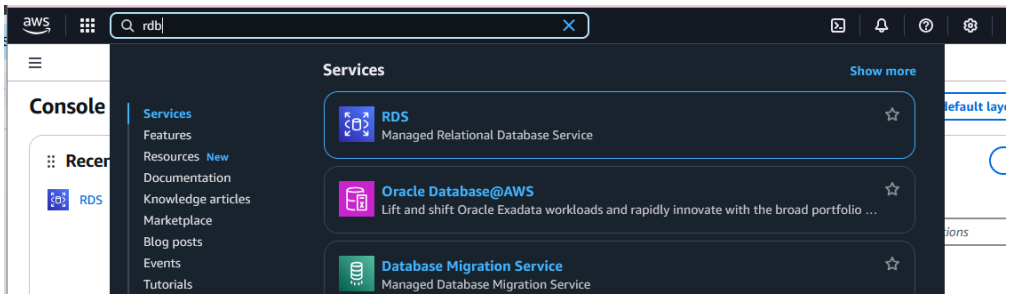
Seneca backend technical - Deployment Task

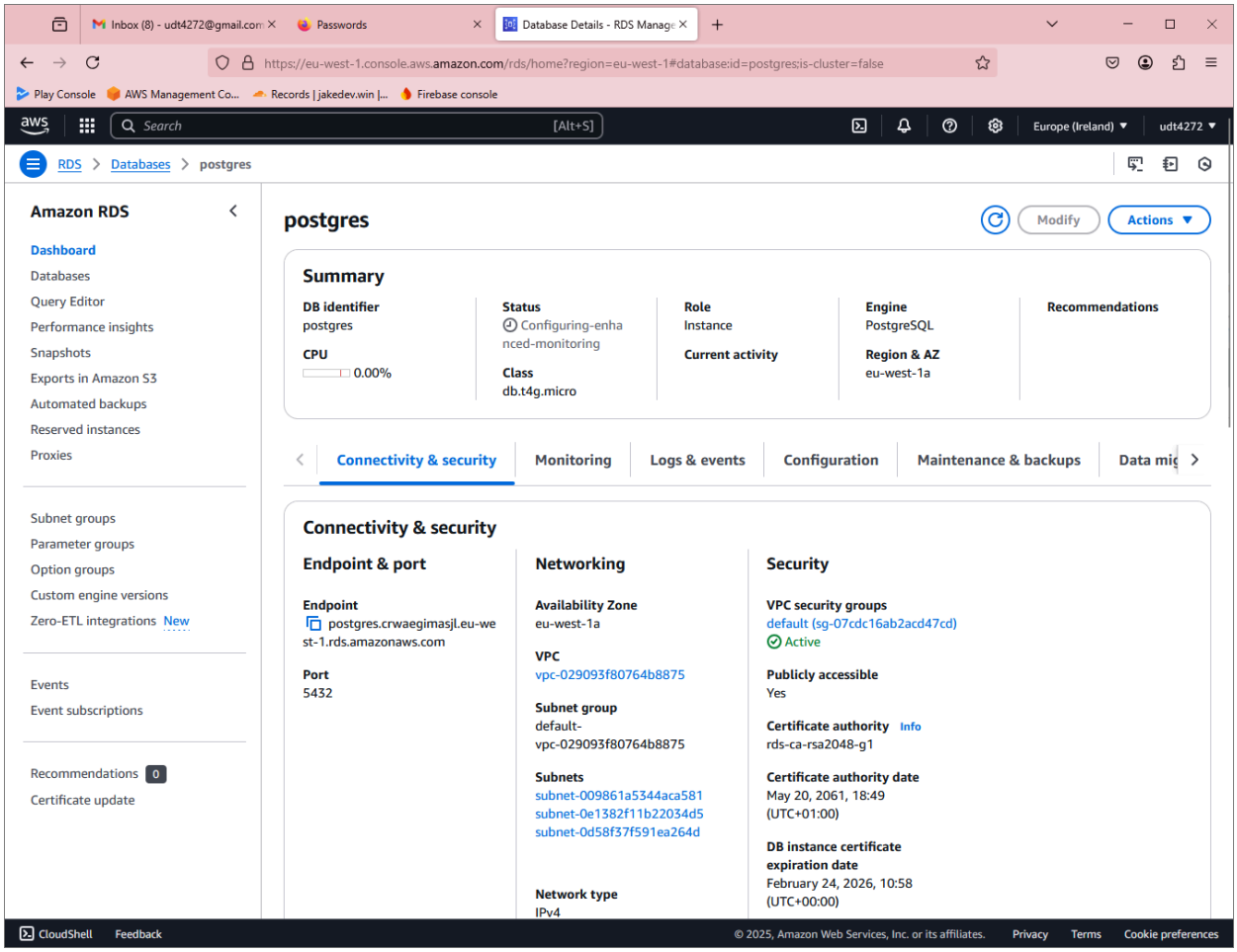
Database Deployment

1. Sign in to AWS console
2. Select 'RDS'

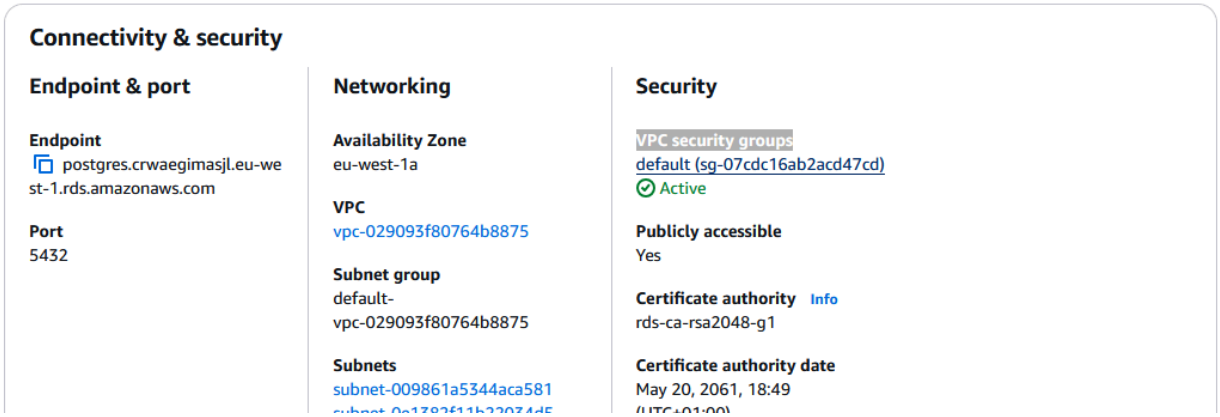


3. Select 'Databases' > Create database
4. Database Info options (leave all other options default)
    a. Choose a database creation method - Standard create
    b. Engine options>Engine type - PostgreSQL
    c. Templates - Free tier
    d. Settings>DB instance identifier - postgres
    e. Settings>Master username - postgres
    f. Settings>Master password/ Confirm master password - password
    g. Connectivity>Public access - yes
    h. Connectivity>Additional configuration>Database port - 5432

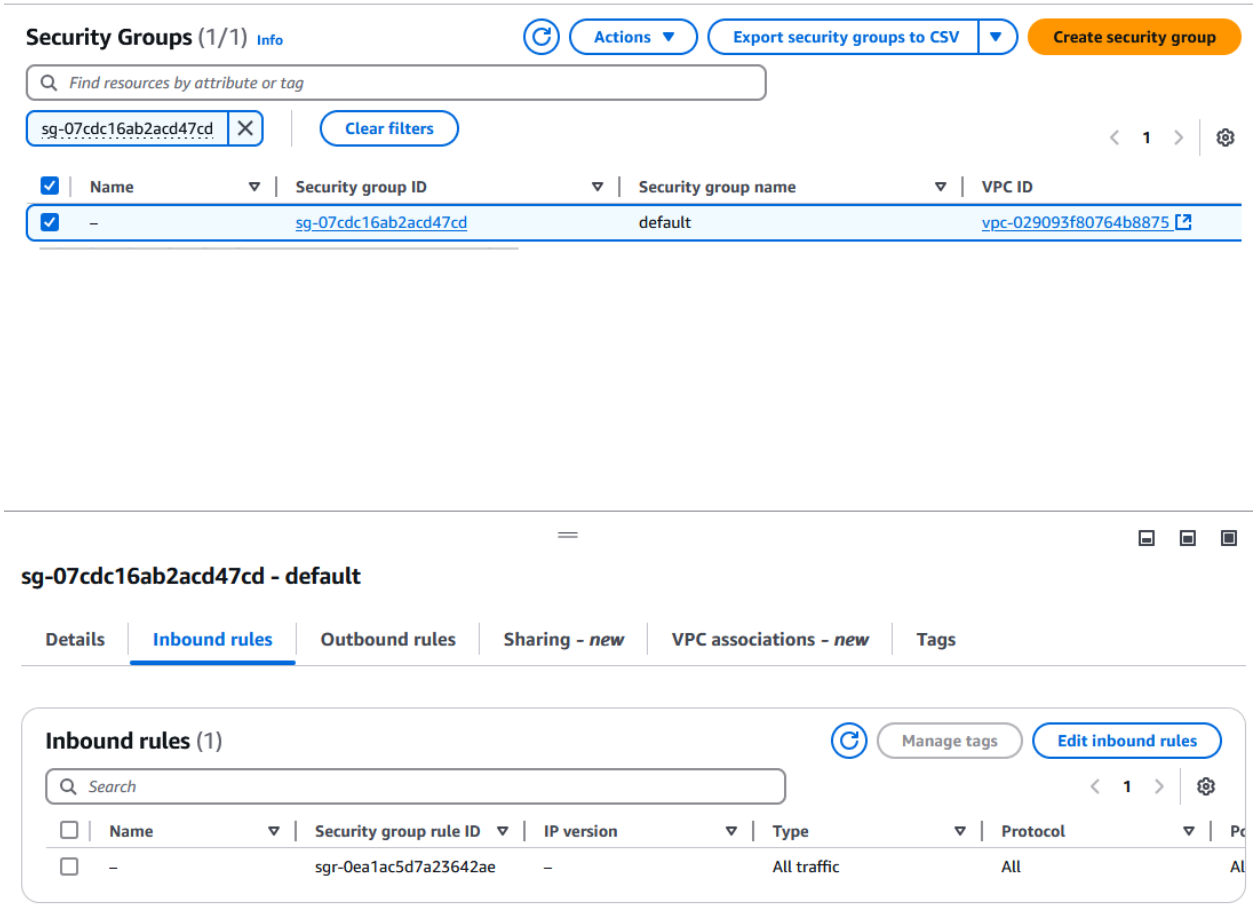
5. We can now see the DB has been created when it is ‘Active’’



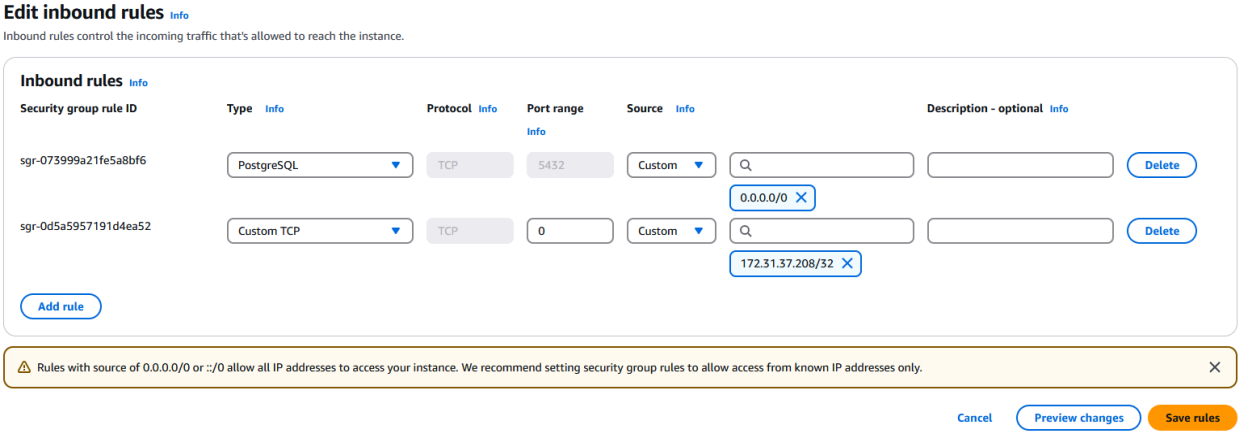
6. We now need to edit the security group, Select the ‘VPC security groups’ link



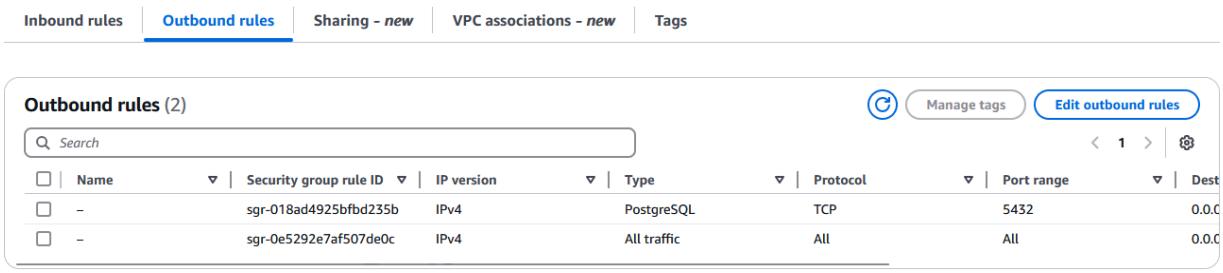
7. In Security Groups, select the corresponding default security group, and then select ‘Inbound Rules’ below. The Select ‘Edit inbound rules’



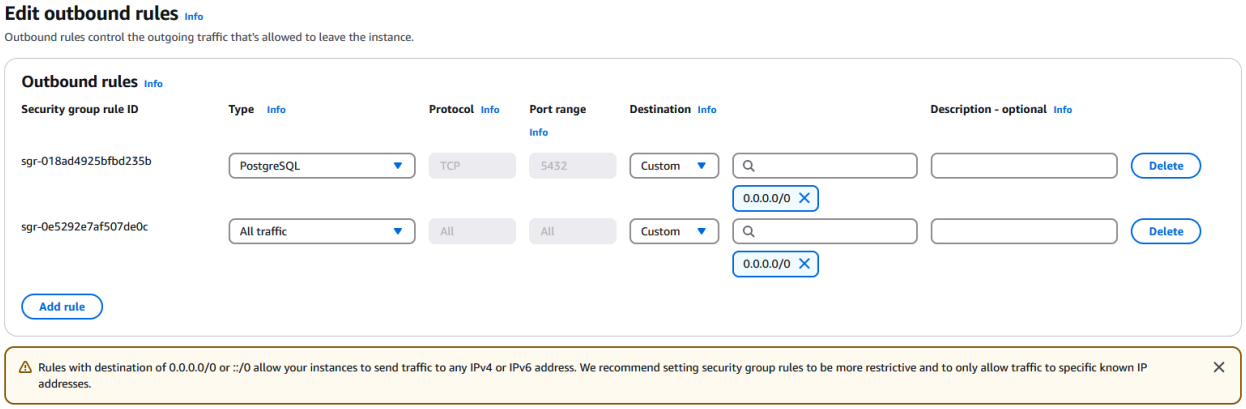
8. Delete the current rule.
9. Select ‘Add Rule’, for the following:



10. Now Select Outbound Rules

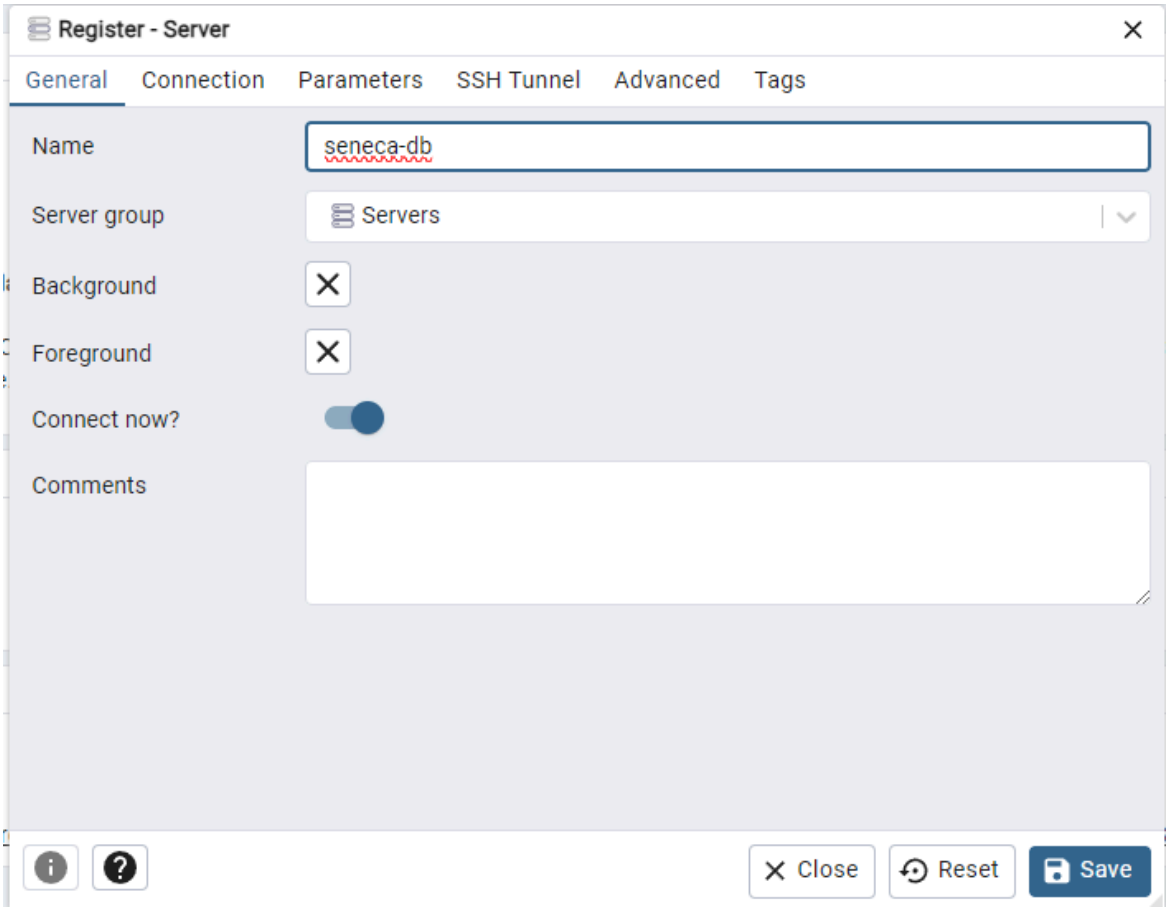


11. Input the following:

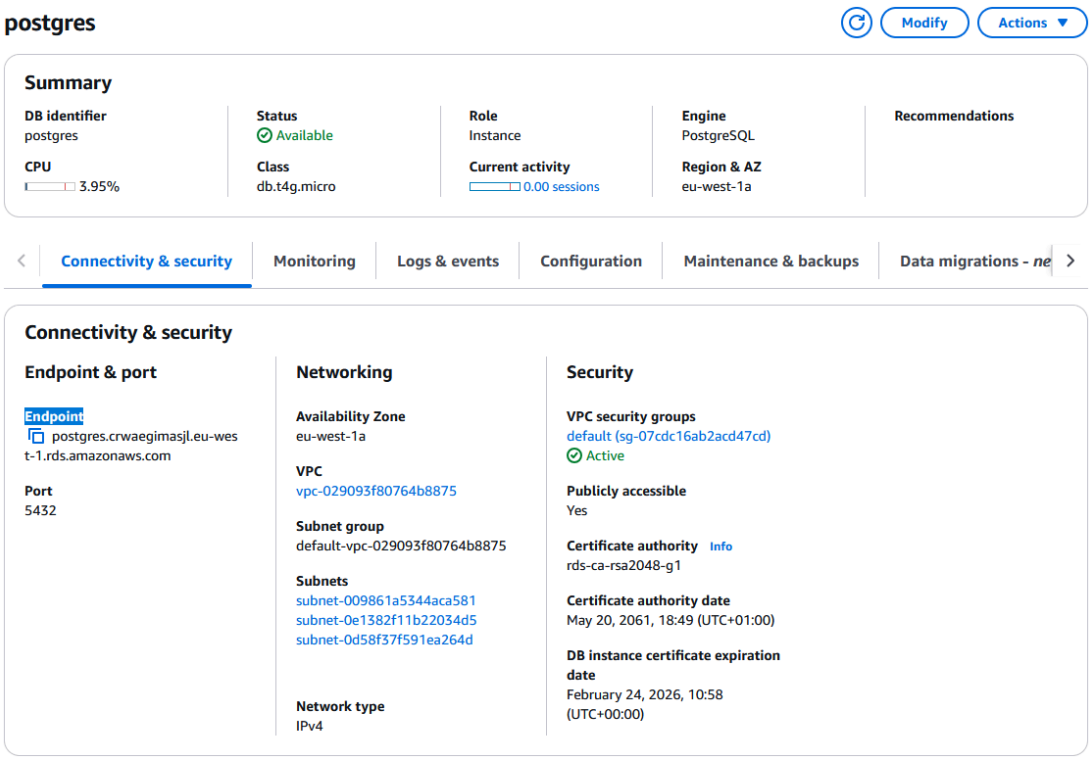


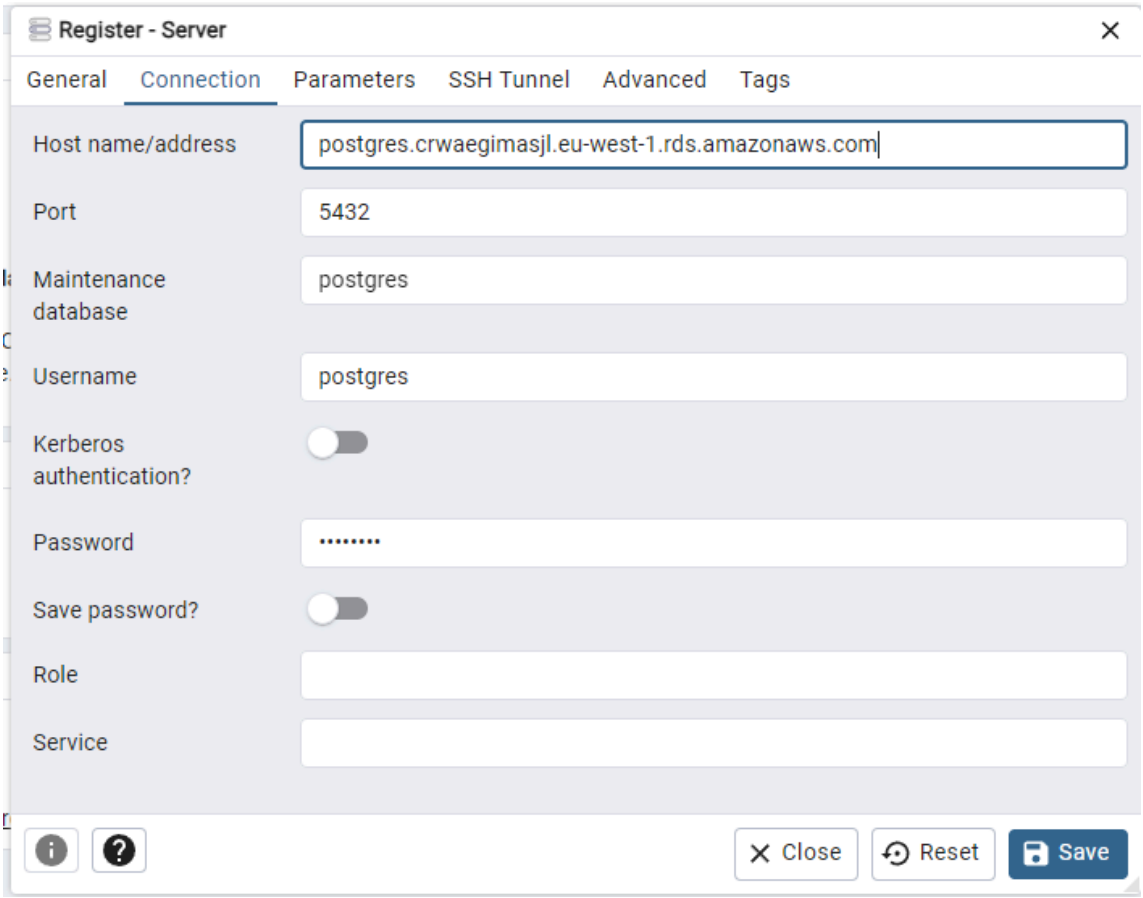
12. Let's Test Connection with ‘pgAdmin’ (or any other SQL-based db management tool)

a. Object > Register >Server



b. Now fill out the Connection details, you can find the Host name/address on the created database with RDS on the AWS console under ‘Endpoint’ here:



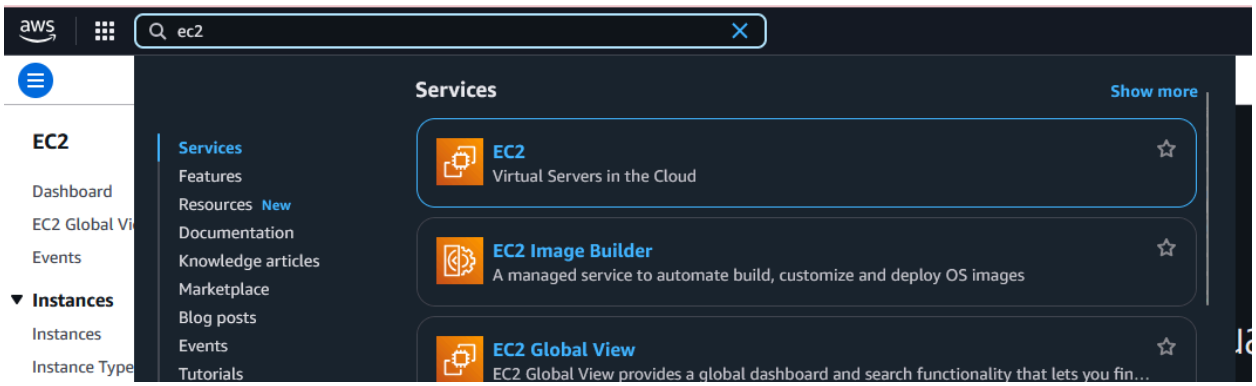


c. If you followed Step 4 ‘Database Info options’, the following details should follow

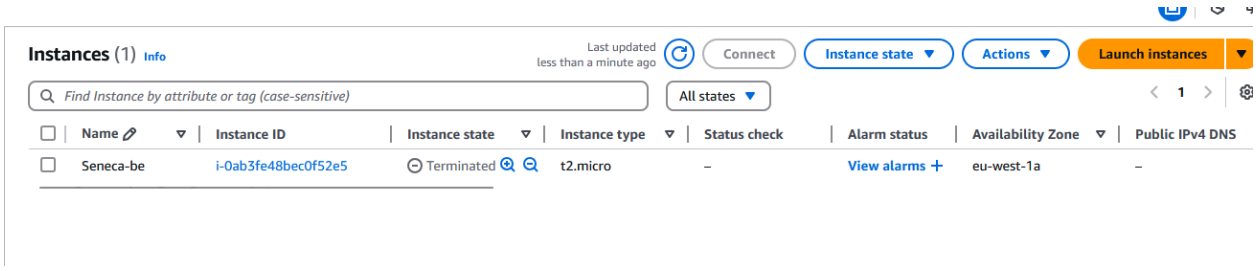
i. Port - 5432 (Connectivity>Additional configuration>Database port)
ii. Maintenance database - postgres (Settings>DB instance identifier)
iii. Username - postgres (Settings>Master username)
iv. Password - password (Settings>Master password/ Confirm master password)

Deploying Backend EC2

1. Navigate to EC2 in AWS console

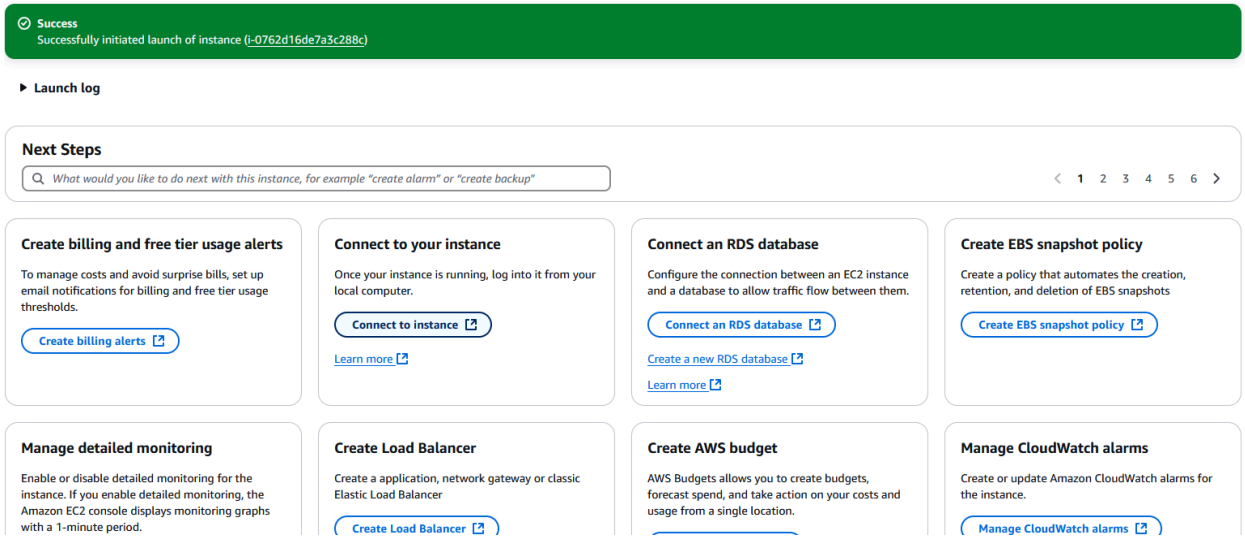


2. Select 'Launch Instances'

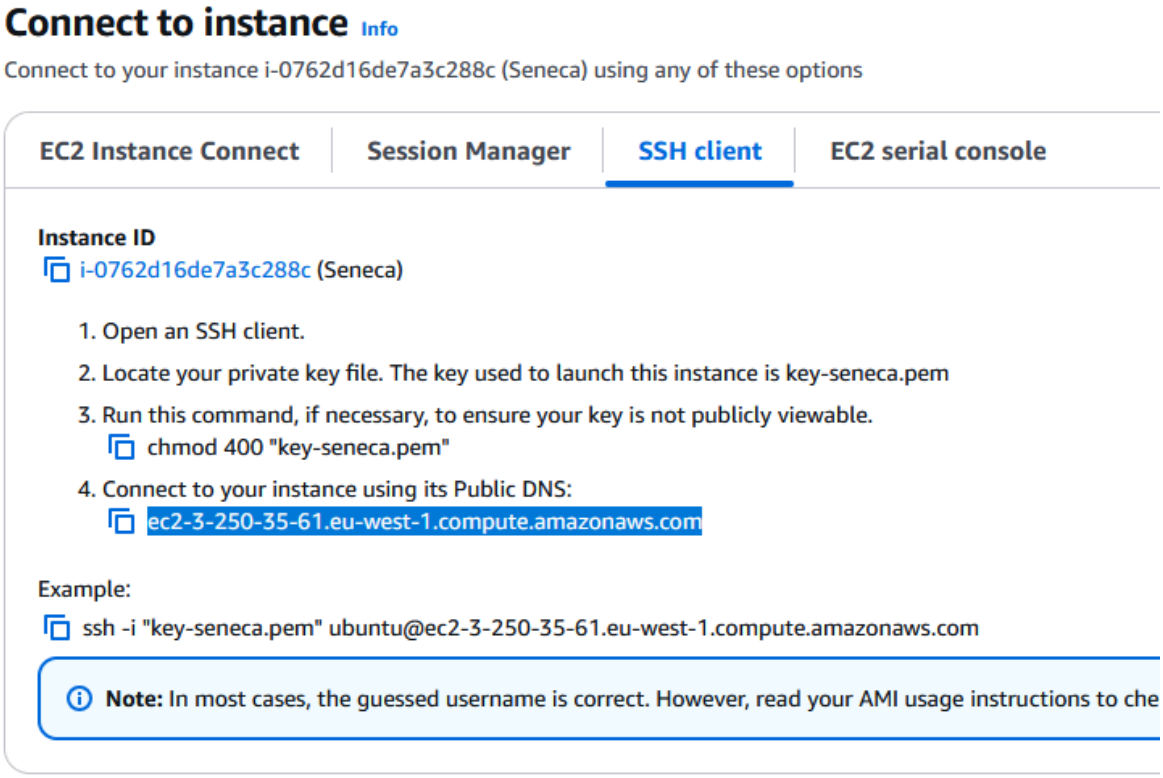


3. Fill out instance Information (leave other setting default / free tier)
   a. Name and tags>Name - Seneca
   b. Application and OS Images (Amazon Machine Image) - Ubuntu
   c. Amazon Machine Image (AMI) - Free tier
   d. Key pair (login)>Create new key pair
      i. Key pair name - key-seneca
      ii. Select 'Create key pair'
      iii. You should now see 'key-seneca' under the Key pair name on the instance information page.
   e. Network settings
      i. Create security group
      ii. Allow SSH traffic from - Anywhere
      iii. Allow HTTPS traffic from the internet
      iv. Allow HTTP traffic from the internet
   f. Select 'Launch Instance'

4. You have successfully created the instance, now let's select 'Connect to instance'



5. Install and open git bash if you dont already have it
6. Navigate to the folder with 'key-seneca.pem', for me it is Downloads (cd ~/Downloads)
7. Copy the example option of the Connect to Instance page and paste it into git bash



8. We are now connected to our instance

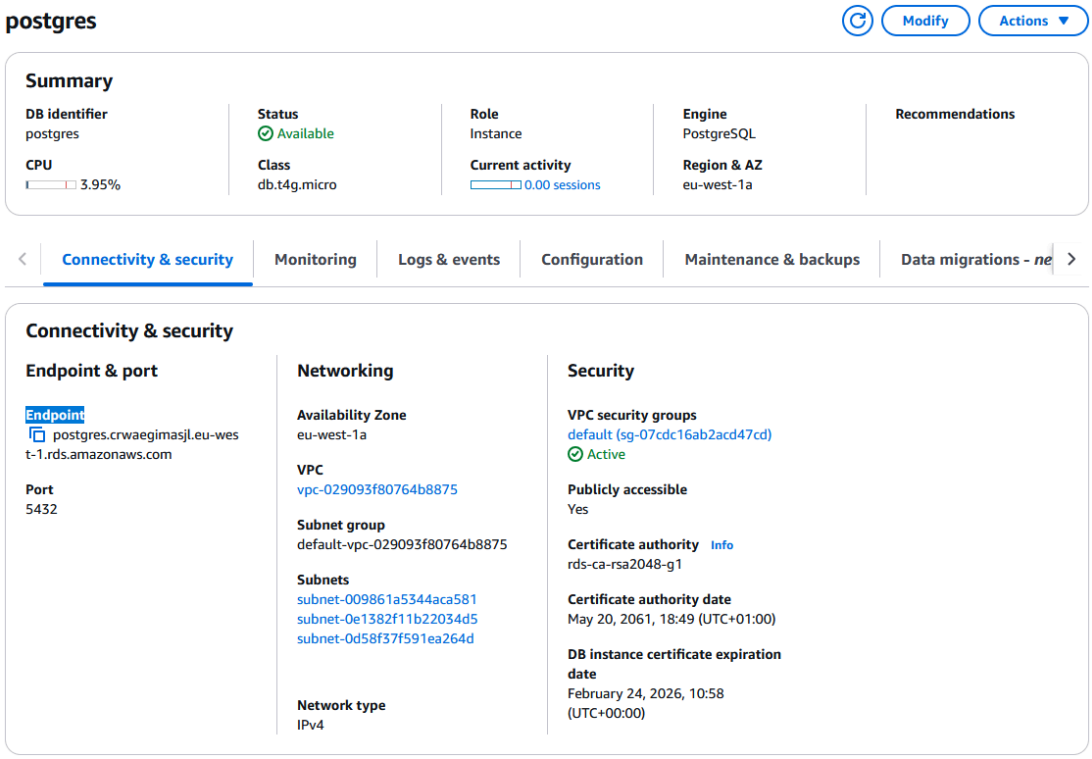
```
jakeh@LAPTOP-OO7KBC1I MINGW64 ~/Downloads
$ ssh -i "key-seneca.pem" ubuntu@ec2-3-250-35-61.eu-west-1.compute.amazonaws.com
The authenticity of host 'ec2-3-250-35-61.eu-west-1.compute.amazonaws.com (3.250.35.61)' can't be established.
ED25519 key fingerprint is SHA256:9IOjhHRQLkTTJxXfLVGOpYqcSOaHiaahvyRzl8s6kfI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'ec2-3-250-35-61.eu-west-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1021-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Mon Feb 24 13:40:37 UTC 2025

  System load:  0.0               Processes:             104
  Usage of /:   24.9% of 6.71GB   Users logged in:       0
  Memory usage: 20%               IPv4 address for enX0: 172.31.39.44
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-39-44:~$
```

9. Let's update our Ubuntu with the latest packages
    a. sudo apt update
    b. sudo apt upgrade
10. Let's install node
    a. curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
    b. sudo apt-get install -y nodejs
11. Let's clone the project git repo
    a. git clone https://github.com/JakeHornerMan/Seneca_BETT.git
12. Let's run NPM
    a. cd Seneca_BETT
    b. npm install
    c. sudo npm install -g nodemon
13. Create .env
    a. nano .env
    b. Input

        i. PORT=3000
        ii. JWT_SECRET_KEY=secret

iii. DB_HOST={YOUR DB HOST CREATED}
iv. DB_PORT=5432
v. DB_USERNAME=postgres
vi. DB_PASSWORD=password
vii. DB_DATABASE=postgres
viii. DB_LOGGING=false

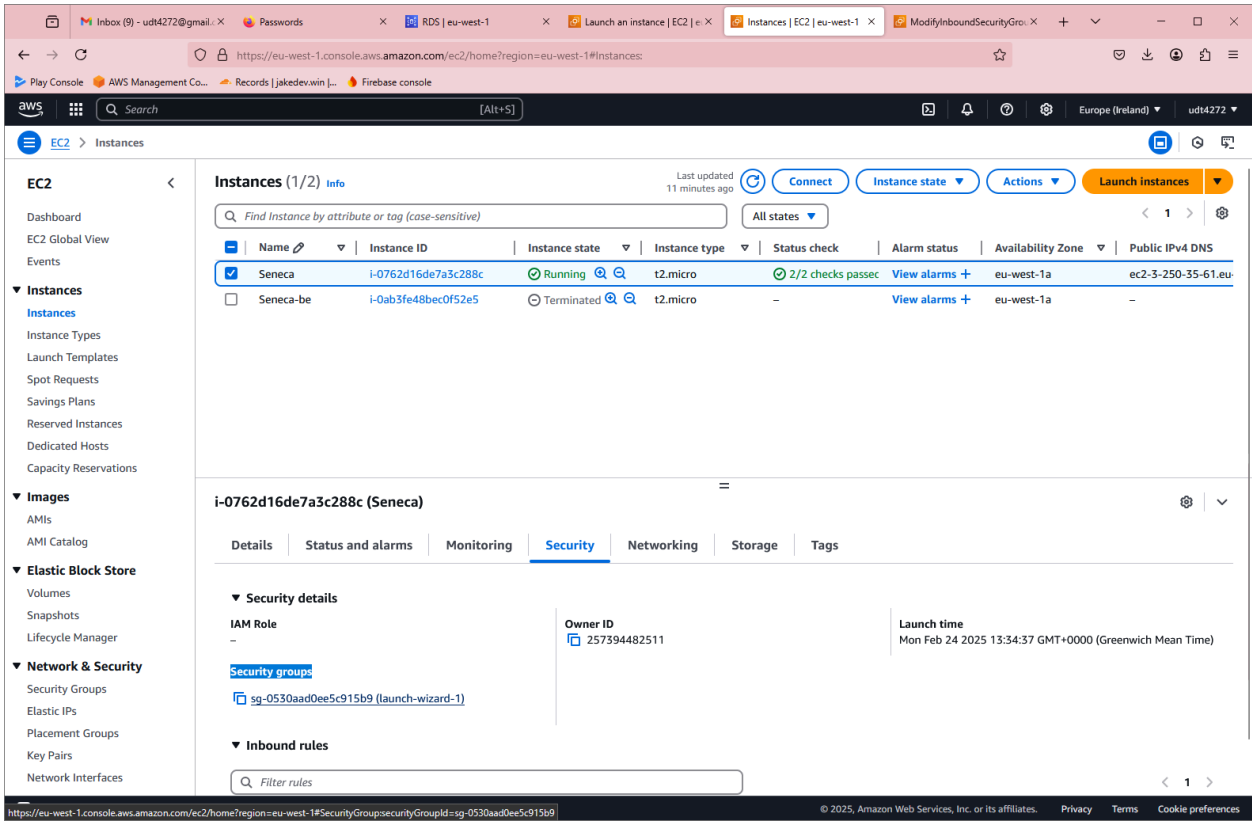
c. See Step 10 of Database deployment for the variables you created for the database.



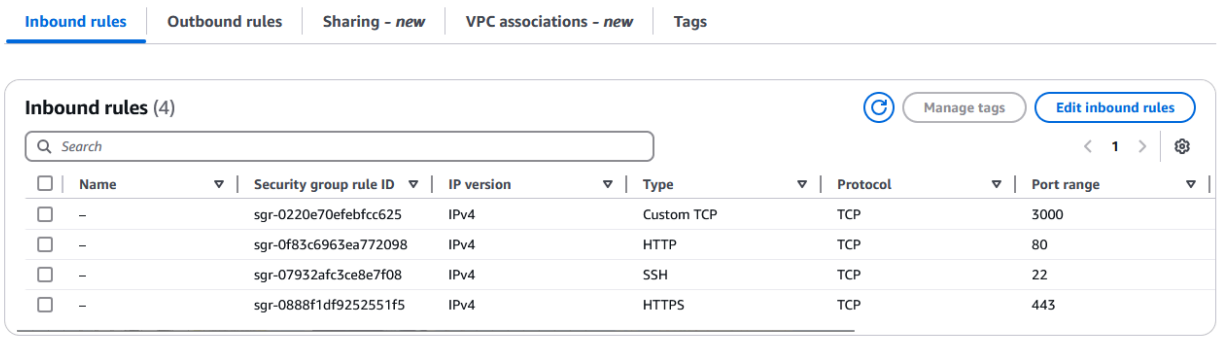
d. CTRL + X, then press Y, and hit Enter to save.

14. Run the application with npm run prod, we need to create the tables in the Database, and then Stop the program with Ctrl+C, once you see:
    a. Connected to DB host {DB_HOST}
    b. Server is running on port 3000

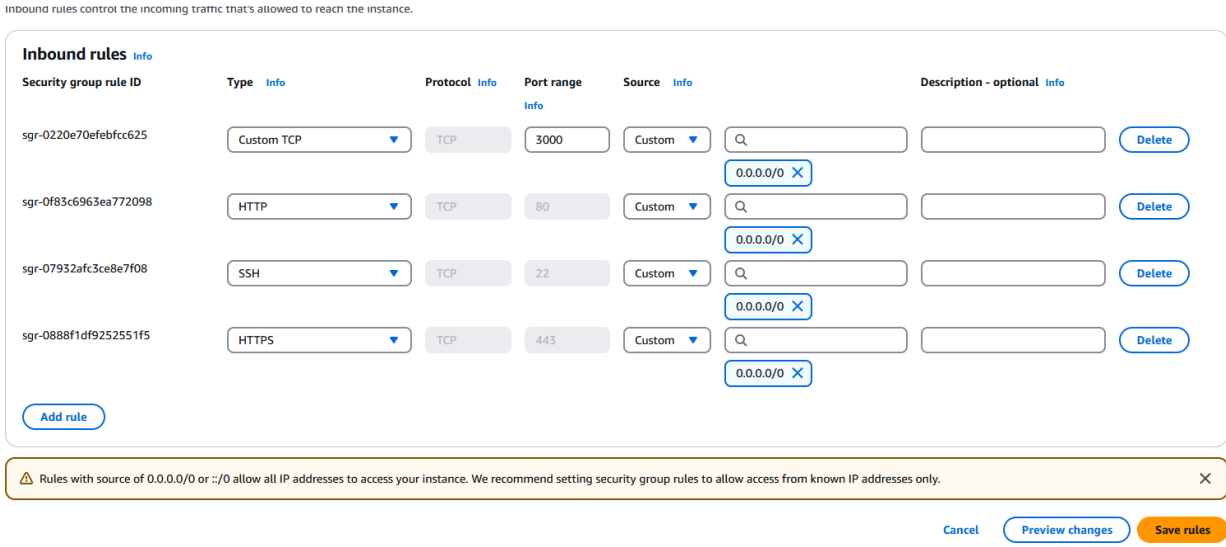
15. We need to make one change to the security group of our EC2 instance



a. Select Edit inbound rule



b. Add rule - Type: Custom TCP | Protocol: TCP | Port: 3000 | Source: 0.0.0.0/0



c. Select 'Save Rules'

16. Return to the EC2 instance and run the application.npm run prod

Populate Database

Within the project, there are 3 SQL files in the testData folder.
https://github.com/JakeHornerMan/Seneca_BETT/tree/main/src/testData
Please follow database deployment step 10 and input:

1. user.sql
2. Courses.sql
3. sessionModules.sql

In this order, we can set up the primary key and foreign key relationship