AWS VPC with EC2 and RDS Integration: A Secure and Scalable Network Architecture

Pella Joshua Naphtali
joshuapellaa@gmail.com

Overview

This project demonstrates the creation of a secure and scalable network architecture on AWS, leveraging multiple services to simulate a real-world cloud environment. It involves setting up a custom Virtual Private Cloud (VPC) with public and private subnets, deploying an EC2 instance for compute resources, and integrating a managed RDS database for data storage.

The architecture emphasizes security and best practices by isolating resources within subnets, restricting direct internet access to sensitive components, and using security groups for fine-grained traffic control. CloudWatch monitoring is integrated to track performance and enable proactive issue resolution.

Services used:

1. VPC
2. EC2
3. RDS
4. CloudWatch

Custom VPC Setup

A **Virtual Private Cloud (VPC)** is a logically isolated section of the AWS cloud where you can launch and manage AWS resources like **EC2**, **RDS**, etc., in a highly configurable network environment.

AWS provides a **default VPC** for quick setups. However, a **custom VPC** gives you better control over network architecture and security.

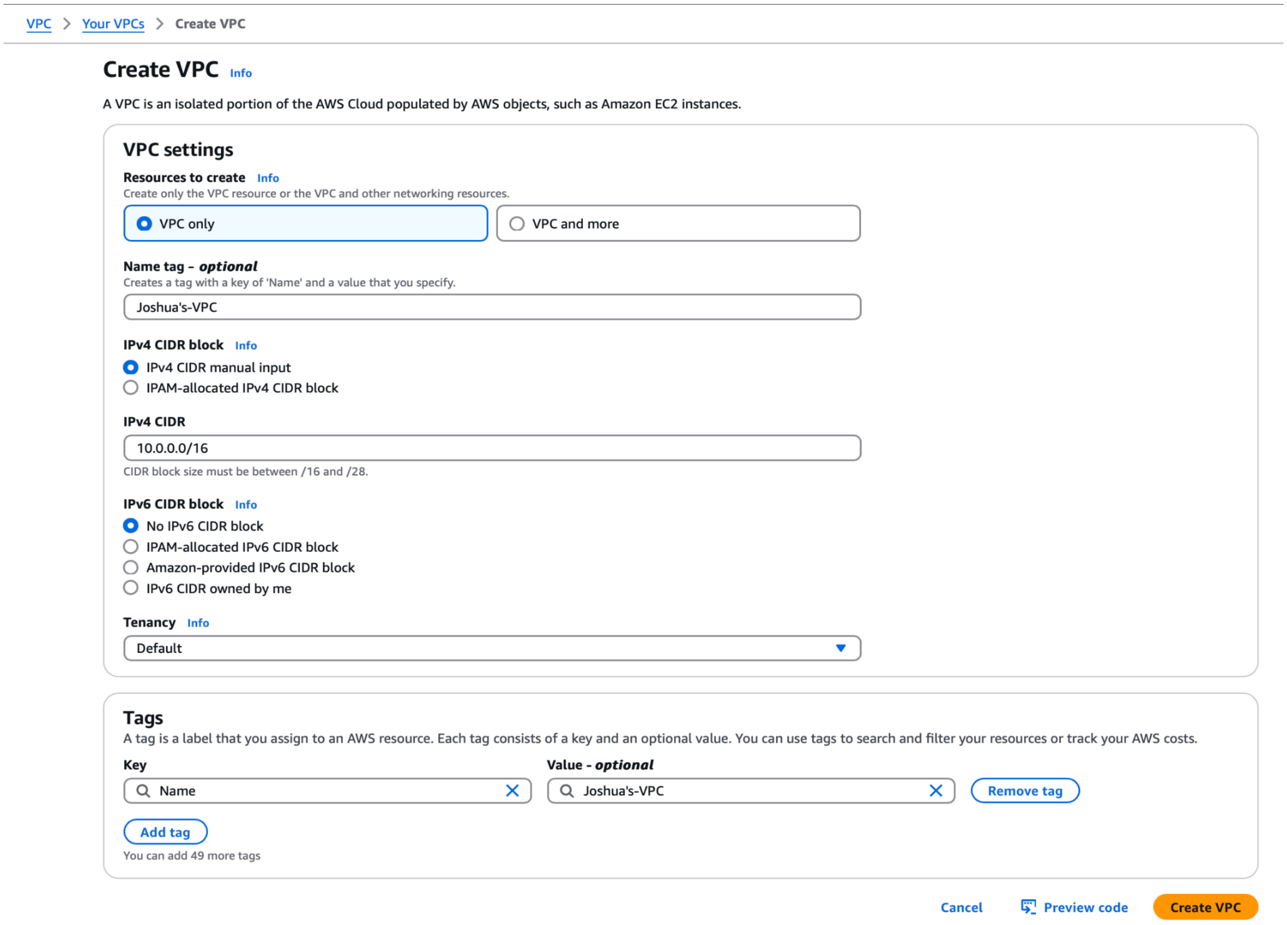
Steps for creating a custom VPC:

1. After logging in to the AWS management console, navigate to the VPC dashboard and select create VPC.
2. Assign a name to your VPC and give it a CIDR (Classless Inter-Domain Routing) block of 10.0.0.0/16 (this gives you 65,536 IPs which you can divide into subnets and your instances pick out a private IP address from that range).

   Note: You can select a different CIDR range e.g. 172.16.0.0/16. Just ensure you select a large enough range.

3. Select no IPv6 block and select default tenancy.
4. Review your settings then click create VPC.

Your VPC setup wizard should look like the image below:



VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
● VPC only ○ VPC and more

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.
Joshua's-VPC

IPv4 CIDR block Info
● IPv4 CIDR manual input
○ IPAM-allocated IPv4 CIDR block

IPv4 CIDR
10.0.0.0/16
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
● No IPv6 CIDR block
○ IPAM-allocated IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block
○ IPv6 CIDR owned by me

Tenancy Info
Default

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key: Name
Value - *optional*: Joshua's-VPC
Remove tag

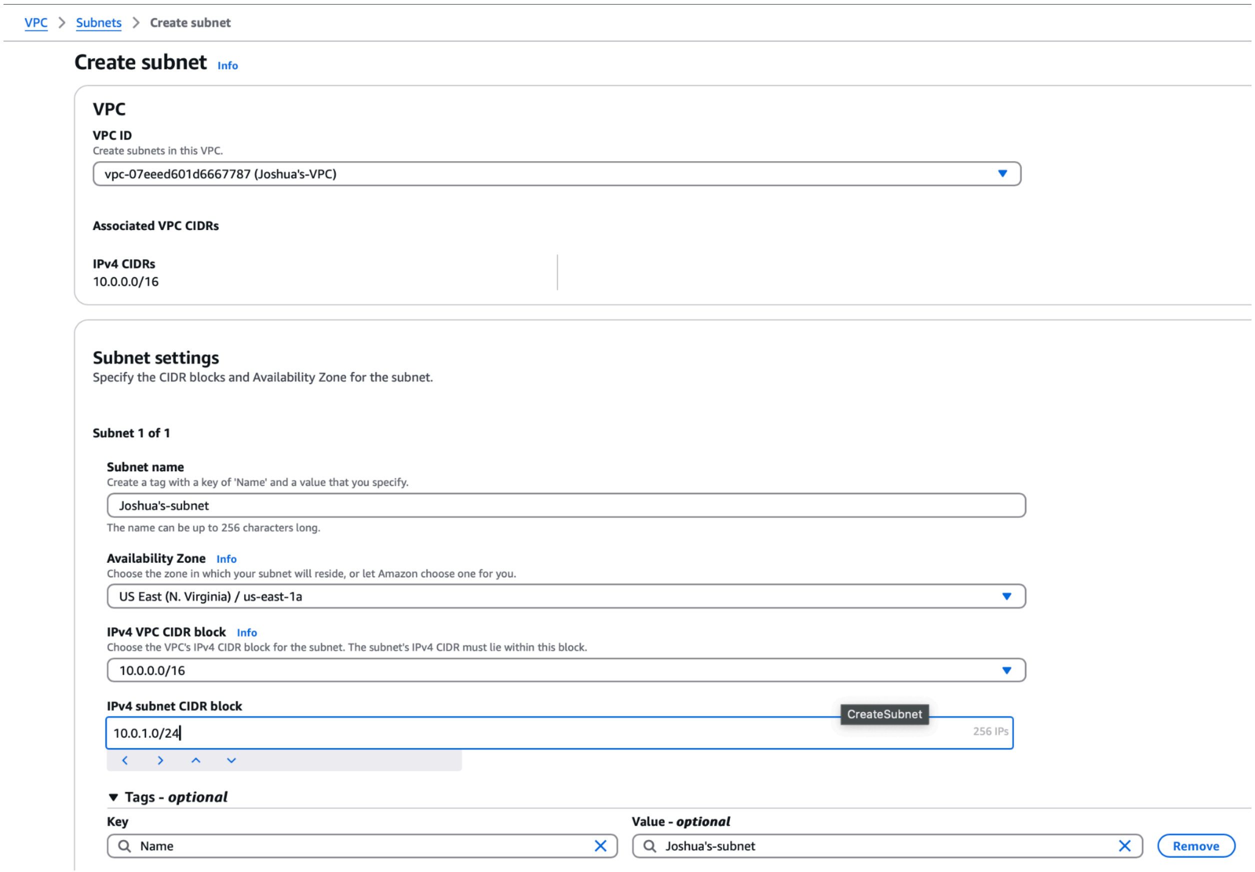
Add tag
You can add 49 more tags

Next up, we create subnets within our VPC

1. In the VPC dashboard, select Subnets then Create Subnet.
2. Under VPC ID, select the VPC you created in the previous step.
3. Assign a name and an availability zone e.g.us-east-1a.
4. Assign a CIDR block to the subnet within the range of the VPC’s CIDR block e.g. 10.0.1.0/24 (this gives you 256 IPs).
5. Create the VPC and repeat the steps again with a subnet CIDR of 10.0.2.0/24

We created two VPC’s because one is public and the other is private. It would be explained later on.

Our subnets should look like so:



VPC > Subnets > Create subnet

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-07eeed601d6667787 (Joshua's-VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Joshua's-subnet
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a

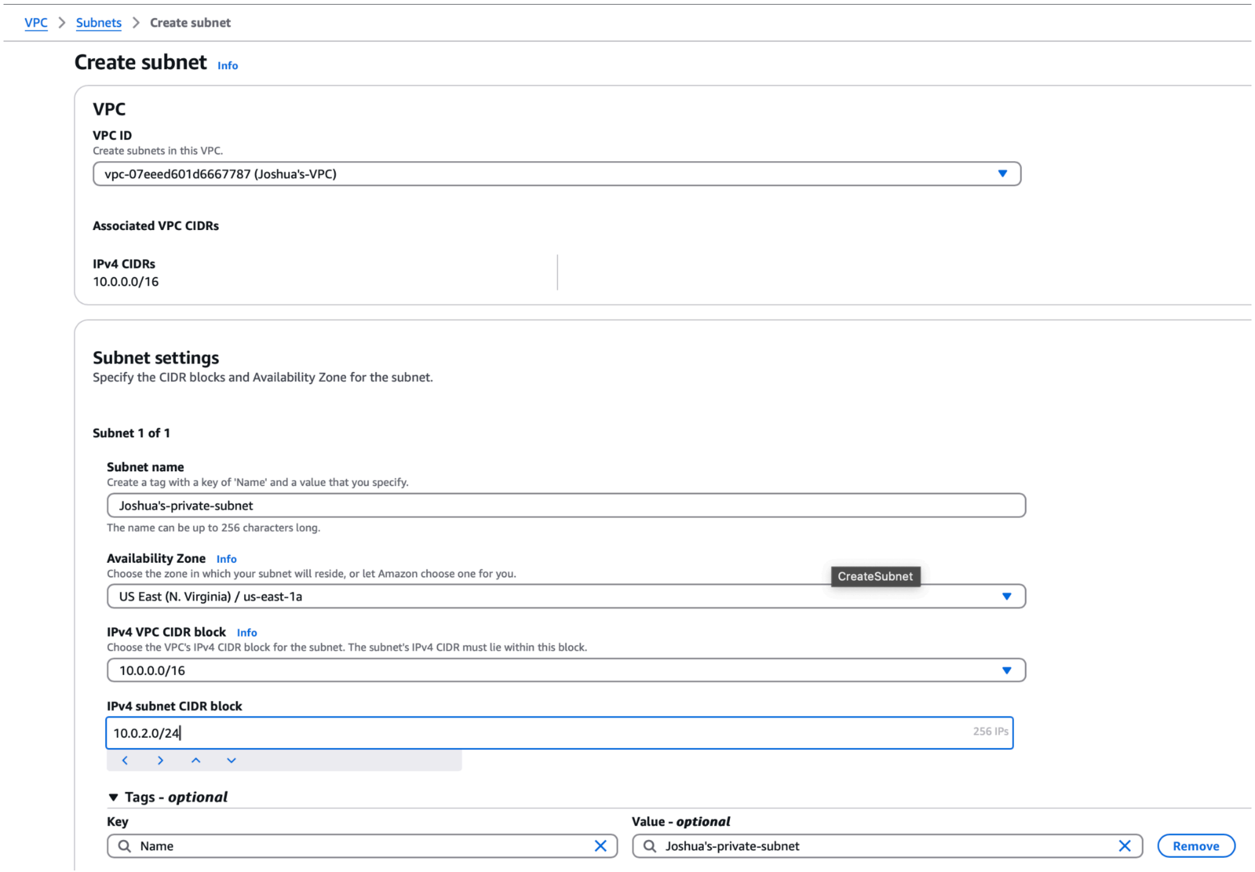
IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24 256 IPs

CreateSubnet

▼ Tags - *optional*

Key	Value - *optional*	
Name	Joshua's-subnet	Remove



VPC > Subnets > Create subnet

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-07eeed601d6667787 (Joshua's-VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Joshua's-private-subnet
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.2.0/24 256 IPs

Tags - *optional*

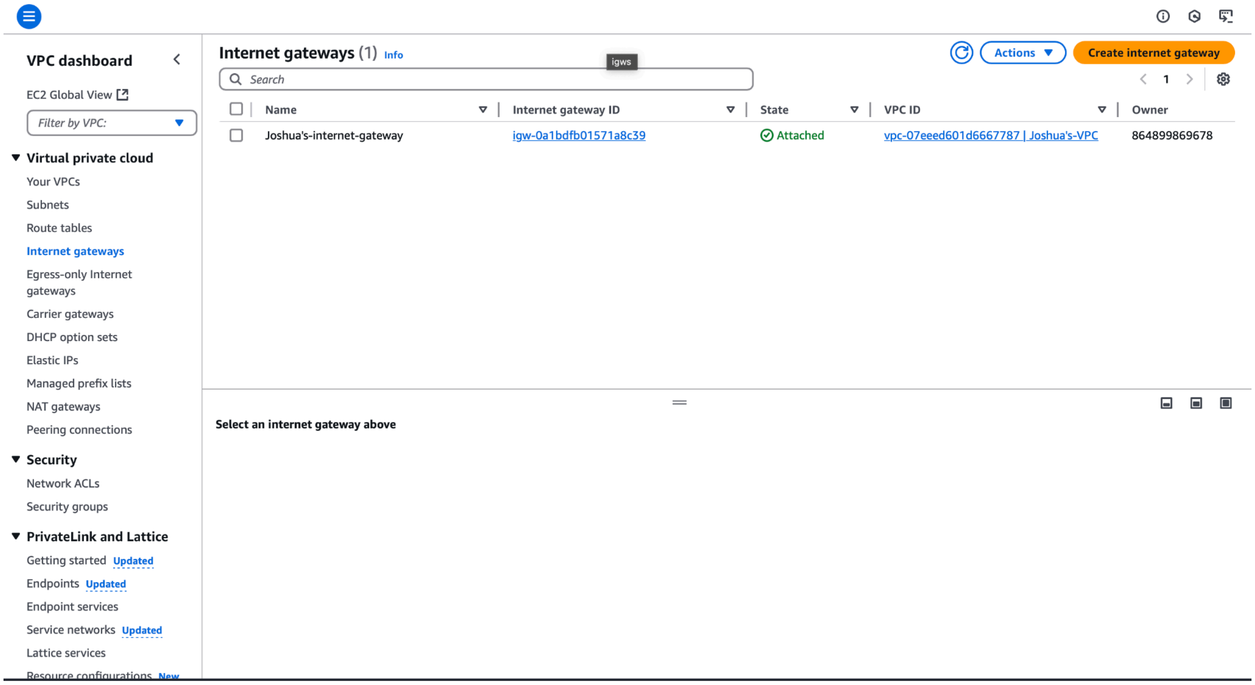
Key	Value - *optional*
Name	Joshua's-private-subnet

After creating our subnets, we create an internet gateway. An internet gateway allows instances in a VPC to communicate with the internet. The default VPC usually comes with an internet gateway attached to it already, but when you create a custom VPC, you also need to create and attach an internet gateway to that VPC.

The steps for creating and attaching are:

1. Go to internet gateways in the VPC dashboard and click on create internet gateway.
2. Give it a name and click create.
3. Select the internet gateway you selected, click actions and click attach to VPC.
4. Choose the VPC you created and attach the internet gateway to it.

It should look like the image below:



Internet gateways (1) Info

☐	Name	Internet gateway ID	State	VPC ID	Owner
☐	Joshua's-internet-gateway	igw-0a1bdfb01571a8c39	✓ Attached	vpc-07eeed601d6667787 | Joshua's-VPC	864899869678

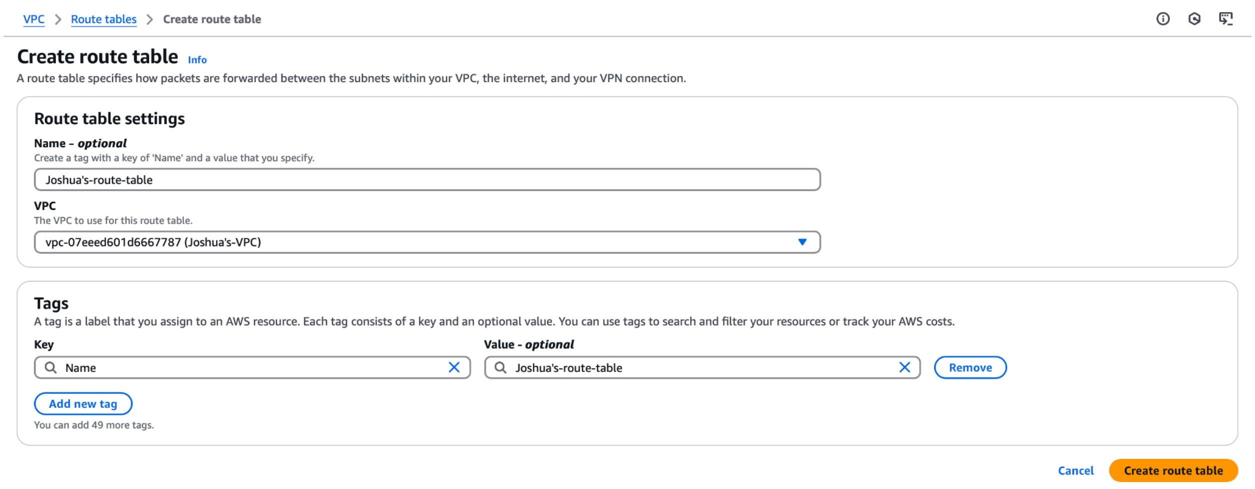
We can see that the internet gateway is attached to the VPC we created.

For traffic from instances in our VPC to flow to the internet gateway, the VPC needs to know the path or the route to the internet gateway and that is done with a route table.

A route table contains routes that determine where network traffic is directed within a VPC. As with internet gateways, the default VPC also has a default route table that contains a route to the default internet gateway.

To create a route table:

1. Navigate to route tables in the VPC dashboard and click create route table.
2. Give it a name and select the public VPC we created then click create route table.



Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
Joshua's-route-table

VPC
The VPC to use for this route table.
vpc-07eeed601d6667787 (Joshua's-VPC)

Tags

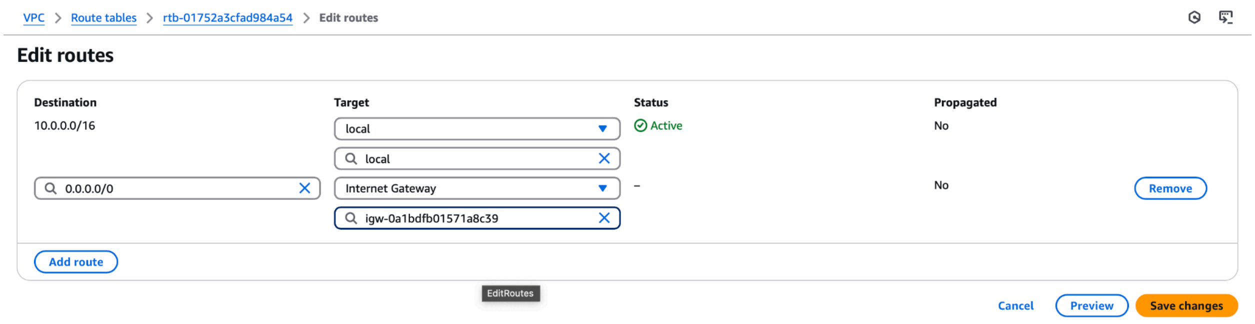
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	Joshua's-route-table

You can add 49 more tags.

Now we need to add a route to the internet gateway:

1. Select the route table we just created, click actions then click edit routes.
2. Click add routes, the destination should be 0.0.0.0/0 (this CIDR block encompasses the entire IPv4 address space).
3. The target should be an internet gateway then select the internet gateway we created.



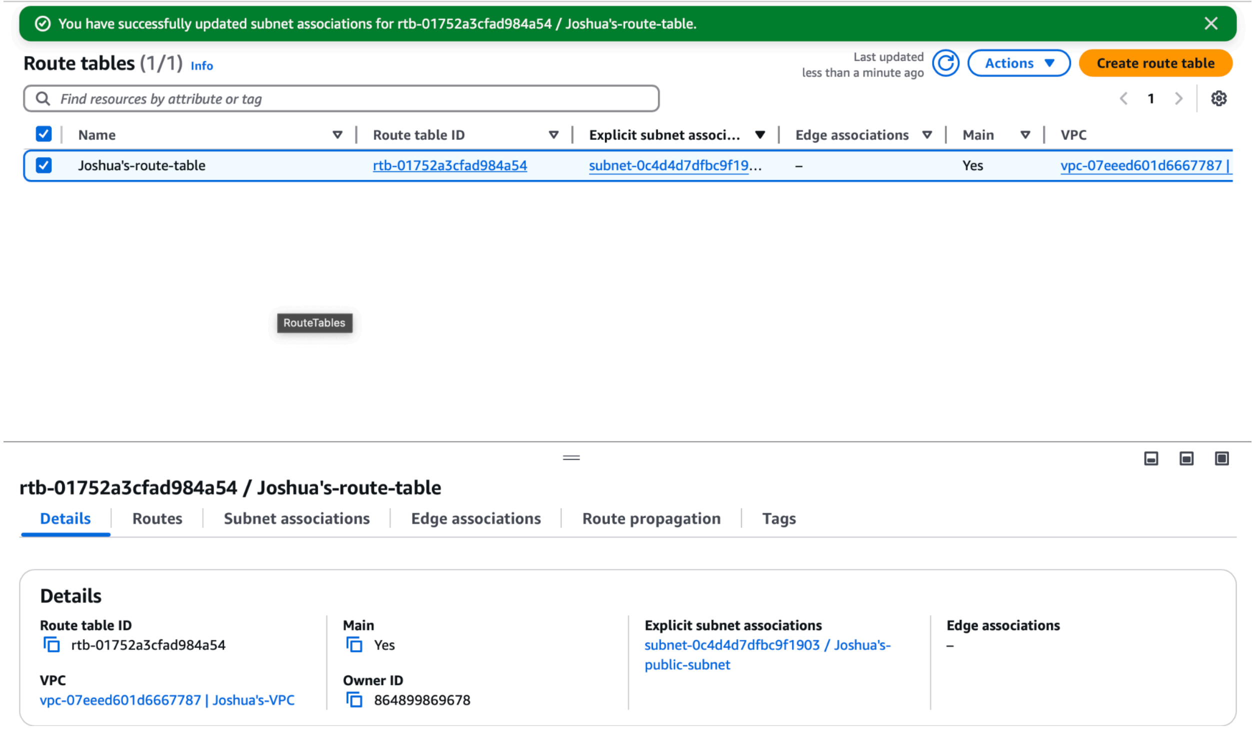
Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local local	Active	No
0.0.0.0/0	Internet Gateway igw-0a1bdfb01571a8c39	-	No

Finally, we attach the route table to a subnet:

1. Select the route table, click actions, and click edit subnet associations.
2. Select the public subnet we created and click save associations.

Note: Only a public subnet can have a route to an internet gateway, a subnet without a route to an internet gateway is a private subnet.



You have successfully updated subnet associations for rtb-01752a3cfad984a54 / Joshua's-route-table.

Route tables (1/1)

Info
Last updated less than a minute ago
Actions
Create route table
Find resources by attribute or tag
1
RouteTables

☑	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
☑	Joshua's-route-table	rtb-01752a3cfad984a54	subnet-0c4d4d7dfbc9f19...	–	Yes	vpc-07eeed601d6667787 |

rtb-01752a3cfad984a54 / Joshua's-route-table

Details

Route table ID: rtb-01752a3cfad984a54

VPC: vpc-07eeed601d6667787 | Joshua's-VPC

Main: Yes

Owner ID: 864899869678

Explicit subnet associations: subnet-0c4d4d7dfbc9f1903 / Joshua's-public-subnet

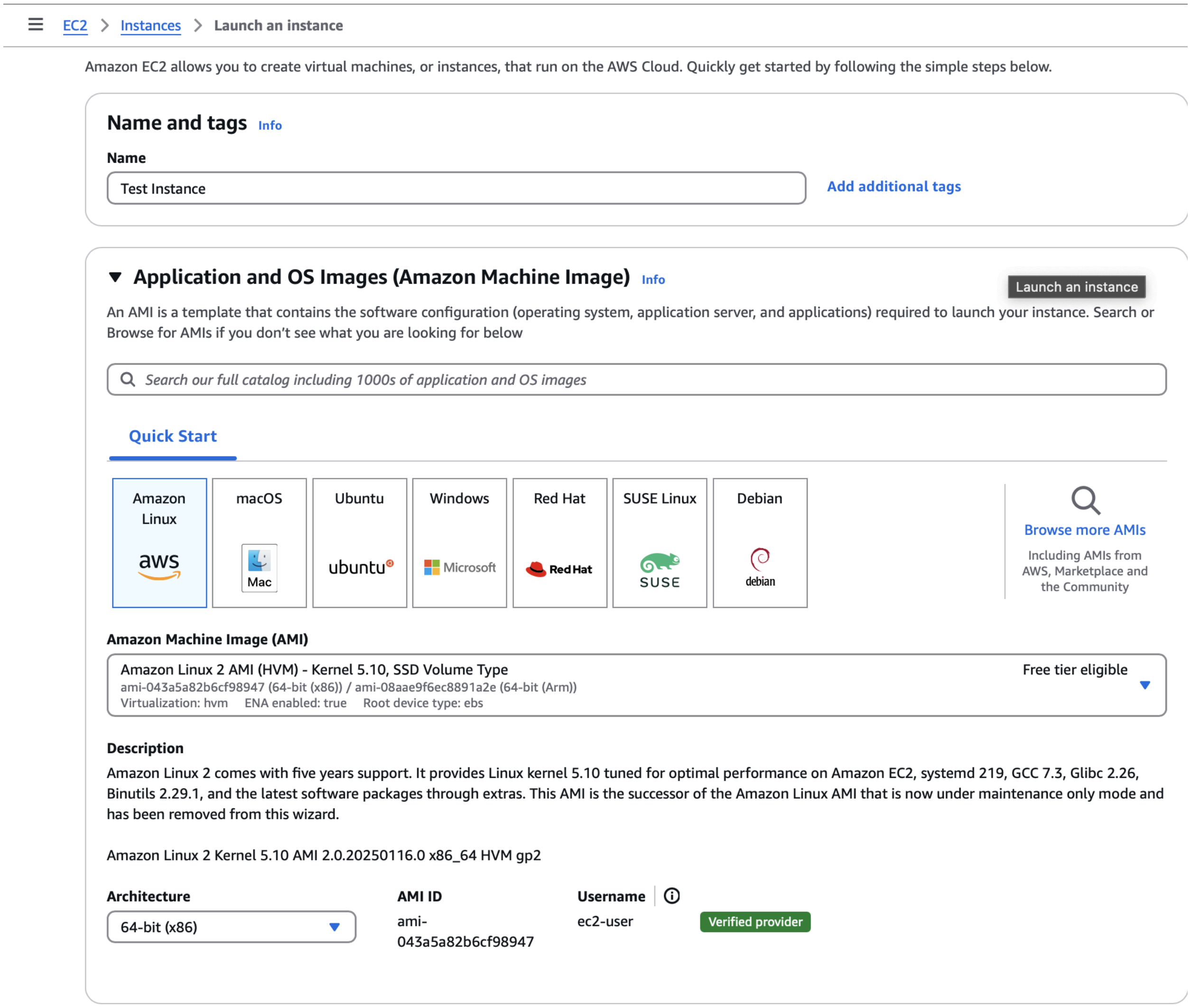
Edge associations: –

As we can see above, the route table is associated with our public subnet.

Instance Setup

Now that we are done configuring our VPC, let's create an EC2 instance to ensure everything works properly. To create an instance:

1. Navigate to the EC2 dashboard and click launch instance.
2. Name the instance, choose the Amazon Linux AMI (it is free tier eligible) and select the t2.micro instance type.



Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: Test Instance

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type — Free tier eligible
ami-043a5a82b6cf98947 (64-bit (x86)) / ami-08aae9f6ec8891a2e (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

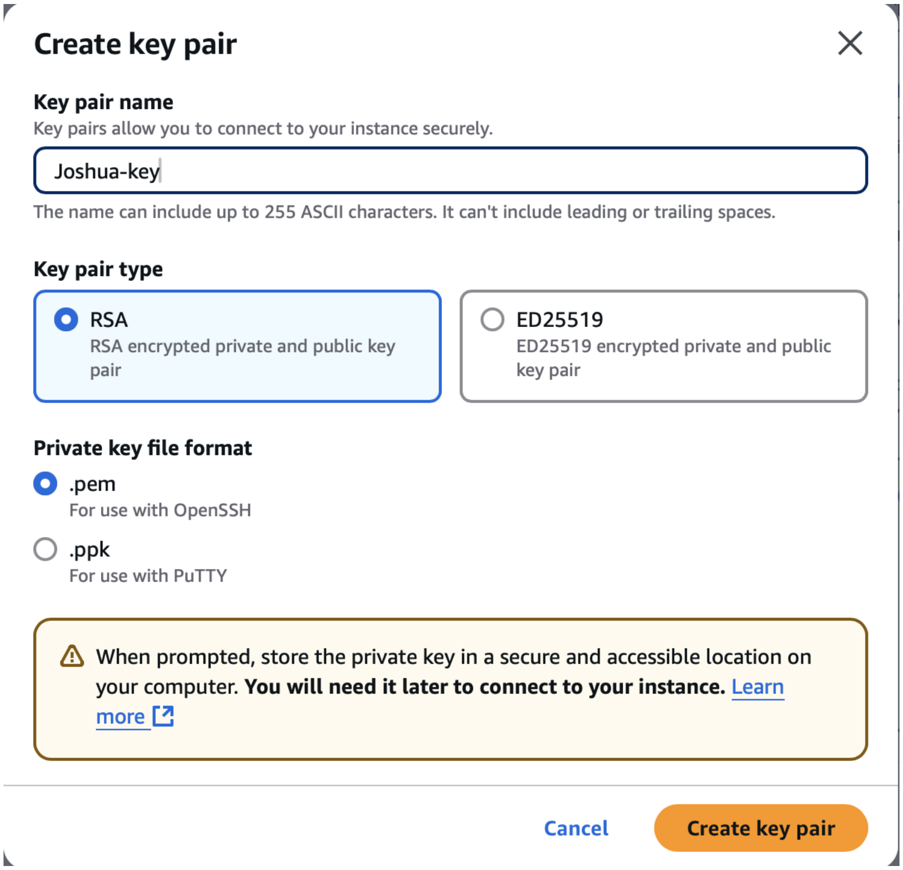
Description

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed from this wizard.

Amazon Linux 2 Kernel 5.10 AMI 2.0.20250116.0 x86_64 HVM gp2

Architecture	AMI ID	Username
64-bit (x86)	ami-043a5a82b6cf98947	ec2-user

3. Create a key pair to allow us to ssh into our instance.



Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

Joshua-key

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

● RSA
RSA encrypted private and public key pair

○ ED25519
ED25519 encrypted private and public key pair

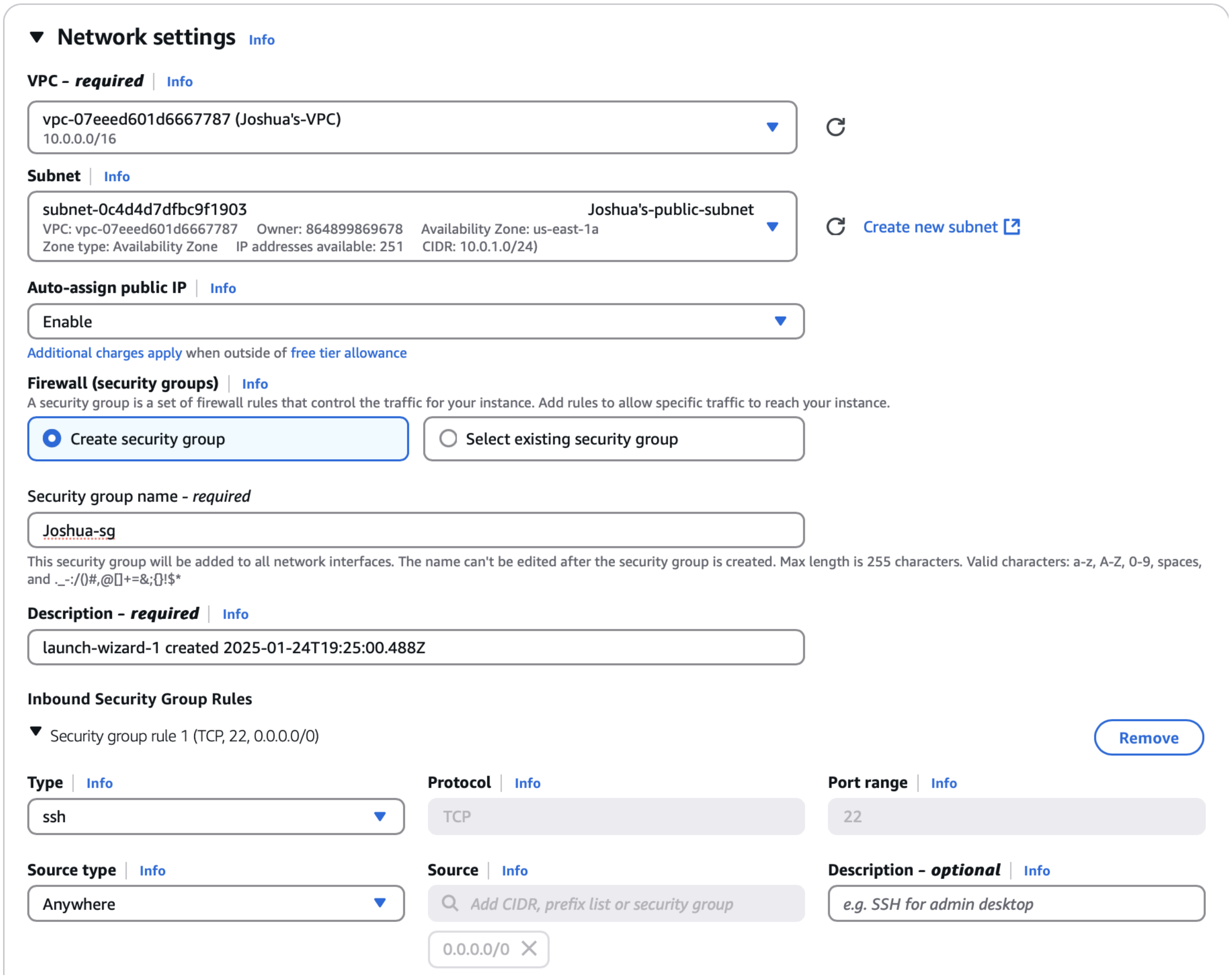
Private key file format

● .pem
For use with OpenSSH

○ .ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** Learn more

4. Select the VPC we created.
5. Select the public subnet we created.
6. Enable auto-assign public IP
7. We create a security group to define what traffic can flow through our instances. We define an inbound ssh rule and a rule that allows all traffic from all ports from any IP address.



▼ Network settings

VPC - *required*

vpc-07eeed601d6667787 (Joshua's-VPC)
10.0.0.0/16

Subnet

subnet-0c4d4d7dfbc9f1903 Joshua's-public-subnet
VPC: vpc-07eeed601d6667787 Owner: 864899869678 Availability Zone: us-east-1a
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.1.0/24)

Auto-assign public IP

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

Security group name - *required*

Joshua-sg

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

Description - *required*

launch-wizard-1 created 2025-01-24T19:25:00.488Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type	Protocol	Port range
ssh	TCP	22
Source type	Source	Description - *optional*
Anywhere	Add CIDR, prefix list or security group 0.0.0.0/0	e.g. SSH for admin desktop

Note: Security groups are stateful so any traffic allowed to ingress is automatically to egress.

8. Review the settings and launch the instance.

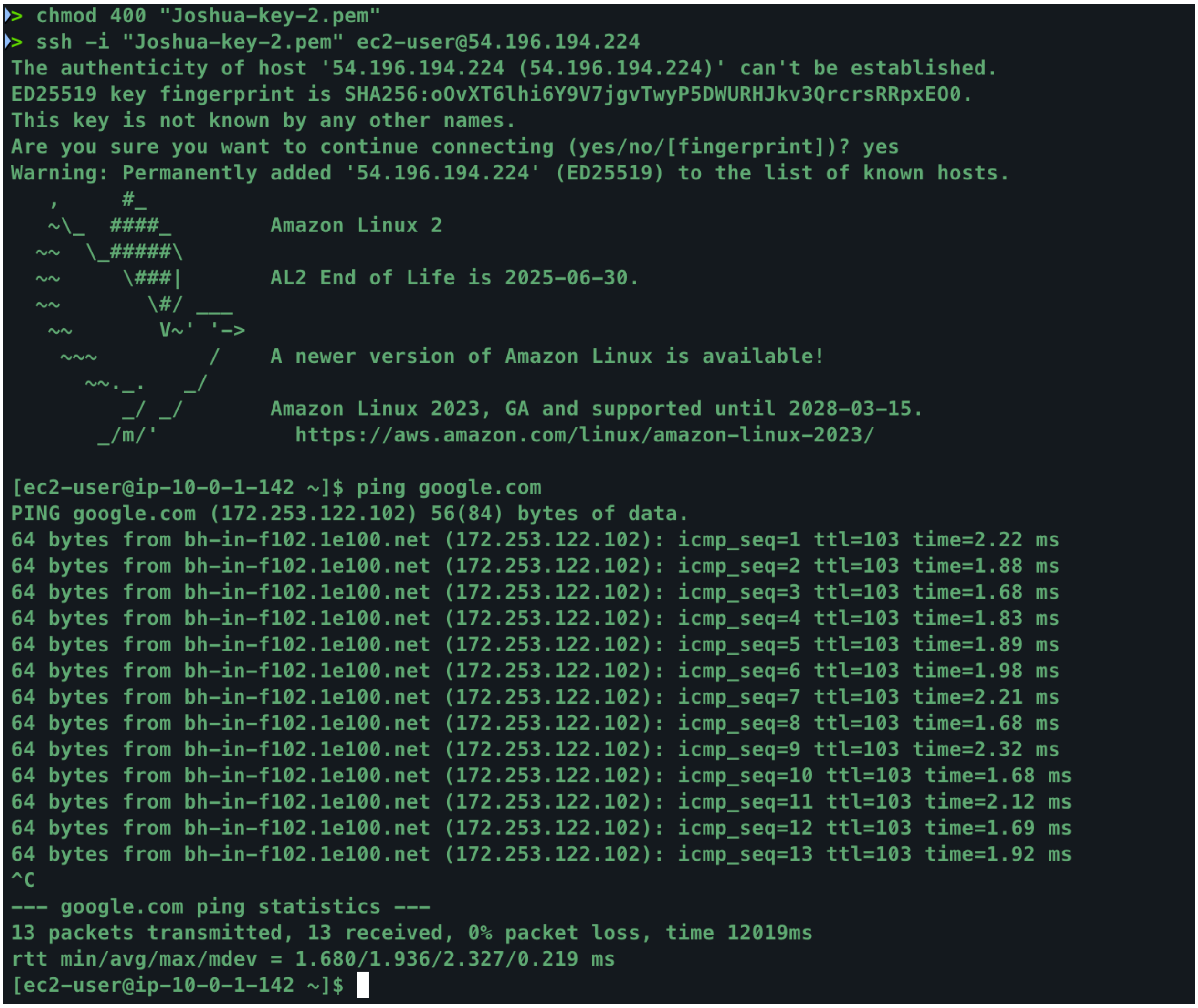
Now we verify our instances internet connection

1. In the EC2 dashboard, click on instances, click on the instance we created and click on connect.
2. Click on the SSH client and use the key pair we created when setting up our instance to SSH into the instance using your local terminal.

   i.e. ssh -i "[private-key].pem" ec2-user@[instance public ip addresses]

   The SSH Client option has instructions for connecting.
3. We ping google.com to confirm internet connectivity

We should get this after finishing:

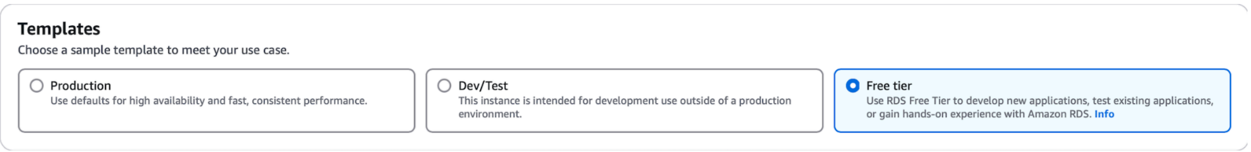


```
> chmod 400 "Joshua-key-2.pem"
> ssh -i "Joshua-key-2.pem" ec2-user@54.196.194.224
The authenticity of host '54.196.194.224 (54.196.194.224)' can't be established.
ED25519 key fingerprint is SHA256:oOvXT6lhi6Y9V7jgvTwyP5DWURHJkv3QrcrsRRpxEO0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.196.194.224' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-1-142 ~]$ ping google.com
PING google.com (172.253.122.102) 56(84) bytes of data.
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=1 ttl=103 time=2.22 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=2 ttl=103 time=1.88 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=3 ttl=103 time=1.68 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=4 ttl=103 time=1.83 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=5 ttl=103 time=1.89 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=6 ttl=103 time=1.98 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=7 ttl=103 time=2.21 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=8 ttl=103 time=1.68 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=9 ttl=103 time=2.32 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=10 ttl=103 time=1.68 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=11 ttl=103 time=2.12 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=12 ttl=103 time=1.69 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=13 ttl=103 time=1.92 ms
^C
--- google.com ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 12019ms
rtt min/avg/max/mdev = 1.680/1.936/2.327/0.219 ms
[ec2-user@ip-10-0-1-142 ~]$ 
```

RDS SETUP

We're going to add a RDS database to our VPC. The RDS database would reside in a private subnet, the reason is because we don't want our database to be publicly accessible from the internet.

1. Navigate to the RDS dashboard and click create database.
2. Select standard create and select MySQL as the engine option.
3. Choose the free tier template.



Templates

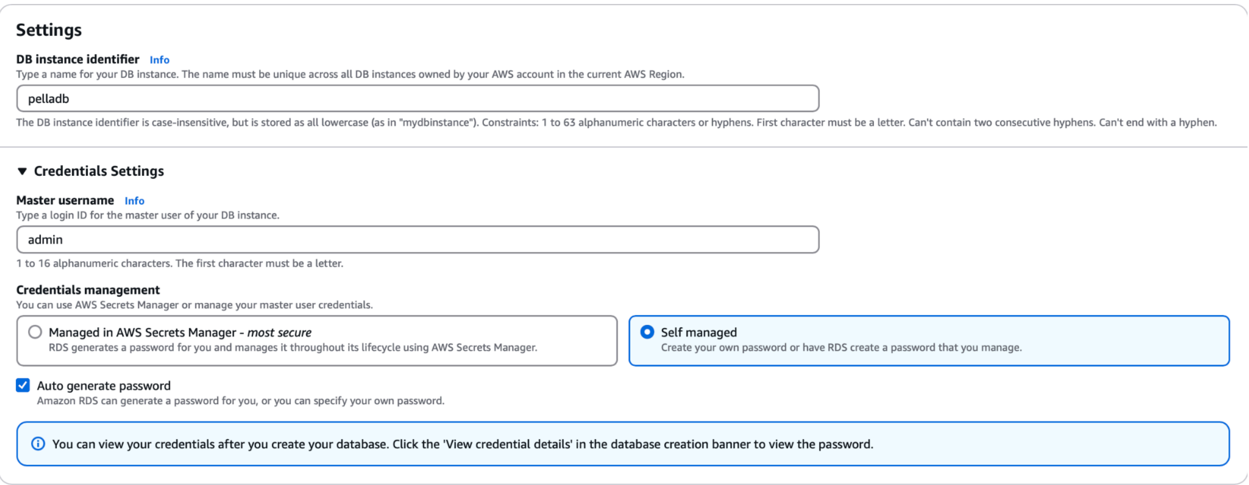
Choose a sample template to meet your use case.

○ Production
Use defaults for high availability and fast, consistent performance.

○ Dev/Test
This instance is intended for development use outside of a production environment.

● Free tier
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. Info

4. Give your database a name, a master username and master user credentials (either create or let AWS create it for you).



Settings

DB instance identifier Info
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

pelladb

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username Info
Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

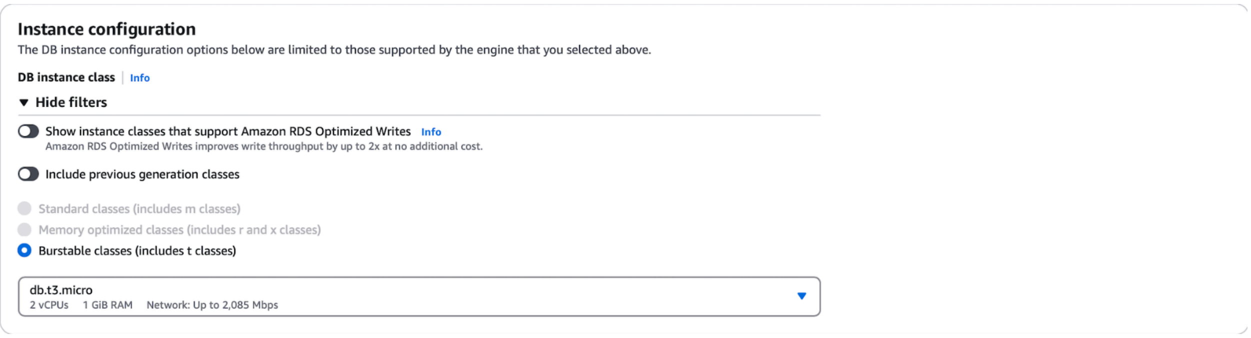
○ Managed in AWS Secrets Manager - *most secure*
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

● Self managed
Create your own password or have RDS create a password that you manage.

☑ Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

ⓘ You can view your credentials after you create your database. Click the 'View credential details' in the database creation banner to view the password.

5. Select db.t3.micro as instance configuration



Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class | Info

▼ **Hide filters**

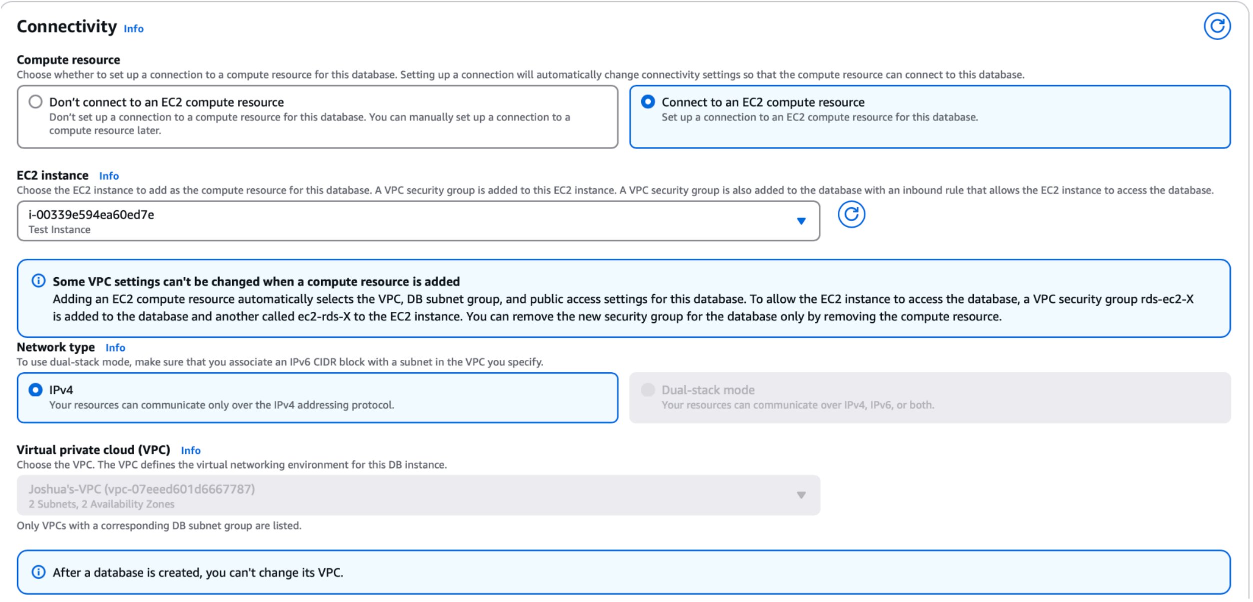
Show instance classes that support Amazon RDS Optimized Writes Info
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

Include previous generation classes

○ Standard classes (includes m classes)
○ Memory optimized classes (includes r and x classes)
● Burstable classes (includes t classes)

db.t3.micro
2 vCPUs 1 GiB RAM Network: Up to 2,085 Mbps

6. Under connectivity, select connect to an EC2 resource, select the instance we created and select IPv4 as the network type.



Connectivity Info

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

○ Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

● Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

EC2 instance Info
Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

i-00339e594ea60ed7e
Test Instance

Some VPC settings can't be changed when a compute resource is added
Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access the database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

Network type Info
To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

● IPv4
Your resources can communicate only over the IPv4 addressing protocol.

○ Dual-stack mode
Your resources can communicate over IPv4, IPv6, or both.

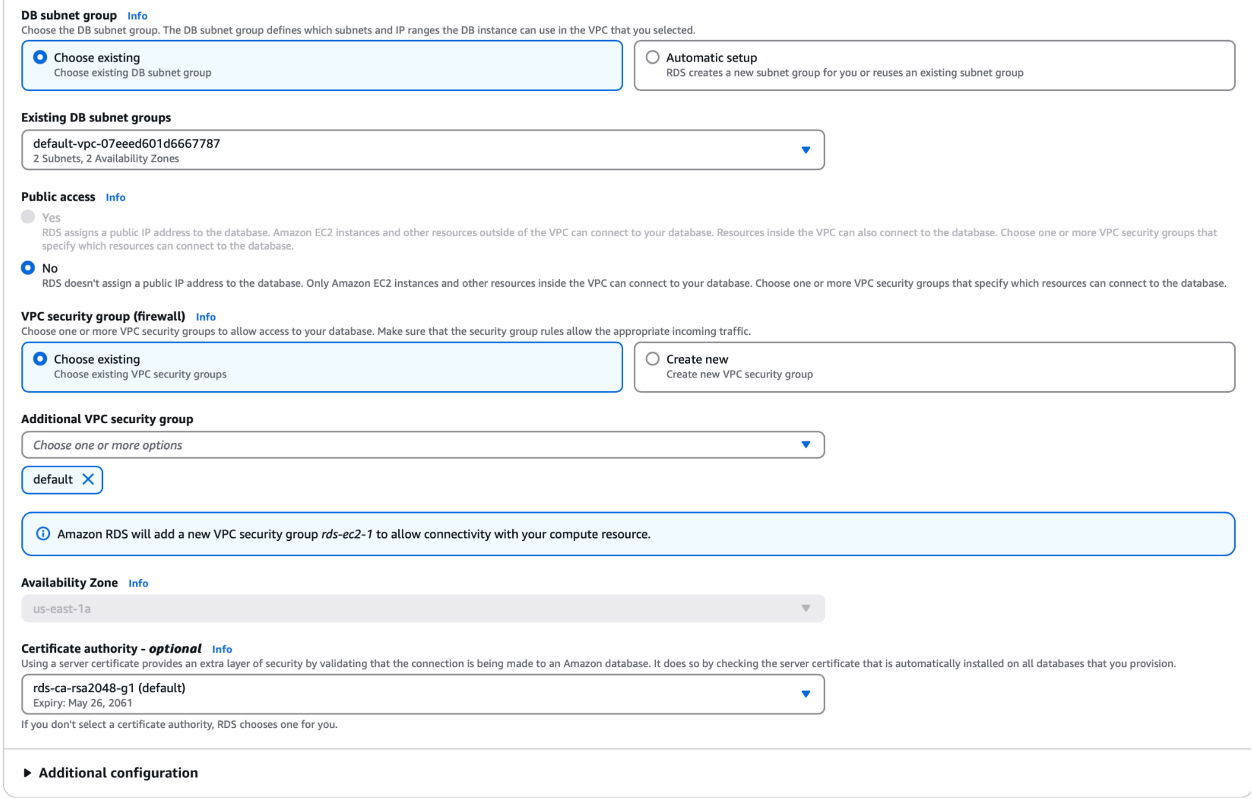
Virtual private cloud (VPC) Info
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Joshua's-VPC (vpc-07eeed601d6667787)
2 Subnets, 2 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

7. Choose an existing DB subnet group, set public access to no, choose an existing VPC security group and select default.



DB subnet group Info
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

● Choose existing
Choose existing DB subnet group

○ Automatic setup
RDS creates a new subnet group for you or reuses an existing subnet group

Existing DB subnet groups

default-vpc-07eeed601d6667787
2 Subnets, 2 Availability Zones

Public access Info

○ Yes
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

● No
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

Additional VPC security group

Choose one or more options

default ✕

Amazon RDS will add a new VPC security group *rds-ec2-1* to allow connectivity with your compute resource.

Availability Zone Info

us-east-1a

Certificate authority - *optional* Info
Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default)
Expiry: May 26, 2061

If you don't select a certificate authority, RDS chooses one for you.

▸ **Additional configuration**

8. Create your database.

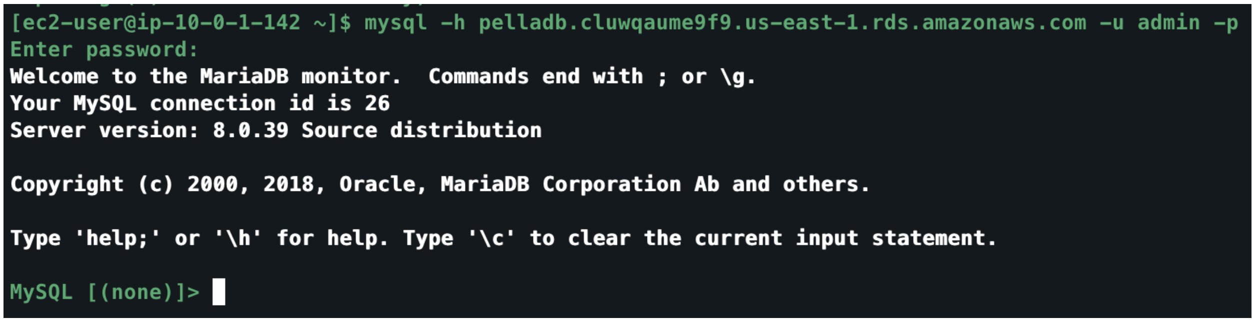
Now, we connect to our RDS instance from our EC2 instance:

1. SSH into your EC2 instance.
2. From your EC2 instance, install the MySQL client

   sudo yum install mysql -y

3. Use the MySQL client to connect to the RDS instance

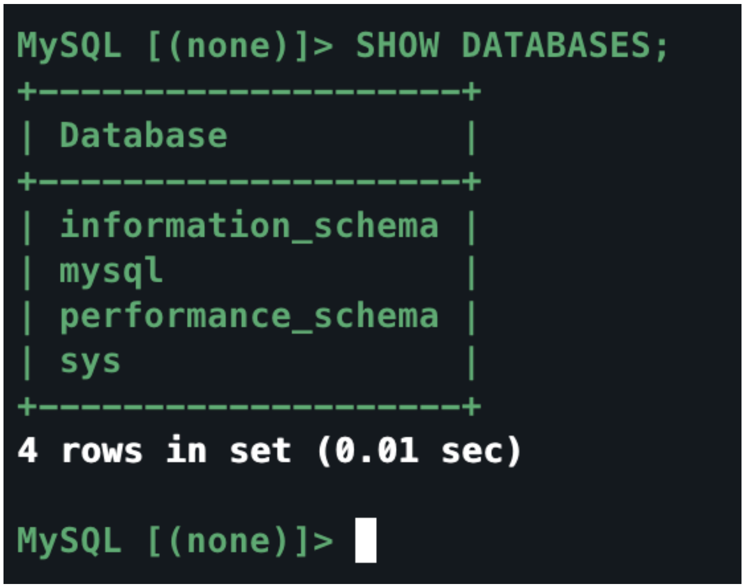
   mysql -h <RDS-endpoint> -u admin -p



```
[ec2-user@ip-10-0-1-142 ~]$ mysql -h pelladb.cluwqaume9f9.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 26
Server version: 8.0.39 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)]>
```

4. Once connected, run basic SQL commands to verify functionality

   SHOW DATABASES;



```
MySQL [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.01 sec)

MySQL [(none)]>
```

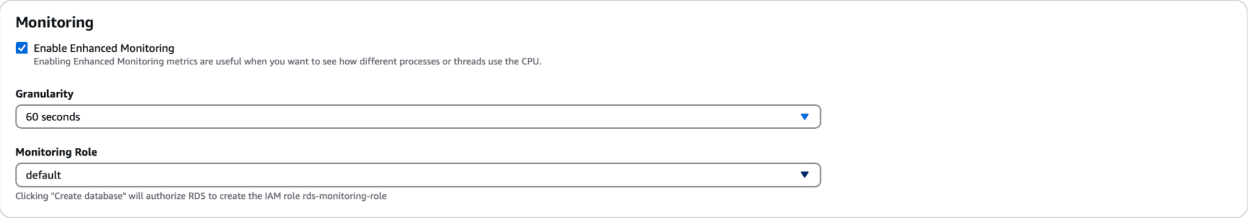
CLOUDWATCH SETUP

Amazon CloudWatch is a monitoring and observability service that provides actionable insights into your AWS resources, applications, and services. It collects and tracks metrics, logs, and events, enabling you to set alarms, analyze performance, and troubleshoot issues across your environment.

We would integrate CloudWatch into our project to monitor and maintain the health of our RDS instance.

Step 1: Enable enhanced monitoring

1. Go to the RDS dashboard and select your RDS instance.
2. Click modify and under the monitoring section, enable enhanced monitoring and you can use the default granularity (60 seconds)



Monitoring

☑ Enable Enhanced Monitoring
Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.

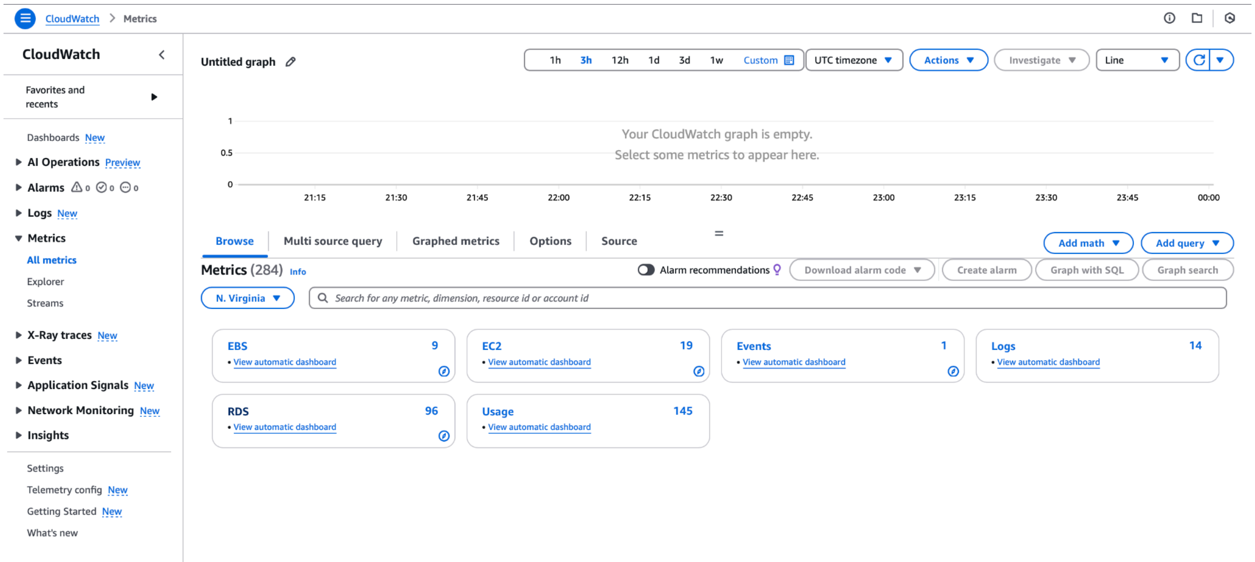
Granularity
60 seconds

Monitoring Role
default
Clicking "Create database" will authorize RDS to create the IAM role rds-monitoring-role

3. Click continue and apply immediately.

Step 2: View RDS metrics in cloud watch

1. Go to the CloudWatch dashboard
2. Navigate to Metrics>All metrics>RDS

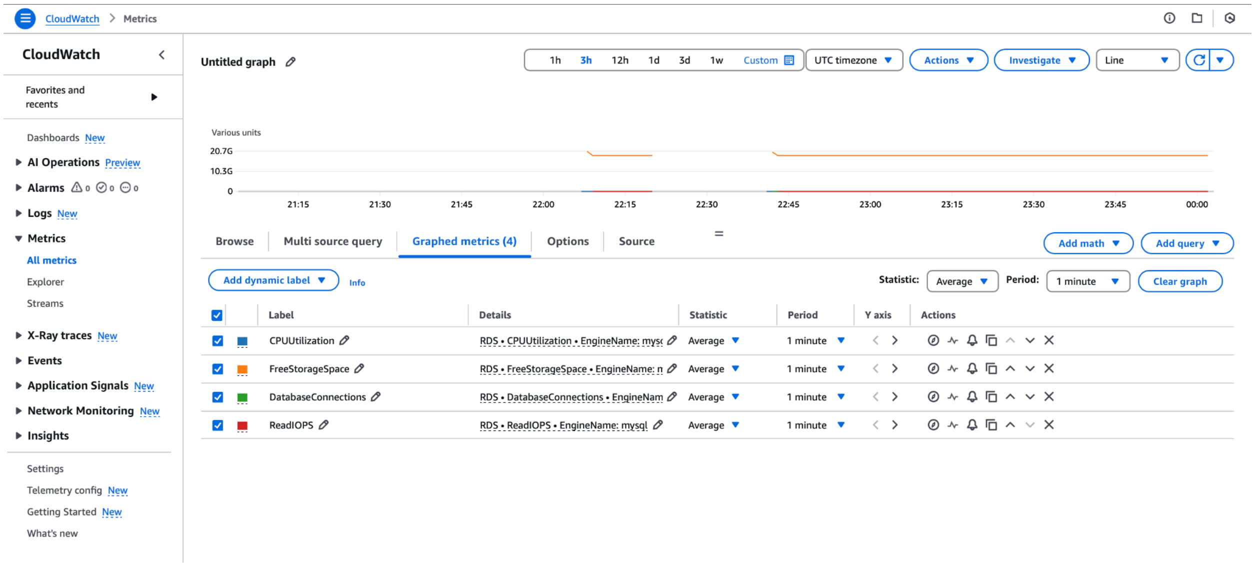


Your CloudWatch graph is empty.
Select some metrics to appear here.

Metrics (284)

EBS	9
EC2	19
Events	1
Logs	14
RDS	96
Usage	145

3. Look for key metrics:
   - **CPUUtilization**: Tracks database CPU usage.
   - **FreeStorageSpace**: Monitors available storage.
   - **DatabaseConnections**: Shows the number of active connections.
   - **ReadIOPS/WriteIOPS**: Measures input/output operations per second.
4. Explore the graph to view your database performance.

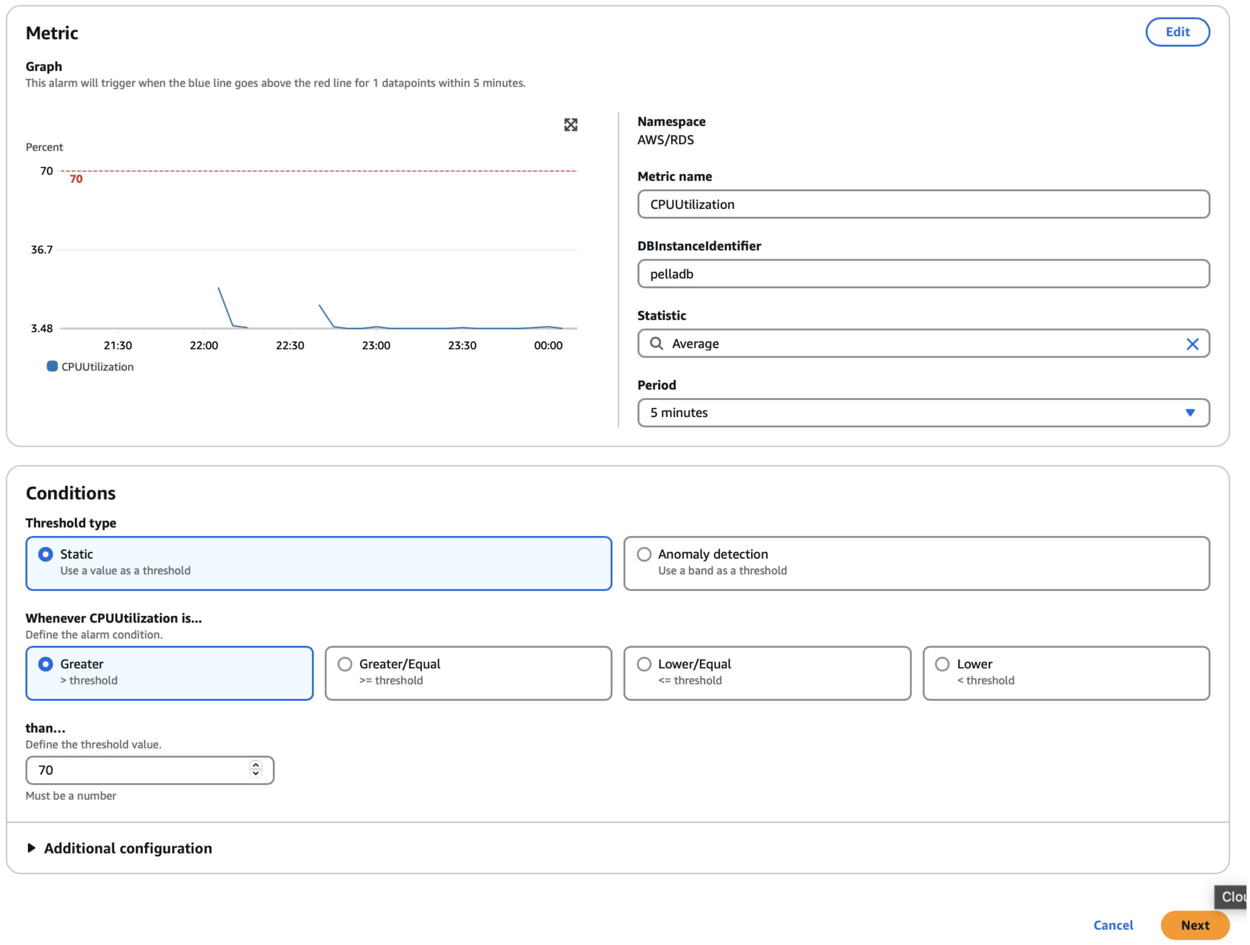


Statistic: Average Period: 1 minute

☑		Label	Details	Statistic	Period	Y axis	Actions
☑	🟦	CPUUtilization	RDS • CPUUtilization • EngineName: mys(	Average	1 minute		
☑	🟧	FreeStorageSpace	RDS • FreeStorageSpace • EngineName: n	Average	1 minute		
☑	🟩	DatabaseConnections	RDS • DatabaseConnections • EngineNam	Average	1 minute		
☑	🟥	ReadIOPS	RDS • ReadIOPS • EngineName: mysql	Average	1 minute		

Step 3: Set up alarms

1. Go to alarms in the CloudWatch dashboard.
2. Click create alarm and select the CPUUTILIZATION metric of your RDS instance.
3. Select the average statistic with a period of 5 minutes.
4. Select the greater than threshold and set the threshold value to 70 and click next.



Metric

Graph
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 5 minutes.

Namespace
AWS/RDS

Metric name
CPUUtilization

DBInstanceIdentifier
pelladb

Statistic
Average

Period
5 minutes

Conditions

Threshold type

● Static
Use a value as a threshold

○ Anomaly detection
Use a band as a threshold

Whenever CPUUtilization is...
Define the alarm condition.

● Greater
> threshold

○ Greater/Equal
>= threshold

○ Lower/Equal
<= threshold

○ Lower
< threshold

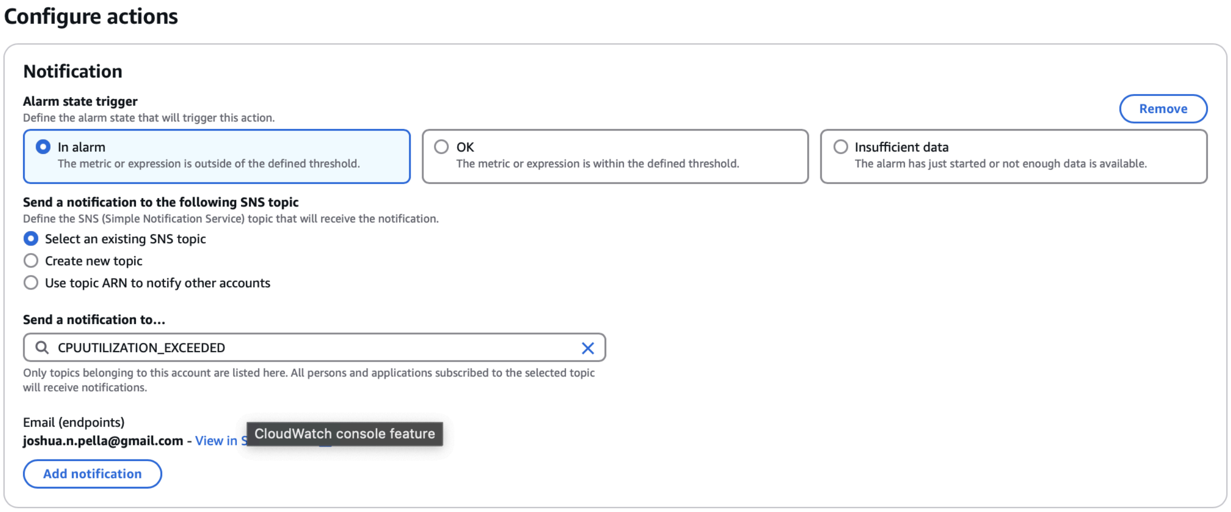
than...
Define the threshold value.

70

Must be a number

▶ **Additional configuration**

5. Select in alarm as the alarm state, create a new topic and type in your email so you can get notified and click next.



Configure actions

Notification

Alarm state trigger
Define the alarm state that will trigger this action.

● In alarm
The metric or expression is outside of the defined threshold.

○ OK
The metric or expression is within the defined threshold.

○ Insufficient data
The alarm has just started or not enough data is available.

Send a notification to the following SNS topic
Define the SNS (Simple Notification Service) topic that will receive the notification.

● Select an existing SNS topic
○ Create new topic
○ Use topic ARN to notify other accounts

Send a notification to...
CPUUTILIZATION_EXCEEDED

Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications.

Email (endpoints)
joshua.n.pella@gmail.com - View in S

6. Give the alarm a name and description then create the alarm.

Now you would be notified when your RDS instance exceeds the configured threshold.