInstanceClass: db.t3.micro
```

```
      Engine: mysql
      MasterUsername: !Ref DBUsername
      MasterUserPassword: !Ref DBPassword
      DBSubnetGroupName: !Ref DBSubnetGroup
      VPCSecurityGroups:
        - !Ref DBSecurityGroup
      PubliclyAccessible: false
      MultiAZ: false

  # ------------------ ROUTE 53 ------------------
  HostedZone:
    Type: AWS::Route53::HostedZone
    Properties:
      Name: !Ref DomainName
      HostedZoneConfig:
        Comment: "XYZ Hosted Zone"

  DNSRecord:
    Type: AWS::Route53::RecordSet
    Properties:
      HostedZoneId: !Ref HostedZone
      Name: !Sub "${DomainName}."
      Type: A
      TTL: '300'
      ResourceRecords:
        - !GetAtt WebInstance.PublicIp

Outputs:
  WebInstancePublicIP:
    Description: "Public IP of Web Instance"
    Value: !GetAtt WebInstance.PublicIp
  AppInstanceID:
    Description: "Application Instance ID"
    Value: !Ref AppInstance
  DBEndpoint:
```

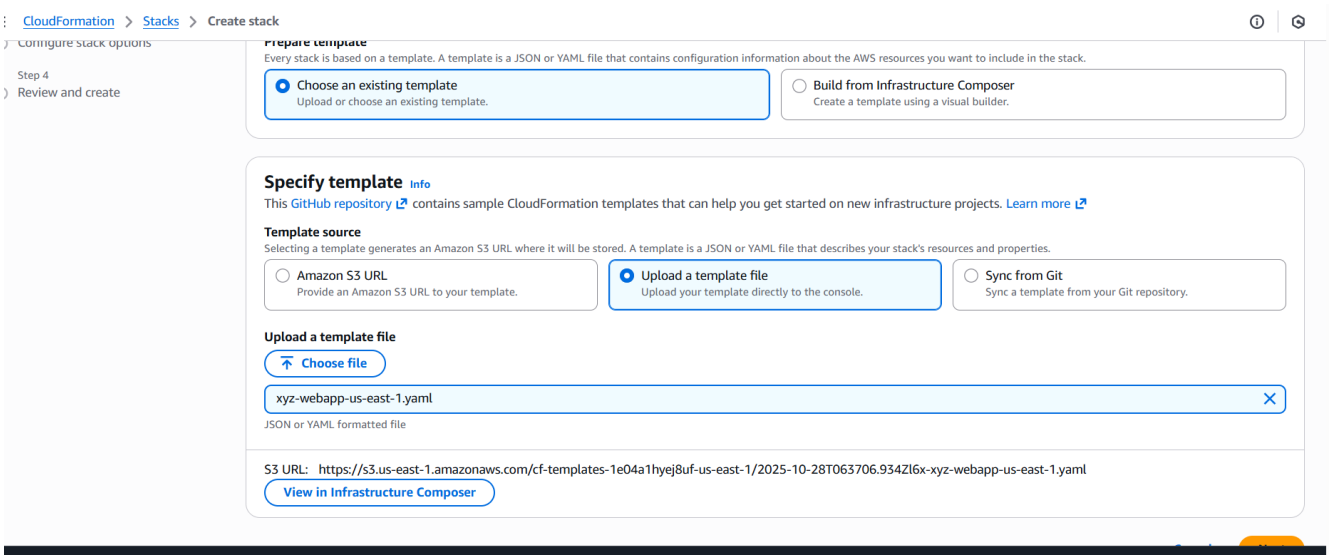
Description: "RDS MySQL Endpoint"

Value: !GetAtt MySQLDB.Endpoint.Address

HostedZoneID:

Description: "Route53 Hosted Zone ID"

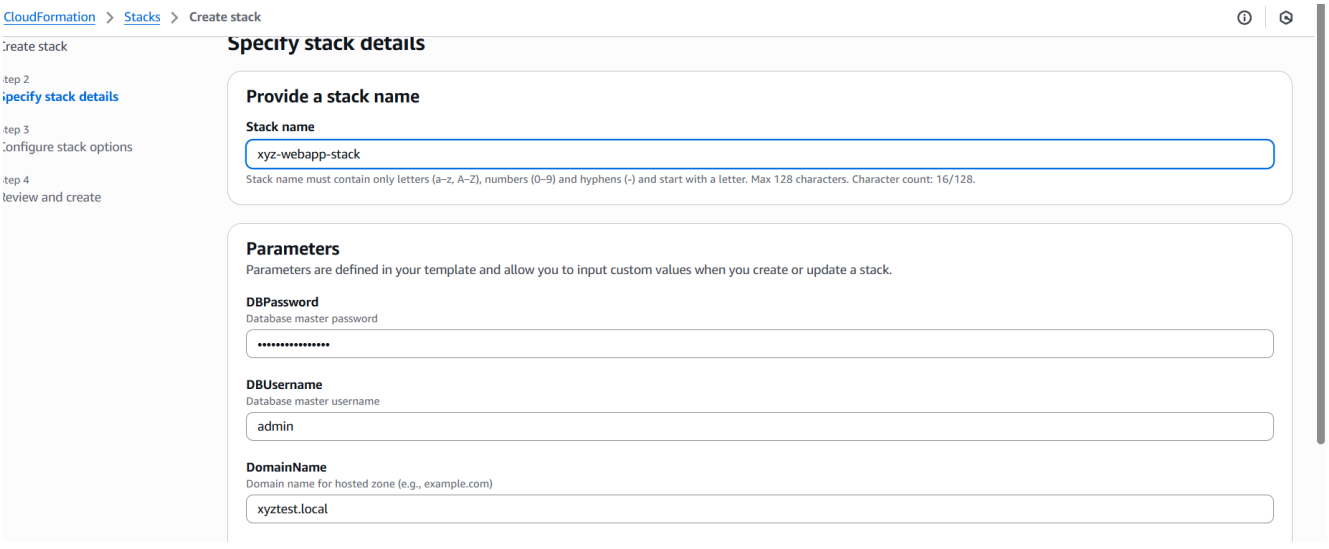
Value: !Ref HostedZone

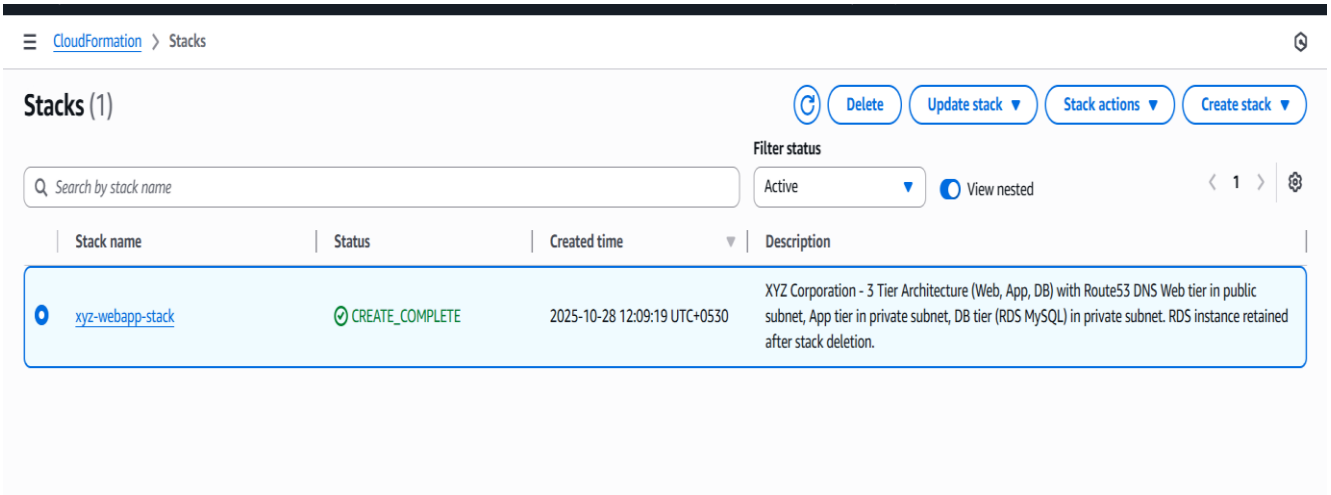


Step 2: Enter Stack Details

- **Stack Name:** xyz-webapp-stack
- **Parameters:**
  - Domain Name → e.g., xyztest.local
  - Key Name → Select your EC2 Key Pair (e.g., vik-87)
  - DB Username → admin
  - DB Password → Provide password *(minimum 8 characters)*

Click **Next**, leave all other options default, and then click **Create Stack**.

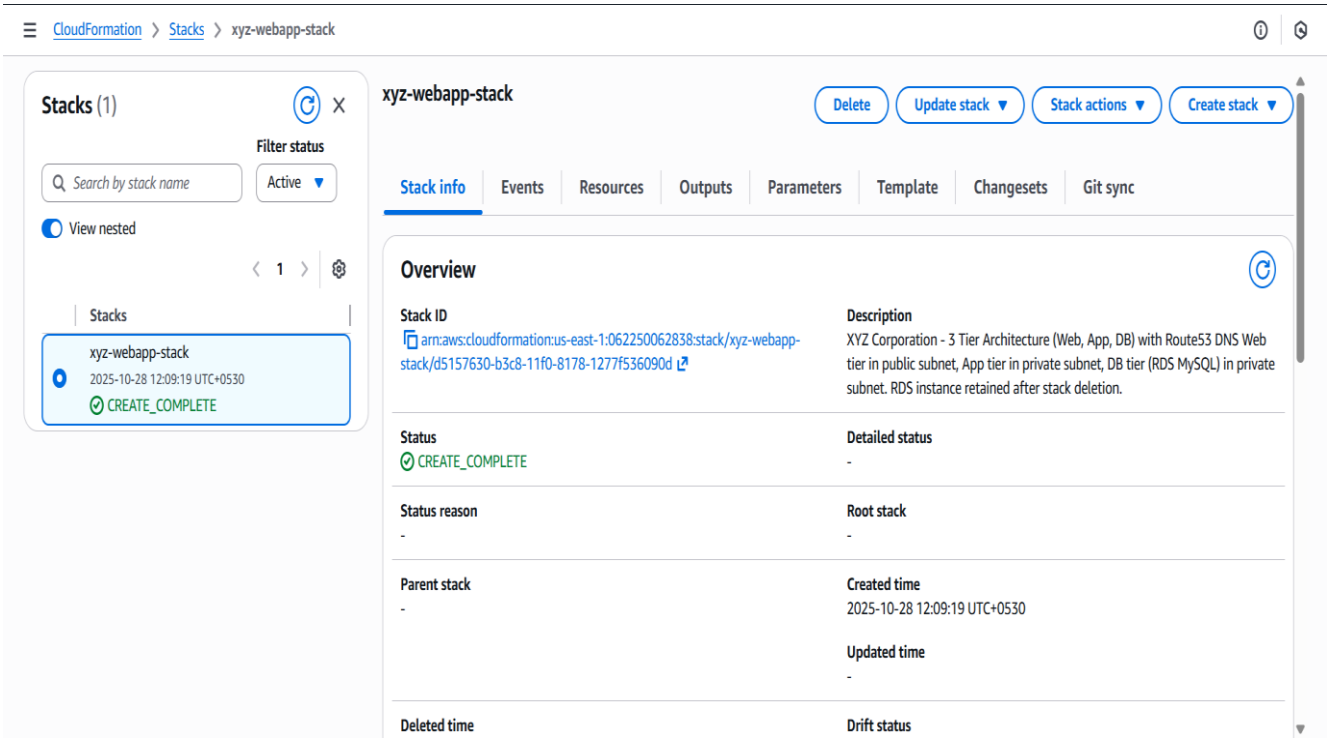


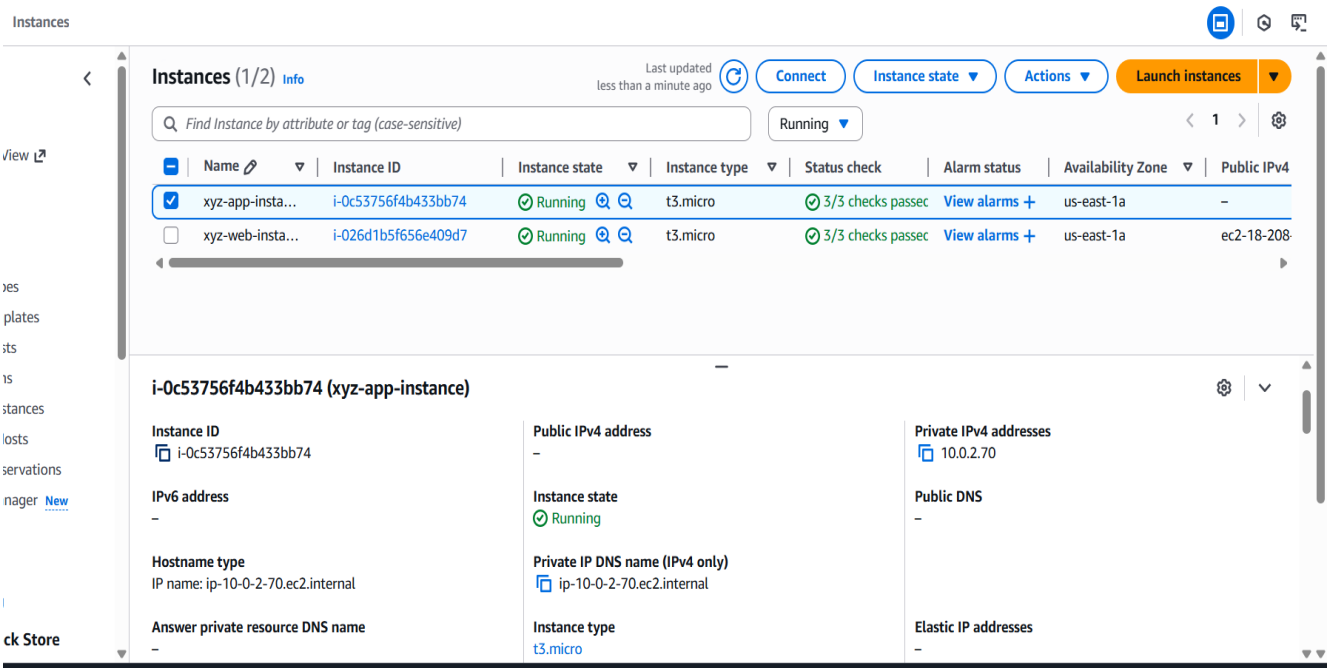


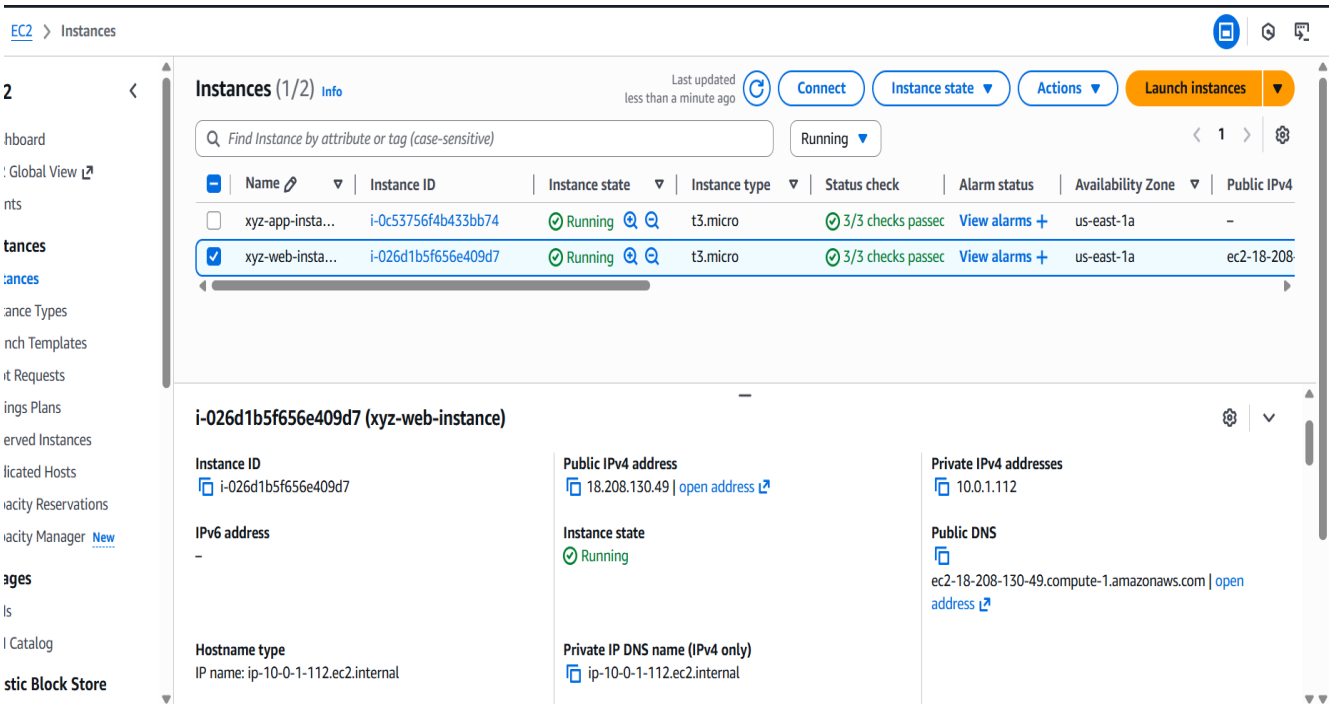
Step 3: Stack Creation Verification

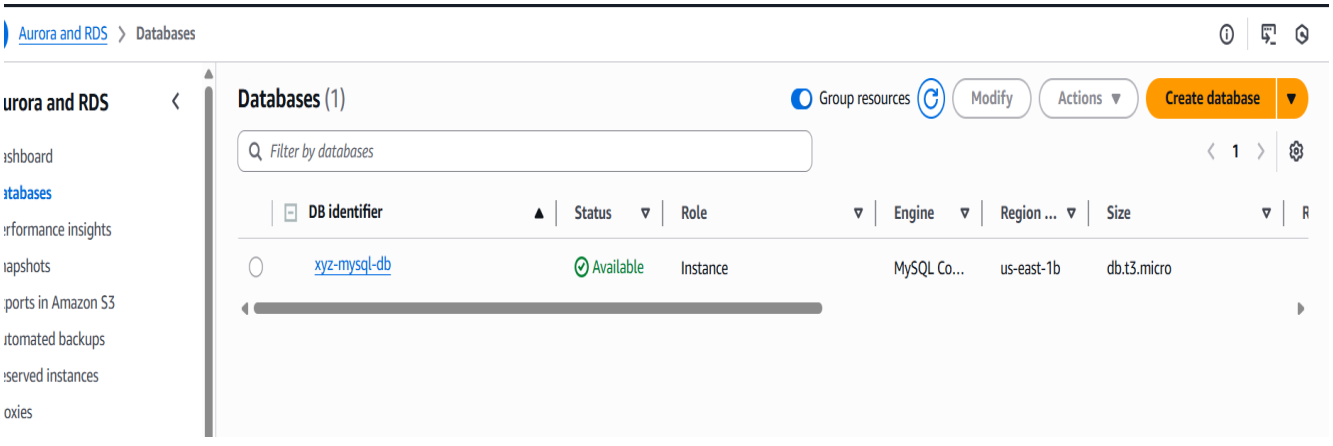
After the stack status becomes **CREATE_COMPLETE**, verify the following:

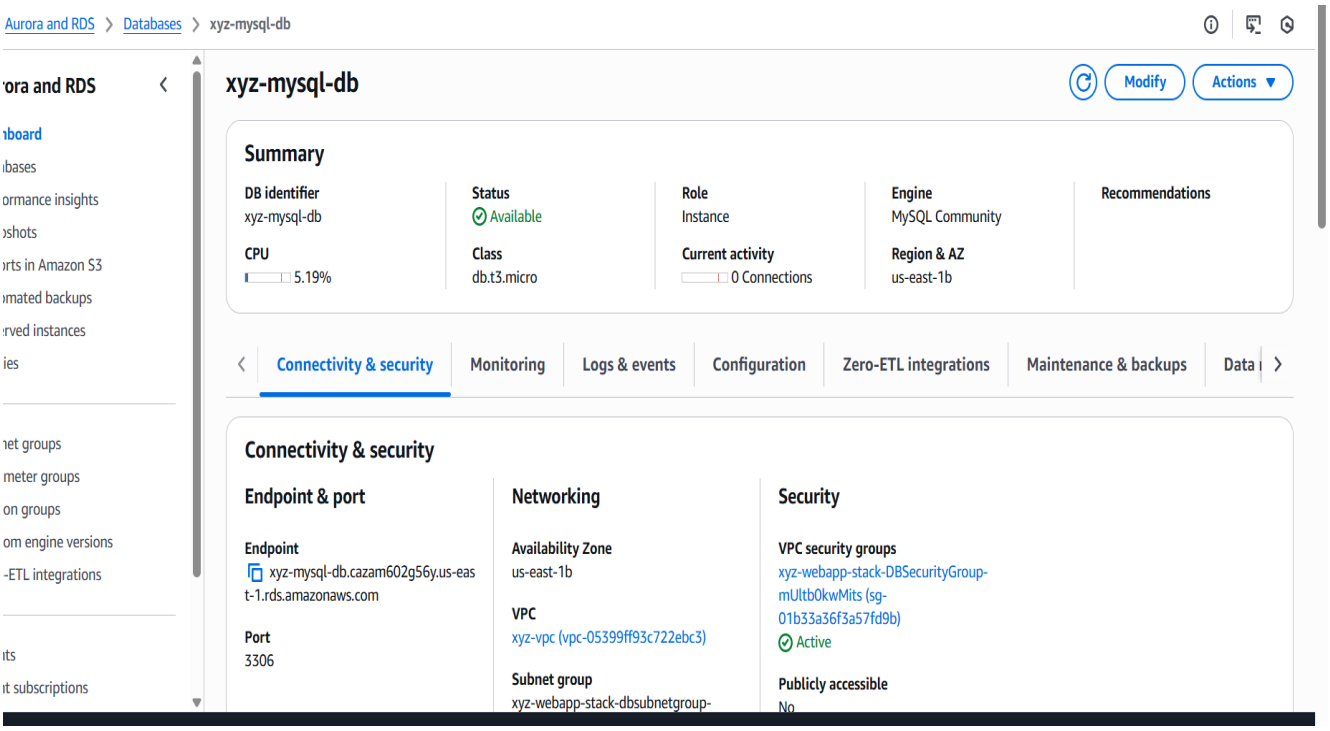
- VPC and Subnets created
- Web, App, and DB security groups
- Web and App EC2 instances
- RDS MySQL instance (private subnet)
- Route 53 hosted zone

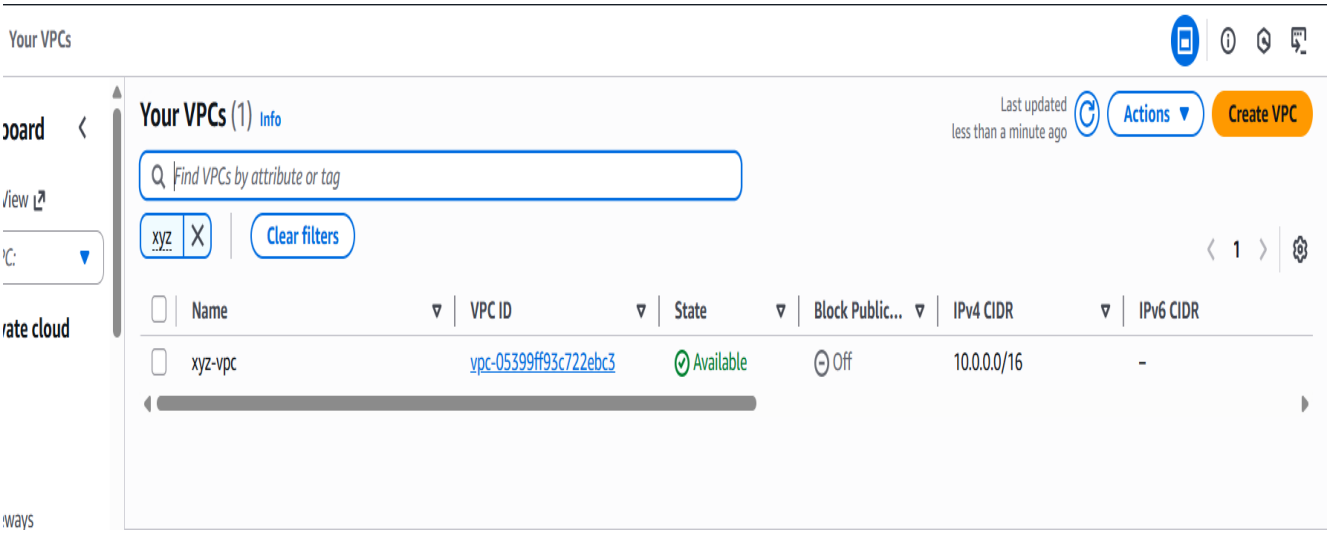


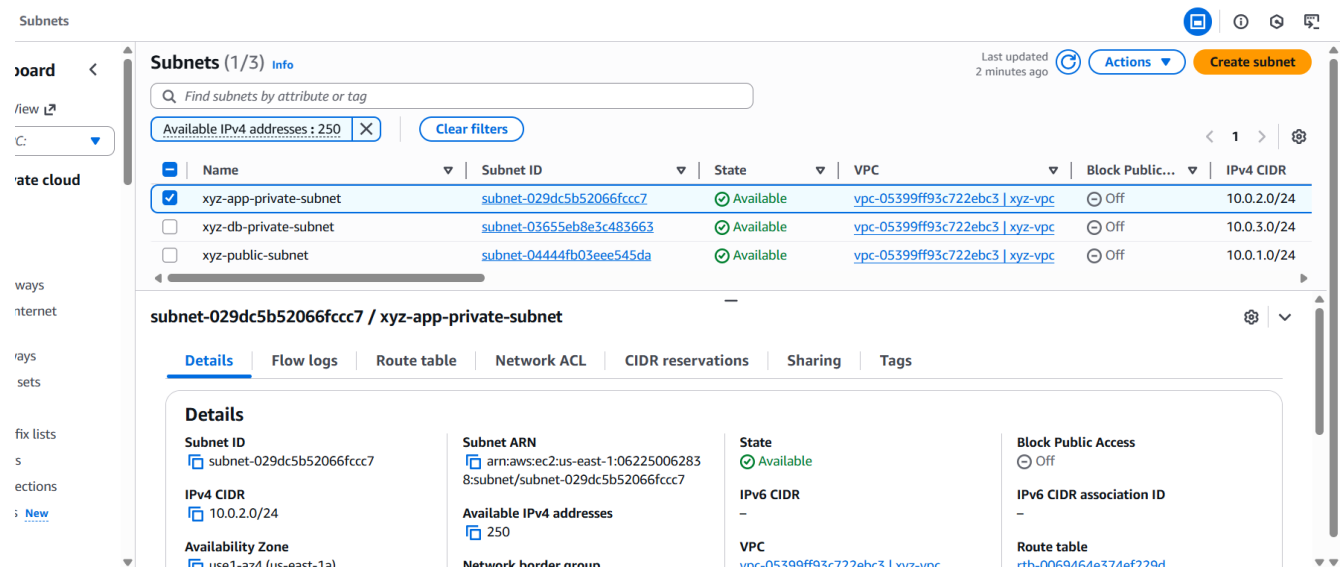

Instances
Instances (1/2) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
Running
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4
xyz-app-insta...
i-0c53756f4b433bb74
Running
t3.micro
3/3 checks passec
View alarms
us-east-1a
xyz-web-insta...
i-026d1b5f656e409d7
Running
t3.micro
3/3 checks passec
View alarms
us-east-1a
ec2-18-208
i-0c53756f4b433bb74 (xyz-app-instance)
Instance ID
i-0c53756f4b433bb74
Public IPv4 address
–
Private IPv4 addresses
10.0.2.70
IPv6 address
–
Instance state
Running
Public DNS
–
Hostname type
IP name: ip-10-0-2-70.ec2.internal
Private IP DNS name (IPv4 only)
ip-10-0-2-70.ec2.internal
Answer private resource DNS name
–
Instance type
t3.micro
Elastic IP addresses
–

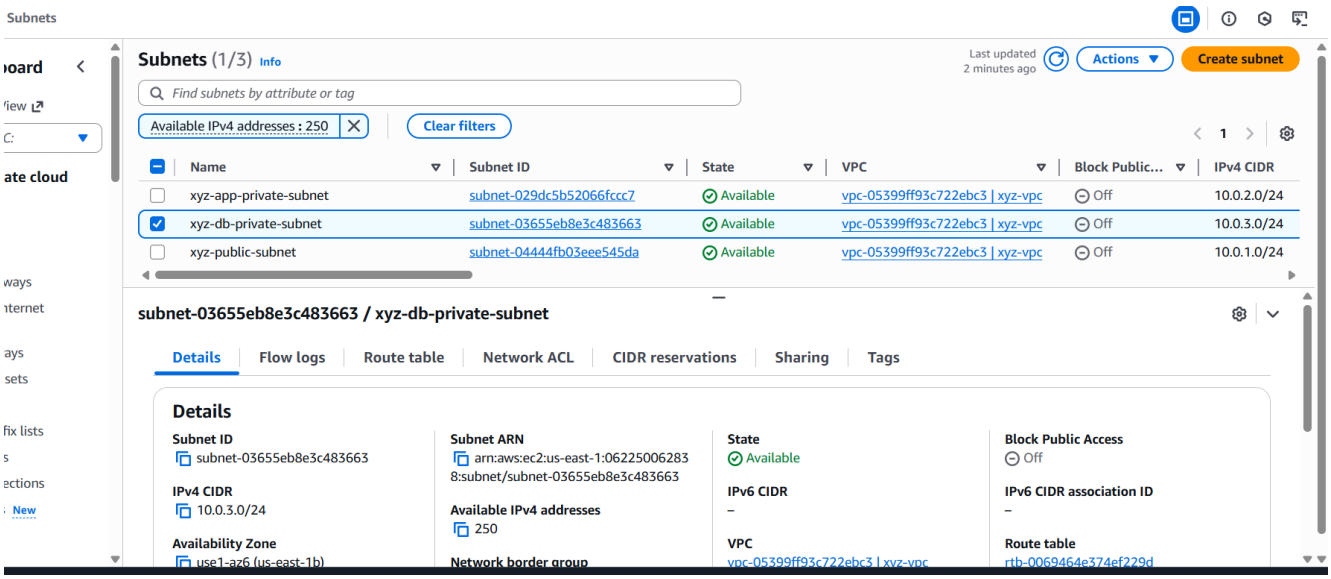

EC2 > Instances
Instances (1/2) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
Running
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4
xyz-app-insta...
i-0c53756f4b433bb74
Running
t3.micro
3/3 checks passec
View alarms
us-east-1a
xyz-web-insta...
i-026d1b5f656e409d7
Running
t3.micro
3/3 checks passec
View alarms
us-east-1a
ec2-18-208
i-026d1b5f656e409d7 (xyz-web-instance)
Instance ID
i-026d1b5f656e409d7
Public IPv4 address
18.208.130.49 | open address
Private IPv4 addresses
10.0.1.112
IPv6 address
–
Instance state
Running
Public DNS
ec2-18-208-130-49.compute-1.amazonaws.com | open address
Hostname type
IP name: ip-10-0-1-112.ec2.internal
Private IP DNS name (IPv4 only)
ip-10-0-1-112.ec2.internal

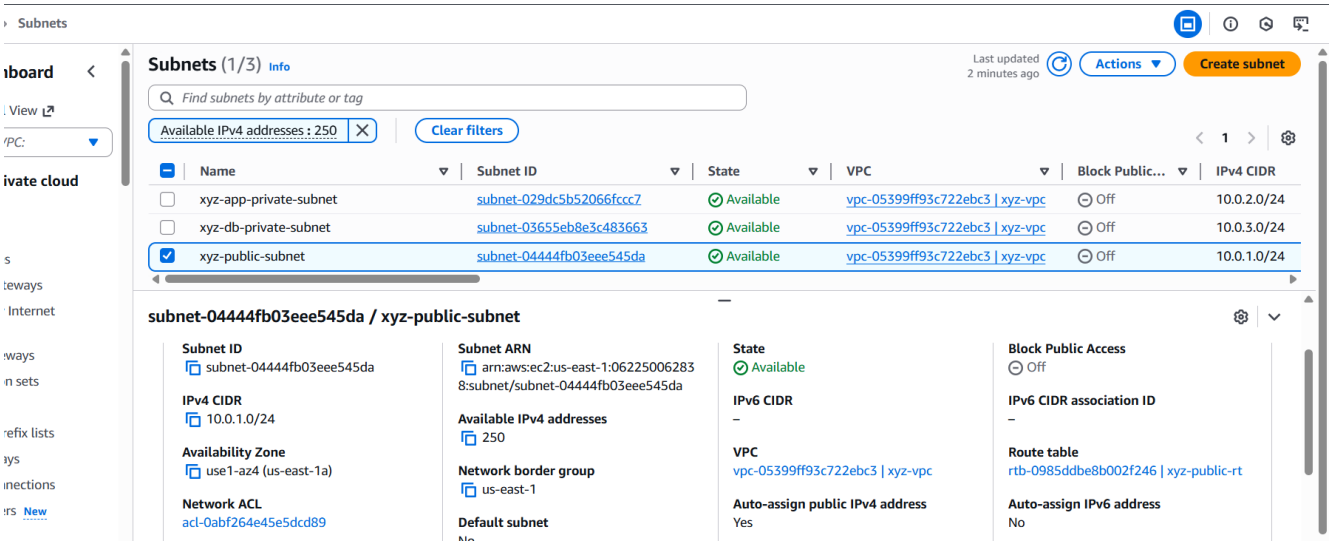

Aurora and RDS > Databases
Databases (1)
Group resources
Modify
Actions
Create database
Filter by databases
DB identifier
Status
Role
Engine
Region ...
Size
xyz-mysql-db
Available
Instance
MySQL Co...
us-east-1b
db.t3.micro

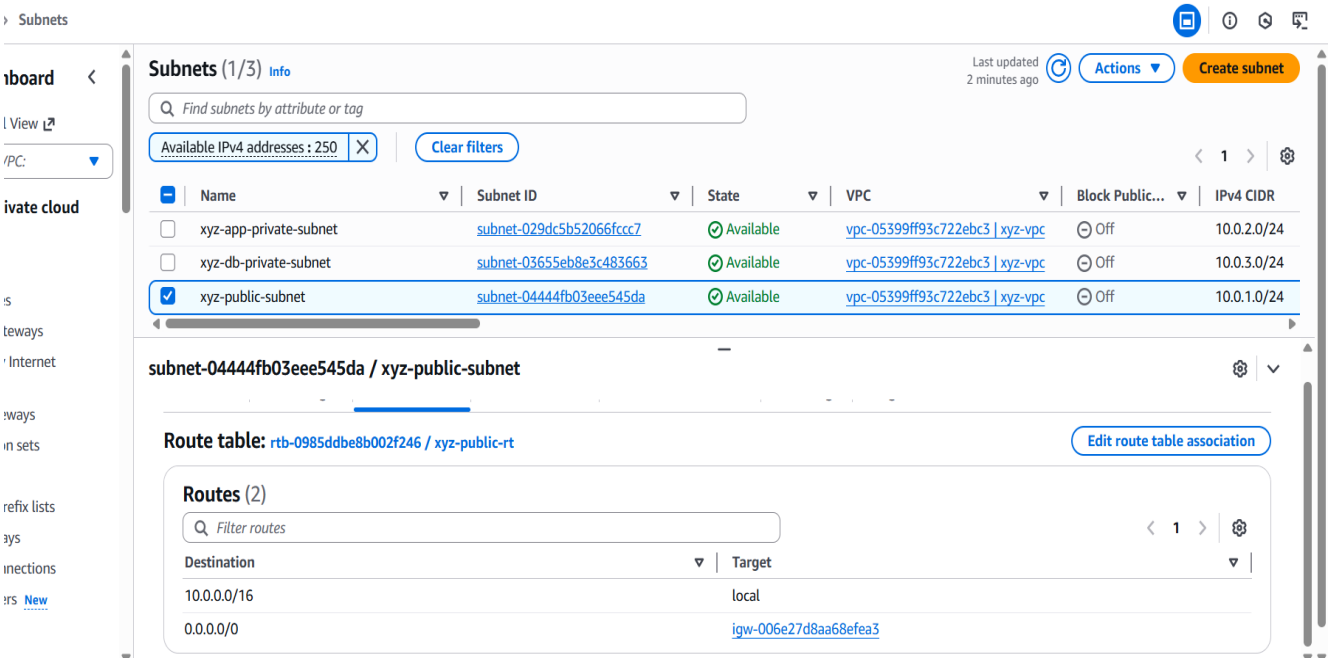

Aurora and RDS > Databases > xyz-mysql-db
xyz-mysql-db
Modify
Actions
Summary
DB identifier
xyz-mysql-db
Status
Available
Role
Instance
Engine
MySQL Community
Recommendations
CPU
5.19%
Class
db.t3.micro
Current activity
0 Connections
Region & AZ
us-east-1b
Connectivity & security
Monitoring
Logs & events
Configuration
Zero-ETL integrations
Maintenance & backups
Connectivity & security
Endpoint & port
Endpoint
xyz-mysql-db.cazam602g56y.us-east-1.rds.amazonaws.com
Port
3306
Networking
Availability Zone
us-east-1b
VPC
xyz-vpc (vpc-05399ff93c722ebc3)
Subnet group
xyz-webapp-stack-dbsubnetgroup-
Security
VPC security groups
xyz-webapp-stack-DBSecurityGroup-mUltb0kwMits (sg-01b33a36f3a57fd9b)
Active
Publicly accessible
No

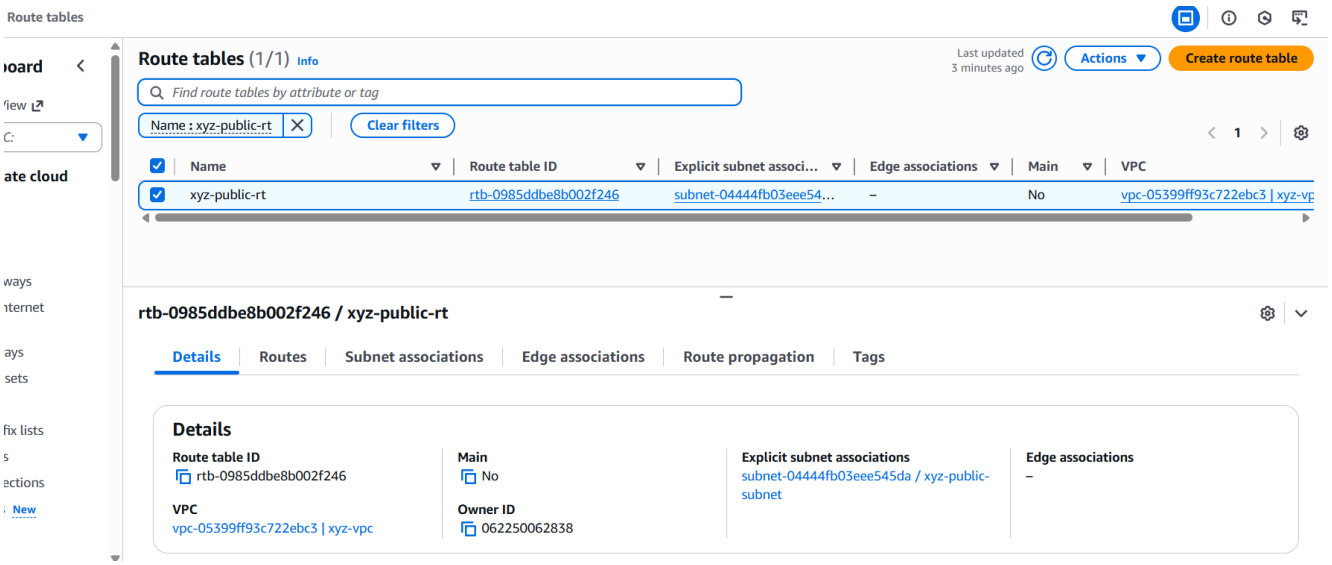

Your VPCs
Your VPCs (1) Info
Last updated
less than a minute ago
Actions
Create VPC
Find VPCs by attribute or tag
xyz
Clear filters
Name
VPC ID
State
Block Public...
IPv4 CIDR
IPv6 CIDR
xyz-vpc
vpc-05399ff93c722ebc3
Available
Off
10.0.0.0/16
-

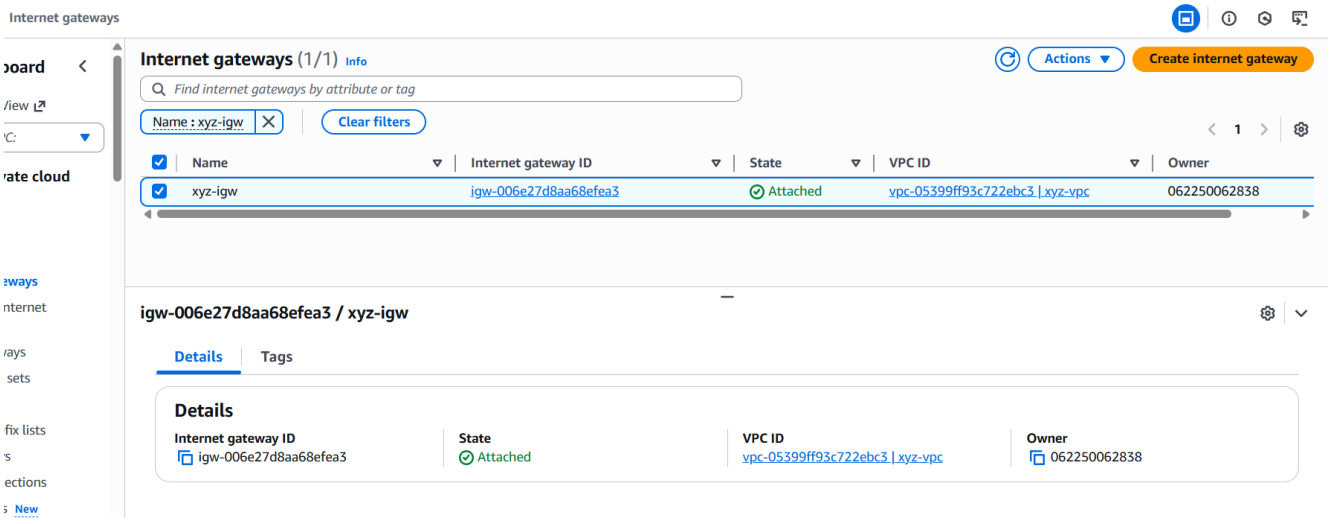

Subnets
Subnets (1/3) Info
Last updated
2 minutes ago
Actions
Create subnet
Find subnets by attribute or tag
Available IPv4 addresses : 250
Clear filters
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
xyz-app-private-subnet
subnet-029dc5b52066fccc7
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.2.0/24
xyz-db-private-subnet
subnet-03655eb8e3c483663
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.3.0/24
xyz-public-subnet
subnet-04444fb03eee545da
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.1.0/24
subnet-029dc5b52066fccc7 / xyz-app-private-subnet
Details
Flow logs
Route table
Network ACL
CIDR reservations
Sharing
Tags
Details
Subnet ID
subnet-029dc5b52066fccc7
Subnet ARN
arn:aws:ec2:us-east-1:062250068838:subnet/subnet-029dc5b52066fccc7
State
Available
Block Public Access
Off
IPv4 CIDR
10.0.2.0/24
Available IPv4 addresses
250
IPv6 CIDR
-
IPv6 CIDR association ID
-
Availability Zone
VPC
Route table

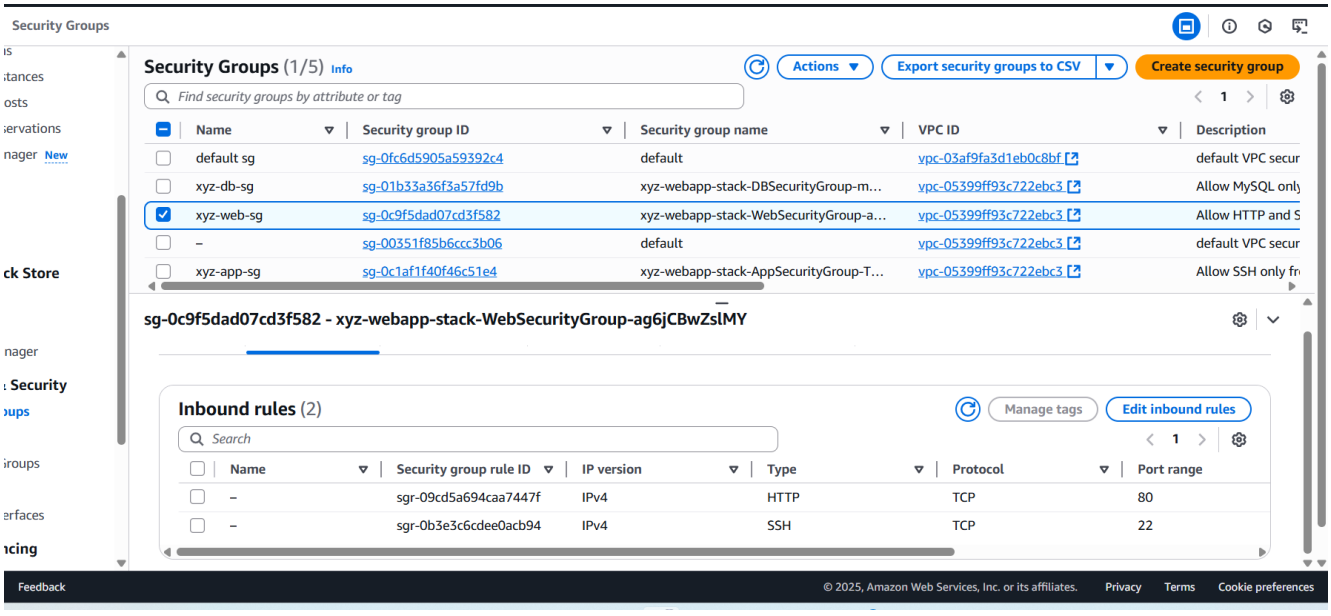

Subnets
Subnets (1/3) Info
Last updated 2 minutes ago
Actions
Create subnet
Find subnets by attribute or tag
Available IPv4 addresses : 250
Clear filters
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
xyz-app-private-subnet
subnet-029dc5b52066fccc7
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.2.0/24
xyz-db-private-subnet
subnet-03655eb8e3c483663
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.3.0/24
xyz-public-subnet
subnet-04444fb03eee545da
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.1.0/24
subnet-03655eb8e3c483663 / xyz-db-private-subnet
Details
Flow logs
Route table
Network ACL
CIDR reservations
Sharing
Tags
Details
Subnet ID
subnet-03655eb8e3c483663
Subnet ARN
arn:aws:ec2:us-east-1:06225006283
8:subnet/subnet-03655eb8e3c483663
State
Available
Block Public Access
Off
IPv4 CIDR
10.0.3.0/24
IPv6 CIDR
–
IPv6 CIDR association ID
–
Availability Zone
use1-az6 (us-east-1b)
Available IPv4 addresses
250
VPC
vpc-05399ff93c722ebc3 | xyz-vpc
Route table
rtb-0069464e374ef229d

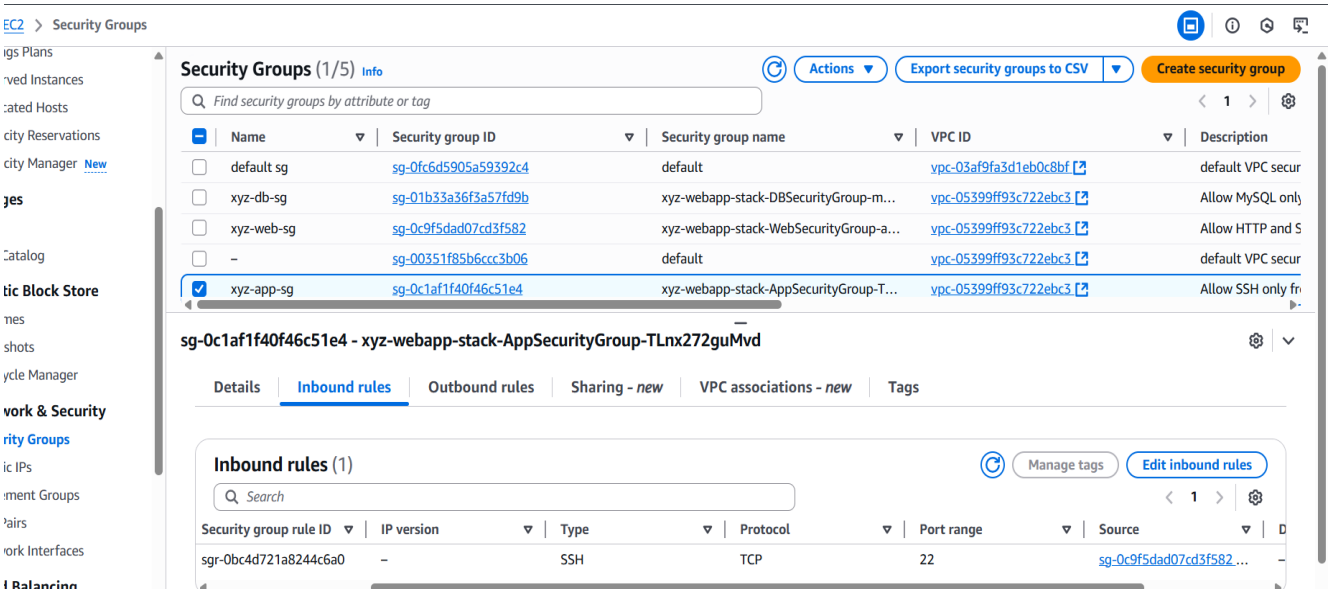

Subnets
Subnets (1/3) Info
Last updated 2 minutes ago
Actions
Create subnet
Find subnets by attribute or tag
Available IPv4 addresses : 250
Clear filters
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
xyz-app-private-subnet
subnet-029dc5b52066fccc7
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.2.0/24
xyz-db-private-subnet
subnet-03655eb8e3c483663
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.3.0/24
xyz-public-subnet
subnet-04444fb03eee545da
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.1.0/24
subnet-04444fb03eee545da / xyz-public-subnet
Subnet ID
subnet-04444fb03eee545da
Subnet ARN
arn:aws:ec2:us-east-1:06225006283
8:subnet/subnet-04444fb03eee545da
State
Available
Block Public Access
Off
IPv4 CIDR
10.0.1.0/24
IPv6 CIDR
–
IPv6 CIDR association ID
–
Availability Zone
use1-az4 (us-east-1a)
Available IPv4 addresses
250
VPC
vpc-05399ff93c722ebc3 | xyz-vpc
Route table
rtb-0985ddbe8b002f246 | xyz-public-rt
Network ACL
acl-0abf264e45e5dcd89
Network border group
us-east-1
Auto-assign public IPv4 address
Yes
Auto-assign IPv6 address
No
Default subnet

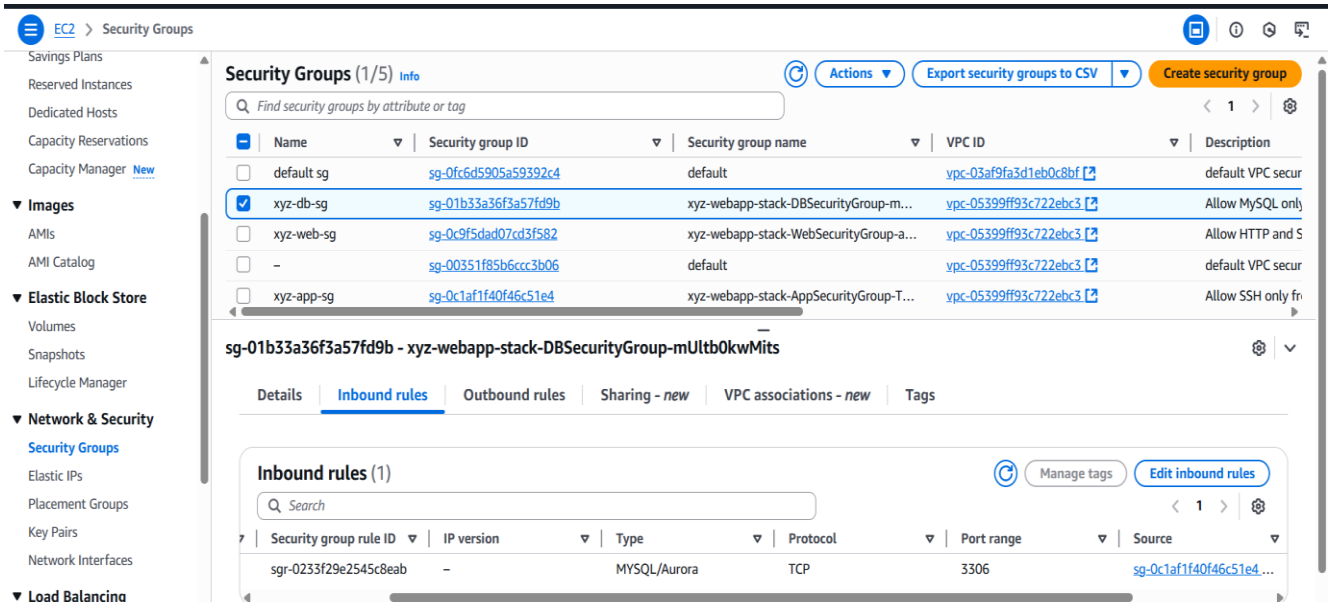

Subnets
Subnets (1/3) Info
Last updated 2 minutes ago
Actions
Create subnet
Find subnets by attribute or tag
Available IPv4 addresses : 250
Clear filters
Name
Subnet ID
State
VPC
Block Public...
IPv4 CIDR
xyz-app-private-subnet
subnet-029dc5b52066fccc7
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.2.0/24
xyz-db-private-subnet
subnet-03655eb8e3c483663
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.3.0/24
xyz-public-subnet
subnet-04444fb03eee545da
Available
vpc-05399ff93c722ebc3 | xyz-vpc
Off
10.0.1.0/24
subnet-04444fb03eee545da / xyz-public-subnet
Route table: rtb-0985ddbe8b002f246 / xyz-public-rt
Edit route table association
Routes (2)
Filter routes
Destination
Target
10.0.0.0/16
local
0.0.0.0/0
igw-006e27d8aa68efea3


Route tables
Route tables (1/1) Info
Last updated 3 minutes ago
Actions
Create route table
Find route tables by attribute or tag
Name : xyz-public-rt
Clear filters
Name
Route table ID
Explicit subnet associ...
Edge associations
Main
VPC
xyz-public-rt
rtb-0985ddbe8b002f246
subnet-04444fb03eee54...
–
No
vpc-05399ff93c722ebc3 | xyz-vp
rtb-0985ddbe8b002f246 / xyz-public-rt
Details
Routes
Subnet associations
Edge associations
Route propagation
Tags
Details
Route table ID
rtb-0985ddbe8b002f246
Main
No
Explicit subnet associations
subnet-04444fb03eee545da / xyz-public-subnet
Edge associations
–
VPC
vpc-05399ff93c722ebc3 | xyz-vpc
Owner ID
062250062838


Internet gateways
Internet gateways (1/1) Info
Actions
Create internet gateway
Find internet gateways by attribute or tag
Name : xyz-igw
Clear filters
Name
Internet gateway ID
State
VPC ID
Owner
xyz-igw
igw-006e27d8aa68efea3
Attached
vpc-05399ff93c722ebc3 | xyz-vpc
062250062838
igw-006e27d8aa68efea3 / xyz-igw
Details
Tags
Details
Internet gateway ID
igw-006e27d8aa68efea3
State
Attached
VPC ID
vpc-05399ff93c722ebc3 | xyz-vpc
Owner
062250062838


Security Groups
Security Groups (1/5) Info
Actions
Export security groups to CSV
Create security group
Find security groups by attribute or tag
Name
Security group ID
Security group name
VPC ID
Description
default sg
sg-0fc6d5905a59392c4
default
vpc-03af9fa3d1eb0c8bf
default VPC secur
xyz-db-sg
sg-01b33a36f3a57fd9b
xyz-webapp-stack-DBSecurityGroup-m...
vpc-05399ff93c722ebc3
Allow MySQL only
xyz-web-sg
sg-0c9f5dad07cd3f582
xyz-webapp-stack-WebSecurityGroup-a...
vpc-05399ff93c722ebc3
Allow HTTP and S
–
sg-00351f85b6ccc3b06
default
vpc-05399ff93c722ebc3
default VPC secur
xyz-app-sg
sg-0c1af1f40f46c51e4
xyz-webapp-stack-AppSecurityGroup-T...
vpc-05399ff93c722ebc3
Allow SSH only fr
sg-0c9f5dad07cd3f582 - xyz-webapp-stack-WebSecurityGroup-ag6jCBwZslMY
Inbound rules (2)
Manage tags
Edit inbound rules
Search
Name
Security group rule ID
IP version
Type
Protocol
Port range
–
sgr-09cd5a694caa7447f
IPv4
HTTP
TCP
80
–
sgr-0b3e3c6cdee0acb94
IPv4
SSH
TCP
22
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences





Step 4 – Web instance Validation

1. Connect via **EC2 Instance Connect** or SSH.
2. Run the following:

```
sudo yum update -y
sudo yum install -y httpd
sudo systemctl start httpd
sudo systemctl enable httpd
echo "<h1>Welcome to XYZ Web Server (Web Tier)</h1>" | sudo tee
/var/www/html/index.html
```

```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Tue Oct 28 09:11:01 2025 from 18.206.107.28
[ec2-user@ip-10-0-1-103 ~]$ sudo yum mupdate
No such command: mupdate. Please use /usr/bin/yum --help
It could be a YUM plugin command, try: "yum install 'dnf-command(mupdate)'"
[ec2-user@ip-10-0-1-103 ~]$ sudo yum update
Last metadata expiration check: 0:31:31 ago on Tue Oct 28 09:05:10 2025.
================================================================================
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

  Version 2023.9.20251027:
    Run the following command to upgrade to 2023.9.20251027:

      dnf upgrade --releasever=2023.9.20251027
```

```
  Available Versions:

  Version 2023.9.20251027:
    Run the following command to upgrade to 2023.9.20251027:

      dnf upgrade --releasever=2023.9.20251027

    Release notes:
     https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.9.20251027.html

================================================================================
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-1-103 ~]$ sudo yum install httpd
Last metadata expiration check: 0:31:40 ago on Tue Oct 28 09:05:10 2025.
Package httpd-2.4.65-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-1-103 ~]$ sudo systemctl start httpd
[ec2-user@ip-10-0-1-103 ~]$ sudo systemctl enable httpd
[ec2-user@ip-10-0-1-103 ~]$ sudo cat /var/www/html/index.html
<h1>Welcome to XYZ Corporation Web Server</h1>
```

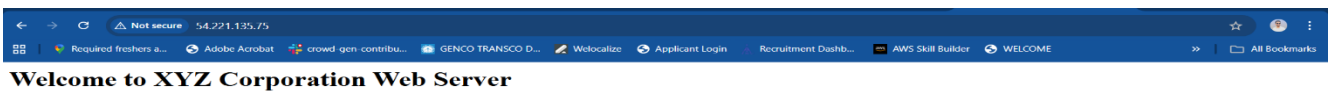
---

3. Open your browser and go to:

   http://<Public-IP>

   ✅ You should see: **Welcome to XYZ Web Server (Web Tier)**

   That confirms the **web tier** is working correctly.



Welcome to XYZ Corporation Web Server

Step 5 – Application Tier Validation

To access the **App Instance** (private subnet), you must connect **through the Web Instance** (public subnet).

1. From the Web Instance terminal:

   nano vik-87.pem

   Paste your PEM key contents into this file, then save (Ctrl + O, Enter, Ctrl + X).

2. Secure the key file permissions:

   chmod 400 vik-87.pem

3. Now connect to your App Instance using its **private IP**:

4. ssh -i vik-87.pem ec2-user@<App-Private-IP>

5. Once connected, verify you are inside the App Instance with:

6. hostname -i

```
[ec2-user@ip-10-0-1-103 ~]$ nano vik-87.pem
[ec2-user@ip-10-0-1-103 ~]$ chmod 400 vik-87.pem
[ec2-user@ip-10-0-1-103 ~]$ ssh -i vik-87.pem ec2-user@10.0.2.162
The authenticity of host '10.0.2.162 (10.0.2.162)' can't be established.
ED25519 key fingerprint is SHA256:SW1cyzuDPrQ8eKMBAy3XVcpzRGf6q9vx3oxEK7Sc50A.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.162' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-2-162 ~]$ hostname -I
10.0.2.162
[ec2-user@ip-10-0-2-162 ~]$
```

Step 6 – Database Tier Validation

From inside the **web Instance**, connect to your **RDS MySQL**:

sudo dnf install -y mariadb105

mysql -h <RDS-ENDPOINT> -u admin -p

Enter your DB password

You should now be seeing the MySQL prompt like this:

MySQL [(none)]>

That confirms successful database connectivity from the App Tier to the RDS instance.

```
[ec2-user@ip-10-0-1-103 ~]$ sudo dnf install -y mariadb105
Last metadata expiration check: 1:03:56 ago on Tue Oct 28 09:05:10 2025.
Dependencies resolved.
================================================================================
 Package                 Architecture    Version                  Repository      Size
================================================================================
Installing:
 mariadb105              x86_64          3:10.5.29-1.amzn2023.0.1 amazonlinux     1.5 M
Installing dependencies:
 mariadb-connector-c     x86_64          3.3.10-1.amzn2023.0.1    amazonlinux     211 k
```

```
[ec2-user@ip-10-0-1-103 ~]$ mariadb -h xyz-mysql-db.cazam602g56y.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 88
Server version: 8.0.42 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

Step 7 – Route 53 Validation

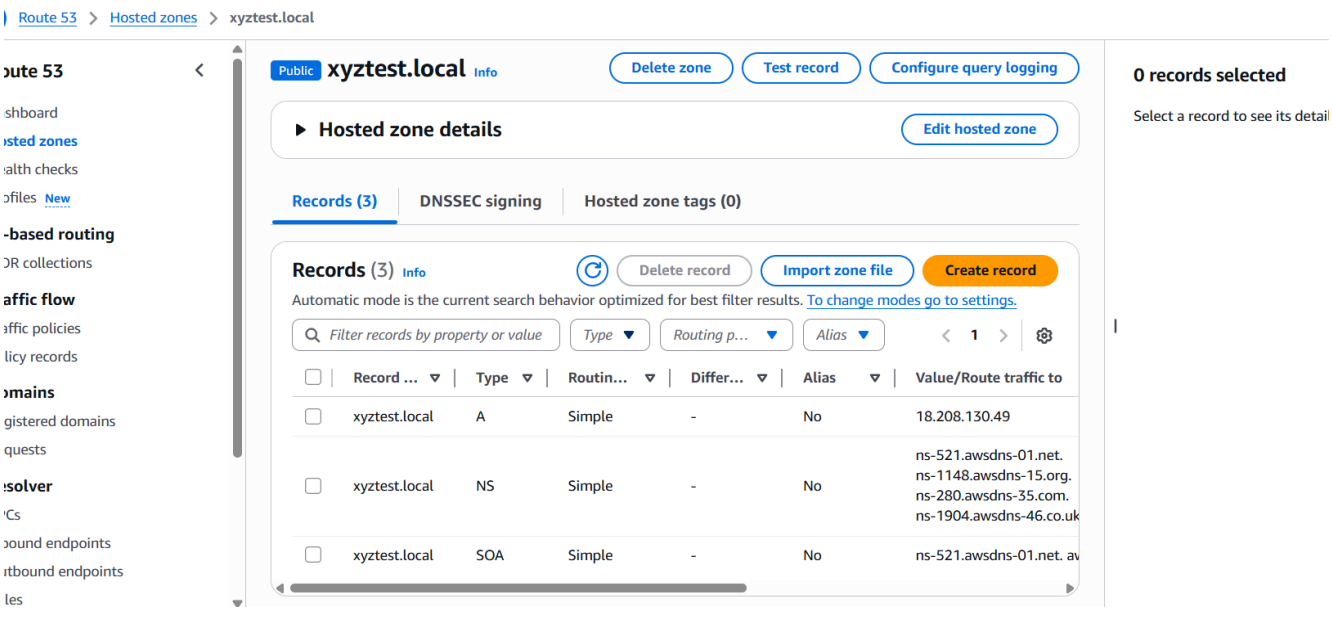
Note: Since a fake domain (xyztest.local) was used, it **won't resolve publicly**.

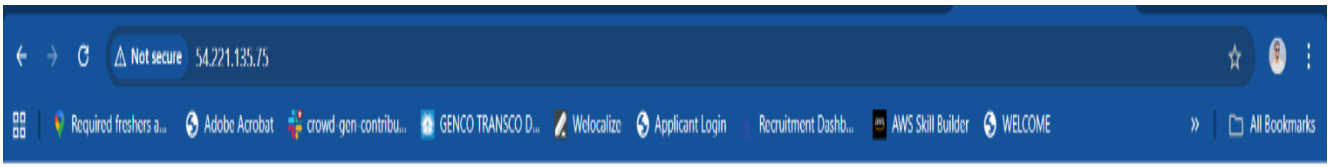
For testing, I used my **Web Instance Public IP** in the browser instead.

1. Go to **Route 53 → Hosted Zones**, confirm that your domain appears.
2. Copy your Web Instance Public IP.
3. Open it in the browser:

   http://<Web-Public-IP>

   ☑ Web page displays your welcome message.





Conclusion

- Successfully implemented a **3-tier AWS architecture** (Web / App / DB) using **CloudFormation**.
- Web and database connectivity verified.
- RDS instance retained safely after stack deletion.
- Route 53 hosted zone created successfully (local domain used for internal testing).
- This project demonstrates fully automated, reusable infrastructure for **development and testing**, eliminating manual setup by system administrators.