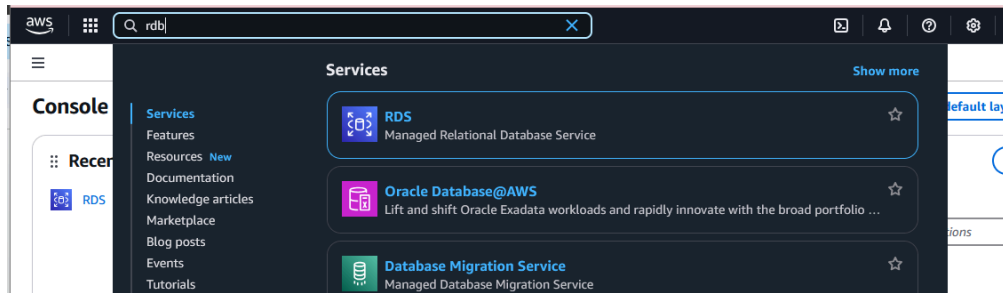


Seneca backend technical - Deployment Task

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'

The screenshot shows the AWS Management Console for an Amazon RDS PostgreSQL instance. The instance is named 'postgres' and is in the 'Active' state. The 'Connectivity & security' tab is selected, showing the following details:




- Endpoint & port:** Endpoint is `postgres.crwaegimasjl.eu-west-1.rds.amazonaws.com`, Port is `5432`.
- Networking:** Availability Zone is `eu-west-1a`, VPC is `vpc-029093f80764b8875`, Subnet group is `default-vpc-029093f80764b8875`, Subnets are `subnet-009861a5344aca581`, `subnet-0e1382f11b22034d5`, and `subnet-0d58f37f591ea264d`, Network type is `IPv4`.
- Security:** VPC security groups are `default (sg-07cdc16ab2acd47cd)` and `Active`, Publicly accessible is `Yes`, Certificate authority is `rds-ca-rsa2048-g1`, Certificate authority date is `May 20, 2061, 18:49 (UTC+01:00)`, DB instance certificate expiration date is `February 24, 2026, 10:58 (UTC+00:00)`.



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

The screenshot shows the 'Connectivity & security' tab for the PostgreSQL instance. The 'VPC security groups' link is highlighted, showing the following details:

- Endpoint & port:** Endpoint is `postgres.crwaegimasjl.eu-west-1.rds.amazonaws.com`, Port is `5432`.
- Networking:** Availability Zone is `eu-west-1a`, VPC is `vpc-029093f80764b8875`, Subnet group is `default-vpc-029093f80764b8875`, Subnets are `subnet-009861a5344aca581`, `subnet-0e1382f11b22034d5`, and `subnet-0d58f37f591ea264d`, Network type is `IPv4`.
- Security:** VPC security groups are `default (sg-07cdc16ab2acd47cd)` and `Active`, Publicly accessible is `Yes`, Certificate authority is `rds-ca-rsa2048-g1`, Certificate authority date is `May 20, 2061, 18:49 (UTC+01:00)`.

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

Security Groups (1/1) [Info](#)  **Actions**  **Export security groups to CSV**  **Create security group**

sg-07cdc16ab2acd47cd  **Clear filters** < 1 > 

<input checked="" type="checkbox"/>	Name	Security group ID	Security group name	VPC ID
<input checked="" type="checkbox"/>	-	sg-07cdc16ab2acd47cd	default	vpc-029093f80764b8875 [?]

sg-07cdc16ab2acd47cd - default

Details **Inbound rules** **Outbound rules** **Sharing - new** **VPC associations - new** **Tags**

Inbound rules (1)  **Manage tags** **Edit inbound rules**



<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	
<input type="checkbox"/>	-	sgr-0ea1ac5d7a23642ae	-	All traffic	All	AL

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


Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [Info](#)

Security group rule ID	Type	Protocol	Port range	Source	Description - optional	
sgr-073999a21fe5a8bf6	PostgreSQL	TCP	5432	Custom	<input type="text" value="0.0.0.0"/>	 Delete
sgr-0d5a5957191d4ea52	Custom TCP	TCP	0	Custom	<input type="text" value="172.31.37.208"/>	 Delete

Add rule

 Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel **Preview changes** **Save rules**

10. Now Select Outbound Rules

Inbound rules

Outbound rules

Sharing - *new*

VPC associations - *new*

Tags

Outbound rules (2)

Manage tags

Edit outbound rules

Q Search

<

1

>

⚙

<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range	Dest
<input type="checkbox"/>	-	sgr-018ad4925bfbdb235b	IPv4	PostgreSQL	TCP	5432	0.0.0.0/0
<input type="checkbox"/>	-	sgr-0e5292e7af507de0c	IPv4	All traffic	All	All	0.0.0.0/0

11. Input the following:

Edit outbound rules

Outbound rules control the outgoing traffic that's allowed to leave the instance.

Outbound rules

Security group rule ID	Type	Protocol	Port range	Destination	Description - optional
sgr-018ad4925bfd235b	PostgreSQL	TCP	5432	Custom Q 0.0.0.0/0	
sgr-0e5292e7af507de0c	All traffic	All	All	Custom Q 0.0.0.0/0	

Add rule

Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IP v4 or IP v6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.

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

- a. Object > Register > Server

The screenshot shows a 'Register - Server' dialog box with the following fields and controls:

- Name:** A text input field containing 'seneca-db'.
- Server group:** A dropdown menu showing 'Servers'.
- Background:** A checkbox that is disabled (indicated by an 'X' icon).
- Foreground:** A checkbox that is disabled (indicated by an 'X' icon).
- Connect now?:** A toggle switch that is turned on.
- Comments:** A large text area for entering comments.

At the bottom of the dialog, there are three buttons: 'Close', 'Reset', and 'Save'.

- 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:

postgres

[Modify](#)[Actions](#) ▼

Summary

DB identifier
postgres

CPU
 3.95%

Status
Available

Class
db.t4g.micro

Role
Instance

Current activity
 0.00 sessions

Engine
PostgreSQL

Region & AZ
eu-west-1a

Recommendations

[Connectivity & security](#)[Monitoring](#)[Logs & events](#)[Configuration](#)[Maintenance & backups](#)[Data migrations - ne](#) >

Connectivity & security

Endpoint & port

Endpoint

☐ postgres.crwaegimasjl.eu-west-1.rds.amazonaws.com

Port
5432

Networking

Availability Zone
eu-west-1a

VPC
vpc-029093f80764b8875

Subnet group
default-vpc-029093f80764b8875

Subnets
subnet-009861a5344aca581
subnet-0e1382f11b22034d5
subnet-0d58f37f591ea264d

Network type
IPv4

Security

VPC security groups
default (sg-07cdc16ab2acd47cd)
Active

Publicly accessible
Yes

Certificate authority [Info](#)
rds-ca-rsa2048-g1

Certificate authority date
May 20, 2061, 18:49 (UTC+01:00)

DB instance certificate expiration date
February 24, 2026, 10:58 (UTC+00:00)

Register - Server

[General](#)[Connection](#)[Parameters](#)[SSH Tunnel](#)[Advanced](#)[Tags](#)

Host name/address

Port

Maintenance database

Username

Kerberos authentication?



Password

Save password?



Role

Service

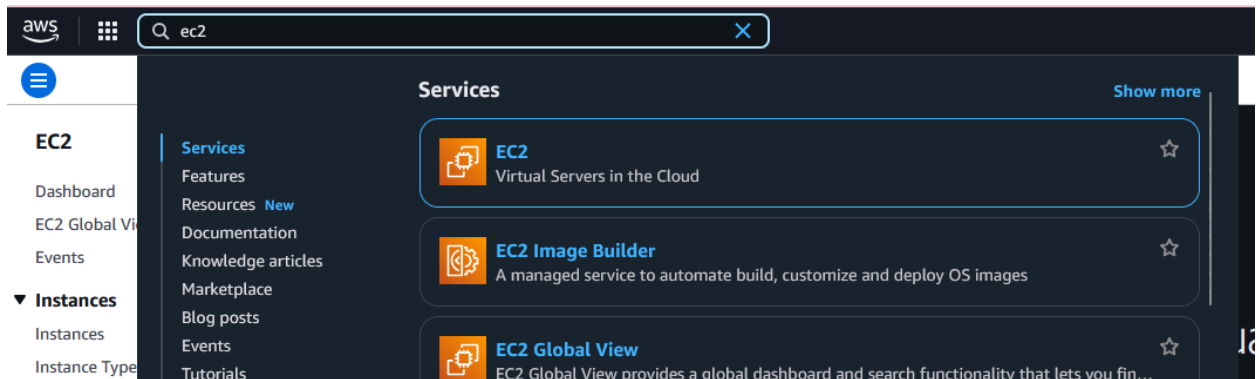
[Close](#)[Reset](#)[Save](#)

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

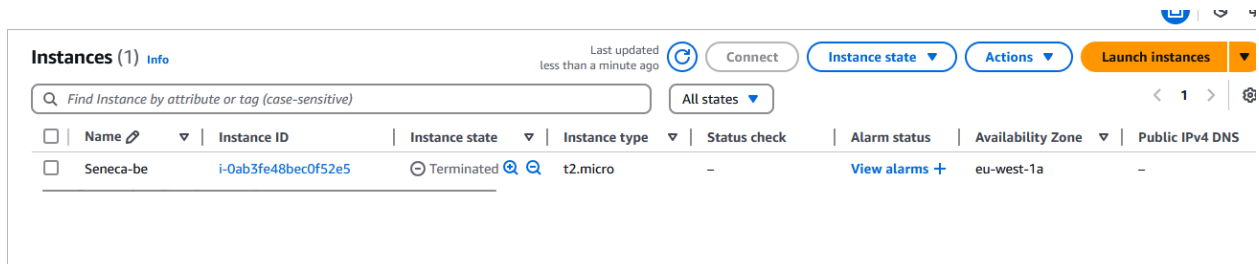
- i. Port - 5432 (Connectivity>Additional configuration>Database port)
- ii. Maintenance database - postgres (Settings>DB instance identifier)
- iii. Username - postgres (Settings>Master username)
- iv. Password - password (Settings>Master password/ Confirm master password)

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'

 **Success**
Successfully initiated launch of instance [\(i-0762d16de7a3c288c\)](#)

► Launch log

Next Steps

Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
[Create billing alerts](#)

Connect to your instance
Once your instance is running, log into it from your local computer.
[Connect to instance](#)
[Learn more](#)

Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
[Connect an RDS database](#)
[Create a new RDS database](#)
[Learn more](#)

Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
[Create EBS snapshot policy](#)

Manage detailed monitoring
Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period.

Create Load Balancer
Create an application, network gateway or classic Elastic Load Balancer
[Create Load Balancer](#)

Create AWS budget
AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location.


Manage CloudWatch alarms
Create or update Amazon CloudWatch alarms for the instance.
[Manage CloudWatch alarms](#)



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


Connect to instance [Info](#)


Connect to your instance [i-0762d16de7a3c288c](#) (Seneca) using any of these options

EC2 Instance Connect | **Session Manager** | **SSH client** | **EC2 serial console**

Instance ID
 [i-0762d16de7a3c288c](#) (Seneca)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is `key-seneca.pem`
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 `chmod 400 "key-seneca.pem"`
4. Connect to your instance using its Public DNS:
 `ec2-3-250-35-61.eu-west-1.compute.amazonaws.com`

Example:
 `ssh -i "key-seneca.pem" ubuntu@ec2-3-250-35-61.eu-west-1.compute.amazonaws.com`

 **Note:** In most cases, the guessed username is correct. However, read your AMI usage instructions to che

8. We are now connected to our instance

```
jakeh@LAPTOP-007KBC1I 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 SHA256:9IOjhHRQLkTTJxXFLVGOpYqcSOaHiaahvyRz18s6kFI.
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
- sudo apt update
 - sudo apt upgrade
10. Let's install node
- curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
 - sudo apt-get install -y nodejs
11. Let's clone the project git repo
- git clone https://github.com/JakeHornerMan/Seneca_BETT.git
12. Let's run NPM
- cd Seneca_BETT
 - npm install
 - sudo npm install -g nodemon
13. Create .env
- nano .env
 - Input

- PORT=3000
- 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.

postgres 🔄 Modify Actions ▾

Summary				
DB identifier postgres	Status 🟢 Available	Role Instance	Engine PostgreSQL	Recommendations
CPU <div><div></div> 3.95%</div>	Class db.t4g.micro	Current activity <div><div></div> 0.00 sessions</div>	Region & AZ eu-west-1a	

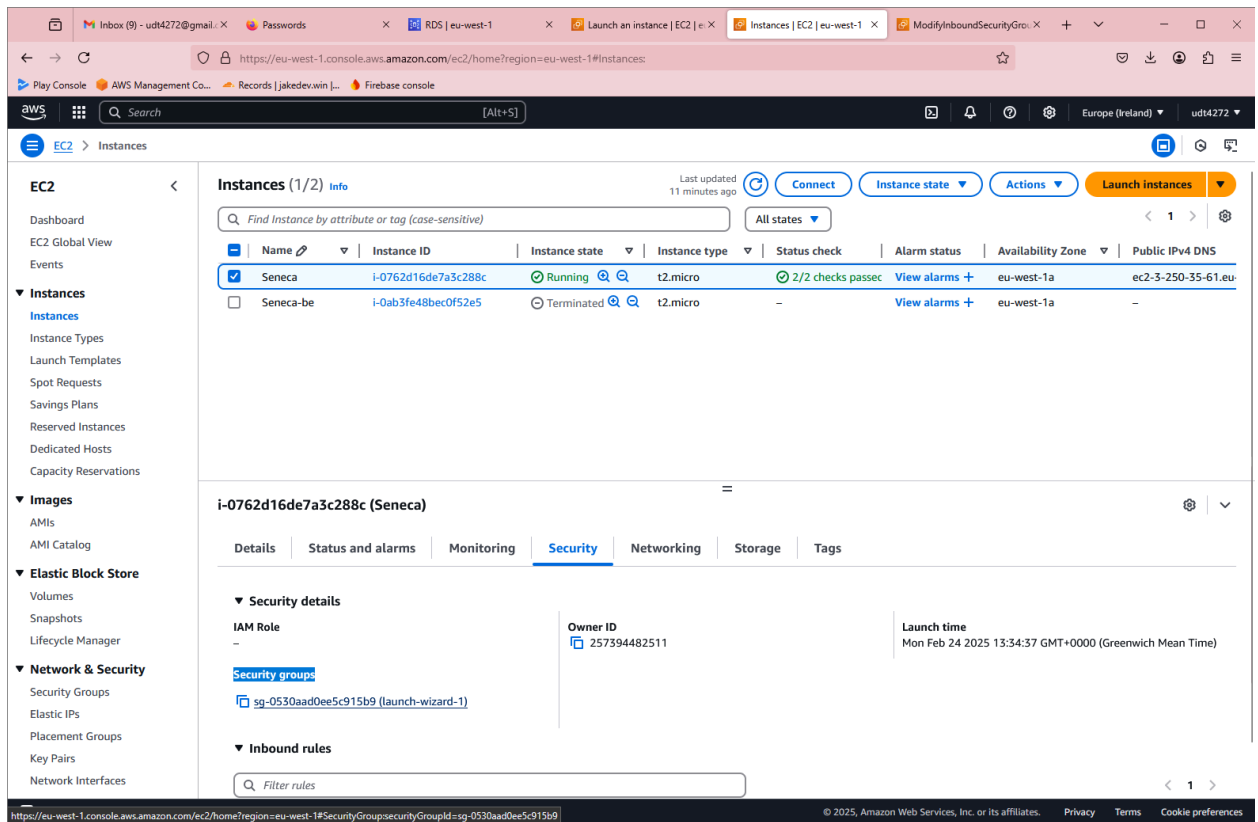
<
Connectivity & security
Monitoring
Logs & events
Configuration
Maintenance & backups
Data migrations - ne
>

Connectivity & security

Endpoint & port	Networking	Security
Endpoint postgres.crwaegimasjl.eu-west-1.rds.amazonaws.com	Availability Zone eu-west-1a VPC vpc-029093f80764b8875 Subnet group default-vpc-029093f80764b8875 Subnets subnet-009861a5344aca581 subnet-0e1382f11b22034d5 subnet-0d58f37f591ea264d	VPC security groups default (sg-07cdc16ab2acd47cd) 🟢 Active Publicly accessible Yes Certificate authority Info rds-ca-rsa2048-g1 Certificate authority date May 20, 2061, 18:49 (UTC+01:00) DB instance certificate expiration date February 24, 2026, 10:58 (UTC+00:00)

- 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



The screenshot shows the AWS Management Console interface. The left sidebar contains navigation links for EC2, including Dashboard, EC2 Global View, Events, Instances, Images, Elastic Block Store, and Network & Security. The main content area displays the 'Instances (1/2)' page. A table lists two instances: 'Seneca' (i-0762d16de7a3c288c) in a 'Running' state, and 'Seneca-be' (i-0ab3fe48bec0f52e5) in a 'Terminated' state. The 'Seneca' instance is selected, and its details are shown below. The 'Security' tab is active, displaying the 'sg-0530aad0ee5c915b9' security group. The 'Inbound rules' section shows four rules: Custom TCP (3000), HTTP (80), SSH (22), and HTTPS (443).

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Seneca	i-0762d16de7a3c288c	Running	t2.micro	2/2 checks passed	View alarms +	eu-west-1a	ec2-3-250-35-61.eu
Seneca-be	i-0ab3fe48bec0f52e5	Terminated	t2.micro	-	View alarms +	eu-west-1a	-

i-0762d16de7a3c288c (Seneca)

Security details

IAM Role: -

Owner ID: 257394482511

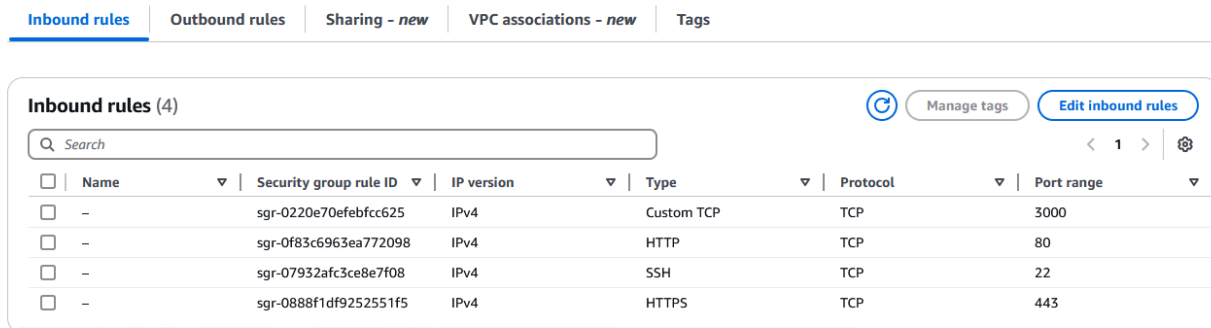
Launch time: Mon Feb 24 2025 13:34:37 GMT+0000 (Greenwich Mean Time)

Security groups: sg-0530aad0ee5c915b9 (launch-wizard-1)

Inbound rules

Name	Security group rule ID	IP version	Type	Protocol	Port range
-	sgr-0220e70efebfcc625	IPv4	Custom TCP	TCP	3000
-	sgr-0f83c6963ea772098	IPv4	HTTP	TCP	80
-	sgr-07932afc3ce8e7f08	IPv4	SSH	TCP	22
-	sgr-0888f1df9252551f5	IPv4	HTTPS	TCP	443

a. Select Edit inbound rule



The screenshot shows the 'Inbound rules' tab for the 'sg-0530aad0ee5c915b9' security group. The 'Edit inbound rules' button is highlighted. The table shows four inbound rules: Custom TCP (3000), HTTP (80), SSH (22), and HTTPS (443).

Name	Security group rule ID	IP version	Type	Protocol	Port range
-	sgr-0220e70efebfcc625	IPv4	Custom TCP	TCP	3000
-	sgr-0f83c6963ea772098	IPv4	HTTP	TCP	80
-	sgr-07932afc3ce8e7f08	IPv4	SSH	TCP	22
-	sgr-0888f1df9252551f5	IPv4	HTTPS	TCP	443

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

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>		
sgr-0220e70efebfcc625	Custom TCP ▼	TCP	3000	Custom ▼	Q	<input type="text"/>	Delete
sgr-0f83c6963ea772098	HTTP ▼	TCP	80	Custom ▼	0.0.0.0/0 ✕	<input type="text"/>	Delete
sgr-07932afc3ce8e7f08	SSH ▼	TCP	22	Custom ▼	0.0.0.0/0 ✕	<input type="text"/>	Delete
sgr-0888f1df9252551f5	HTTPS ▼	TCP	443	Custom ▼	0.0.0.0/0 ✕	<input type="text"/>	Delete
					0.0.0.0/0 ✕		

Add rule

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

CancelPreview changesSave rules

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