

Three Tire Architecture project using AWS



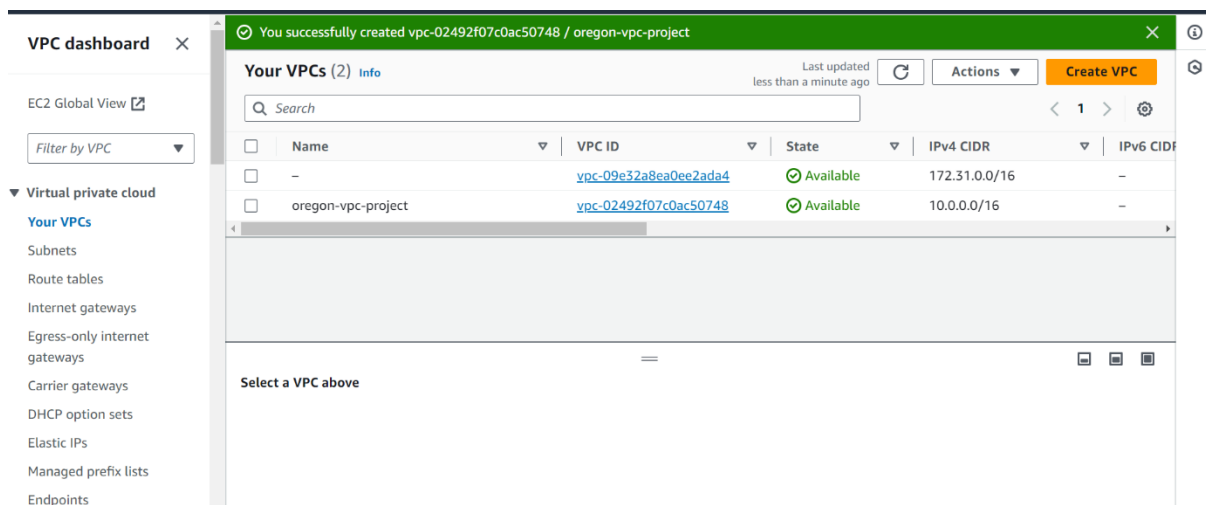
The above architecture is the “Architecting 3 Tier Architecture on AWS”.

- First tier of our architecture is a web tier. It consists of 2 public subnets in separate availability zones, and an Auto scaling group with launch template and security group.
- The second tier is an Application tier. This tier will consist of 2 Private subnets, an Autoscaling group with launch a template and same security group used in above web tier.
- The third tier is a Database tier. This tier will have an RDS(relational database service) in 2 Private subnets and an same security group used in above both the tiers.

Required AWS Services:

- 1) Virtual Private Cloud, Subnets – 6, Internet gate way – 1, Route tables – 2, Nat gate way – 1.
- 2) Elastic cloud Compute
- 3) Load Balancer
- 4) Auto- Scaling Group
- 5) Relational Database

Step 1: Create VPC and its components:.



- Go to VPC dashboard click on create VPC.

- Click on VPC only and name tag as Oregon-vpc-project
- Give IPV4 CIDR (classless inter domain routing) as 10.0.0.0/16.
- Click on VPC, it is created.

Name	Subnet ID	State	VPC
-	subnet-0dd51af8adc2c89b8	Available	vpc-09e32a8ea0ee2ada4
-	subnet-0a79b98ad3d2539d8	Available	vpc-09e32a8ea0ee2ada4
subnet-private-1-2b	subnet-02050f834ec9b7abc	Available	vpc-02a9191b34f375ff9 proje...
subnet-public-2b	subnet-097c089bfef412b9c	Available	vpc-02a9191b34f375ff9 proje...
subnet-public-2a	subnet-076580373ce28a88c	Available	vpc-02a9191b34f375ff9 proje...
subnet-private-2-2a	subnet-077d68c083eb61c72	Available	vpc-02a9191b34f375ff9 proje...
subnet-private-1-2a	subnet-0b689010acde695b6	Available	vpc-02a9191b34f375ff9 proje...
subnet-private-2-2b	subnet-00dfc2f78a7ffead5	Available	vpc-02a9191b34f375ff9 proje...
-	subnet-043268ddab8448885	Available	vpc-09e32a8ea0ee2ada4
-	subnet-0c8e3ea30392da1d1	Available	vpc-09e32a8ea0ee2ada4

- Create 6 subnets (2-public, 4-private).
- Create first subnet.
- Click on subnet, click on create subnet, select our VPC (Oregon-vpc-project).
- Give name tag as subnet public-2a, select availability zone as us-west-2a. Give CIDR as 10.0.0.0/24 and created it.
- Create Second subnet.
- Click on subnet, click on create subnet, select our VPC (Oregon-vpc-project).
- Give name tag as subnet public-2b, select availability zone as us-west-2b. Give CIDR as 10.0.15.0/24 and created it.

Create Third subnet.

- Click on subnet, click on create subnet, select our VPC (Oregon-vpc-project).

- Give name tag as subnet private-1-2a, select availability zone as us-west-2a. Give CIDR as 10.0.20.0/24 and created it.

Create fourth subnet.

- Click on subnet, click on create subnet, select our VPC (Oregon-vpc-project).
- Give name tag as subnet private-1-2b, select availability zone as us-east-1b. Give CIDR as 10.0.30.0/24 and created it.

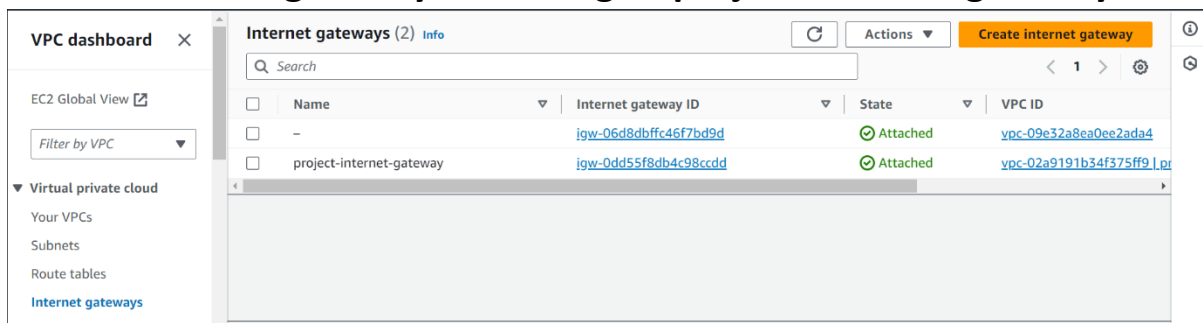
Create FIFTH subnet.

- Click on subnet, click on create subnet, select our VPC (Oregon-vpcproject).
- Give name tag as subnet-2-2a, select availability zone as us-east-1b. Give CIDR as 10.0.40.0/24 and created it.

Create SIXTH subnet.

- Click on subnet, click on create subnet, select our VPC (Oregon-vpc-project).
- Give name tag as subnet private-2-2b, select availability zone as us-east-1b. Give CIDR as 10.0.50.0/24 and created it.

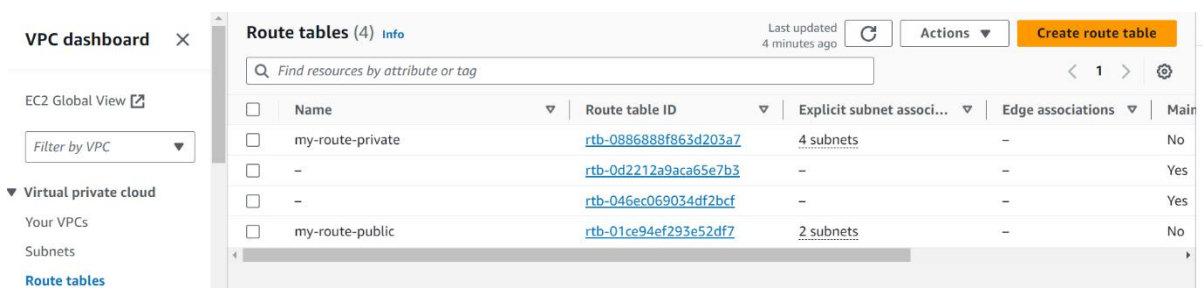
Create internet gateway, name tag as project-internet-gateway



This igw is attached to VPC.

- Go to actions in internet gate way and click on attach to VPC
- Select our VPC (Oregon-vpc-project). Click on attach internet gateway. Click on attach internet gateway

Create two route table, give name as my-route-public and my-route private



Create route table name as my-route-public

- Select our VPC (Oregon-vpc-project), create it. Click on route table id, open it.
- Go down click on edit subnet association.
- Select both public subnet and click on save association.

Go to actions click on edit routes.

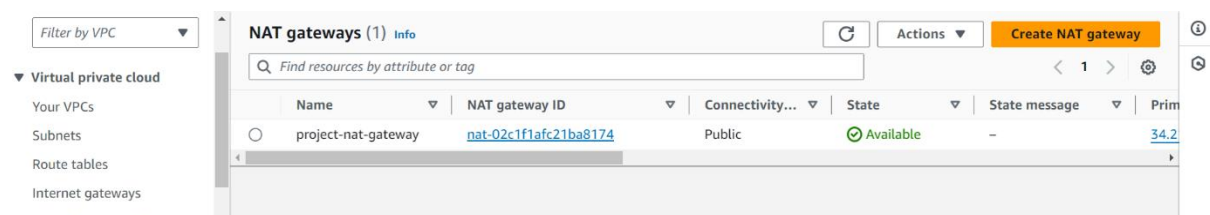
- Click on add routes give all traffic (0.0.0.0/0) and select our internet gateway, save changes.

Create private route table name as my-route-private.

- Select our VPC, (Oregon-vpc-project), create it.
- Click on route table id, open it.
- Go down click on edit subnet association.
- Select all private subnet and click on save association.

Create NAT gateway name as project-nat-gateway

- Select public subnet(private-2a).
- Select connectivity type as IPV4.
- Click on allocate Elastic IP



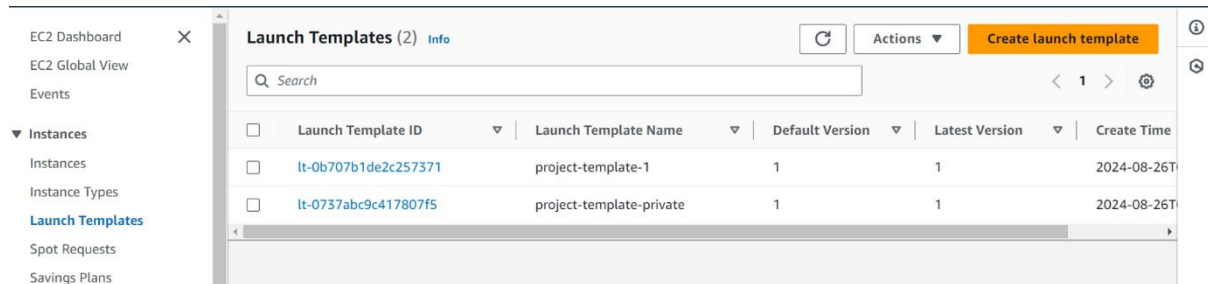
Now go to private route and click on actions.

- Click on edit routes and add route.
- Give all traffic (0.0.0.0/0) and select NAT gateway.

Create Autoscaling group.

- For creating autoscaling group we need to create an launch template, create two template
- Click on create a launch template.
- Template name as project-template-1-public & project-template-private, description as nothing.
- Select my-image.

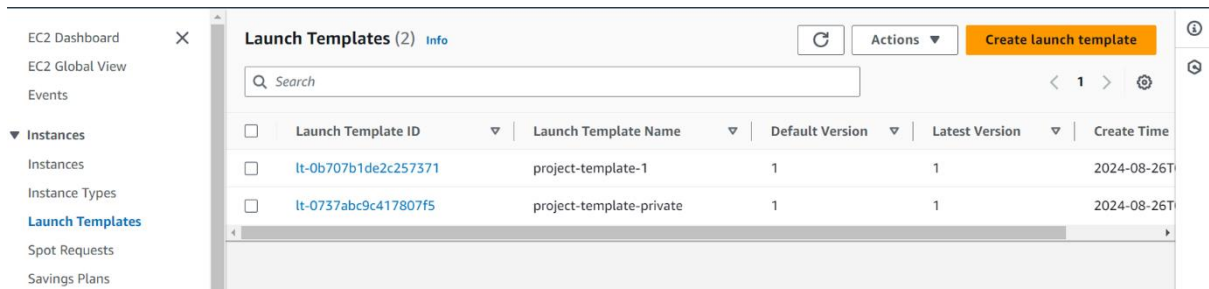
- Instance type as t2.micro and key pair as project-01.
- create security group name as project-sg
- Now click on create launch template



- After creating two templates go to auto scaling groups and
Auto scaling for the public subnets (instances subnet public-2a & subnet public-2b)
And the auto scaling for private subnet (instances subnet private-1-2a & subnet-private-1-2b)

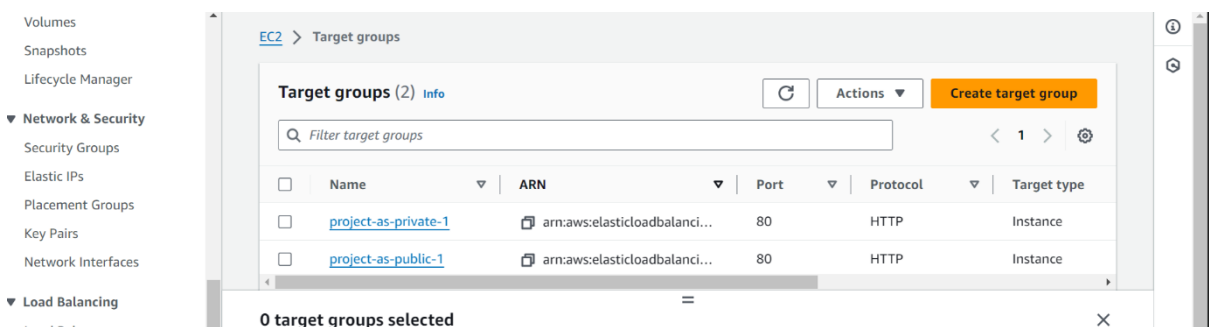
Open a auto scaling

- Click on create autoscaling group, create two auto scaling
- Give name as (project-as-public and project-as-private).
- Choose the created launch template (project-template-1 and project-template-private) and click on next.
- Select our VPC (Oregon-vpc-project),
- Click on next and click on no load balancer.
- Give desired capacity as 2 in sizing desired capacity min – 2 and max – 5 and click on next.
- Click on next and click on create auto scaling groups

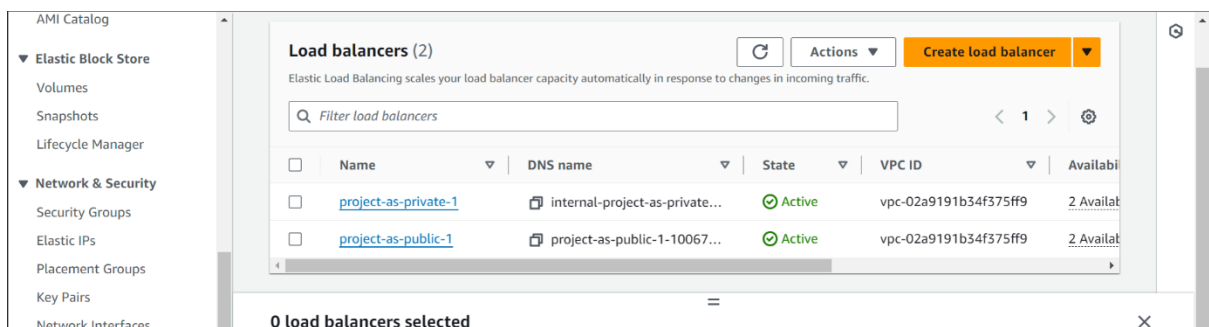


- While creating the auto scaling we can launch automatically target group, load balancer and ec2 instances
- No need to create separate target group, load balancer and ec2 instances

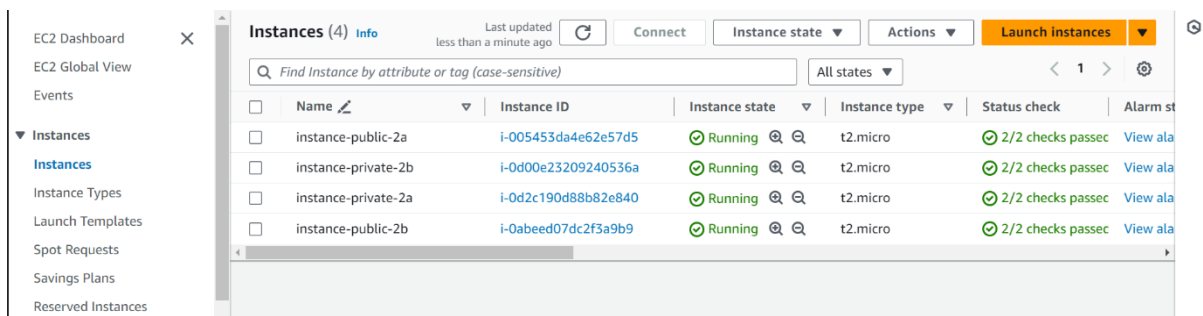
Target groups



Load balancers

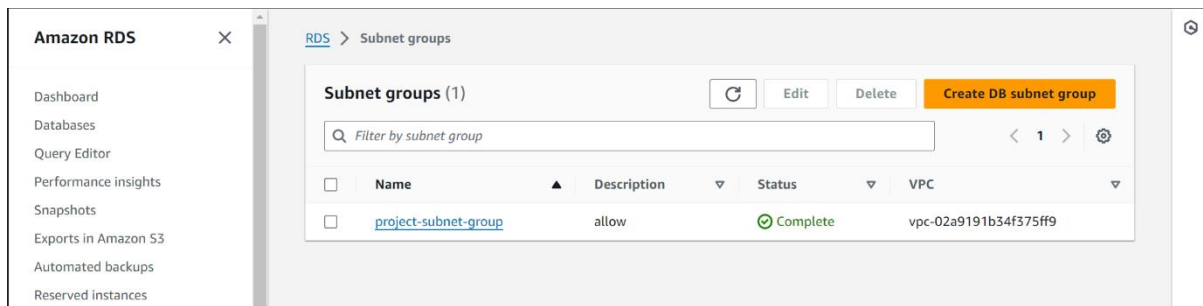


Ec2 instances



Create subnet group

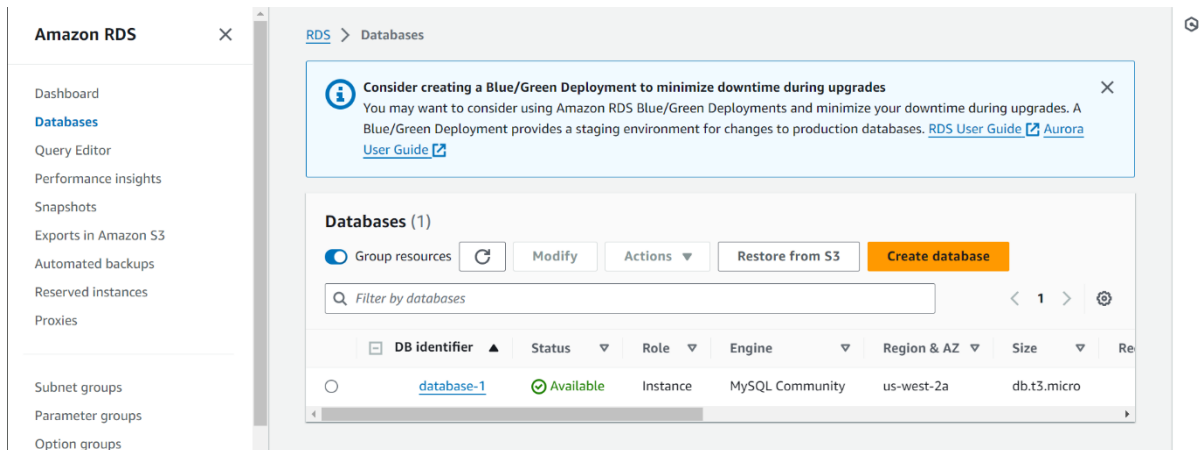
- Give name as project-subnet-group and description nothing.
- Select created VPC
- Give availability zones and select private subnets from each zone.
- Create the DB subnet group.



Create Database (RDS) Relational Databases

- Click on create database, select standard create, select engine type as MySQL.
- Select templates as production and select multi-AZ DB cluster.
- Select on self-managed, give password and confirm the password.
- Select memory optimized class.
- In connectivity, click on Don't connect to the EC2 compute resource and select created VPC (Oregon-vpc-project).

- Select subnet group (extra) and give public access as yes.
- Choose existing security group (project-sg).
- Go to VPC dashboard, click on VPC, click on actions, go to edit VPC settings and click on the enable DNS hostnames.
- Create the database



Establish connection

- Go to EC2 instance, click on instance-public-2a, open it click on connect and again click on connect. Give commands as:

1. `sudo -i` (convert from normal user to root user).
2. `apt update -y`
3. `sudo apt install mysql-server` (to install mysql)

- After installing mysql connect to RDS server through the RDS Endpoint and provide user name & password.

Now we can see that we are connected to MySQL server.

Use commands like:

- `show databases;` (to show the list of databases are there)

```
root@ip-10-0-1-21: ~  
No user sessions are running outdated binaries.  
  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
root@ip-10-0-1-21:~# sudo systemctl start mysql.service  
root@ip-10-0-1-21:~# chmod 400 project-01.pem  
root@ip-10-0-1-21:~# mysql -h database-1.cjc284ooo5a1.us-east-1.rds.amazonaws.com -u admin -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 26  
Server version: 8.0.35 Source distribution  
  
Copyright (c) 2000, 2024, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
+-----+
```

- create database project; (to create a new database)
- use project; (server will take that database to use)
- create a table using this command

CREATE TABLE Persons

```
CREATE TABLE Persons (  
    PersonID int,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255)  
);
```

Insert the fiels and the values

INSERT INTO persons (personsID,LastName, FirstName, Address, City ,)
VALUES ('101', 'pedapudi', 'manisha', 'Hyd', 'gunture');

```
mysql> create database manisha;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| manisha |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> use manisha;
Database changed
mysql> CREATE TABLE Persons (
->     PersonID int,
->     LastName varchar(255),
->     FirstName varchar(255),
->     Address varchar(255),
->     City varchar(255)
-> );
Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO Persons(PersonID,LastName,FirstName,Address,City)
```

```
5 rows in set (0.00 sec)

mysql> use manisha;
Database changed
mysql> CREATE TABLE Persons (
->     PersonID int,
->     LastName varchar(255),
->     FirstName varchar(255),
->     Address varchar(255),
->     City varchar(255)
-> );
Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO Persons(PersonID,LastName,FirstName,Address,City)
-> Values('101','pedapudi','manishaa','hyd','guntur');
Query OK, 1 row affected (0.01 sec)

mysql> select * from Persons;
+-----+-----+-----+-----+-----+
| PersonID | LastName | FirstName | Address | City |
+-----+-----+-----+-----+-----+
| 101 | pedapudi | manishaa | hyd | guntur |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```