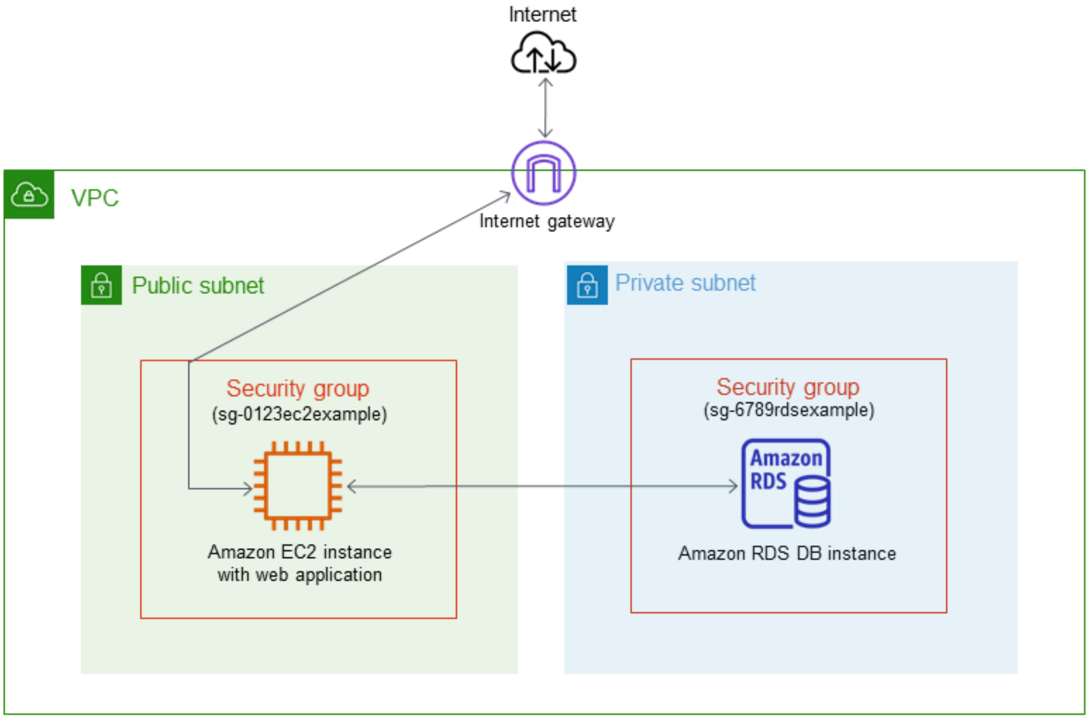
Midterm Project

This tutorial helps you install an Apache web server with PHP and create a MySQL database. The web server runs on an Amazon EC2 instance using Amazon Linux, and the MySQL database is an MySQL DB instance. Both the Amazon EC2 instance and the DB instance run in a virtual private cloud (VPC) based on the Amazon VPC service.

In the lab that follows, you specify the VPC, subnets, and security groups when you create the DB instance. You also specify them when you create the EC2 instance to host your web server. The VPC, subnets, and security groups are required for the DB instance and the web server to communicate. After the VPC is set up, this tutorial shows you how to create the DB instance and install the web server. You connect your web server to your DB instance in the VPC using the DB instance endpoint endpoint.



Before you begin this tutorial, make sure that you have a VPC with both public and private subnets, and corresponding security groups. If you don't have these, complete the following tasks in the tutorial:

Prerequisites:

FROM YOUR COMPUTER or FROM CLOUD9:
CREATE A S3 BUCKET FROM THE COMMAND LINE

```
aws s3 mb s3://bucket-name
```

MOVE FILE TO S3 BUCKET
Download the following file to your computer and then copy it to your s3 bucket.
File: SamplePage.php

```
aws s3 cp SamplePage.php s3://bucket-name/
```

N**OW FROM THE CONSOLE:**

- Create an EC2 role and attach the policy **AmazonS3FullAccess**
- **Name this role midterm-ec2-s3-role**
- You will use this role when creating your ec2 webserver.

Create a VPC with private and public subnets

Use the following procedure to create a VPC with both public and private subnets.

To create a VPC and subnets

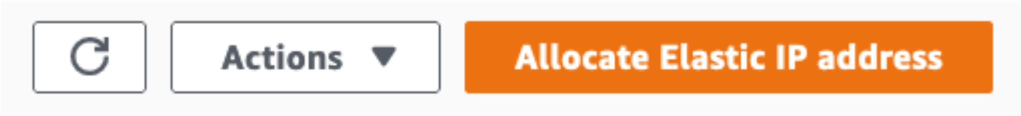
- If you don't have an Elastic IP address to associate with a network address translation (NAT) gateway, allocate one now. A NAT gateway is required for this tutorial. If you have an available Elastic IP address, move on to the next step.

- Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/

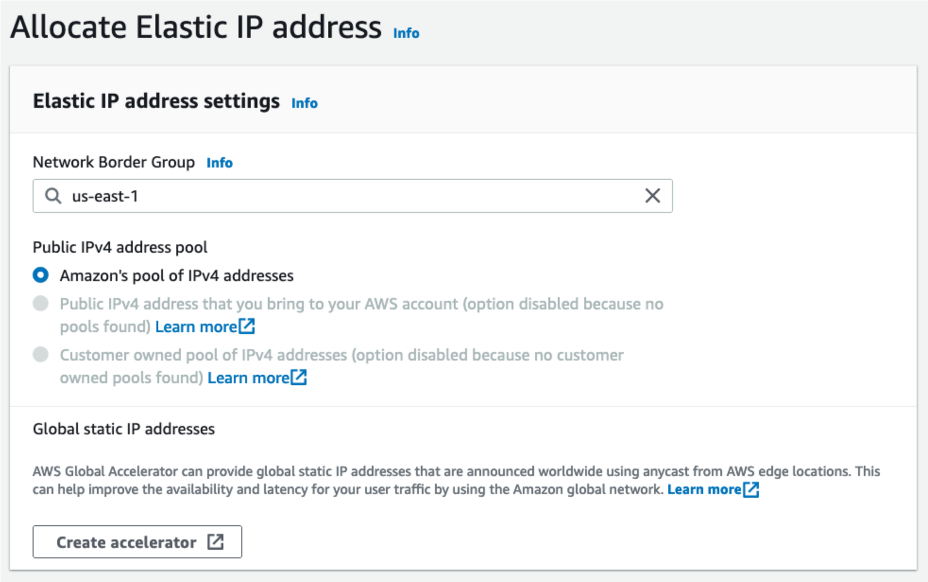
- In the top-right corner of the AWS Management Console, choose the Region to allocate your Elastic IP address in. The Region of your Elastic IP address should be the same as the Region where you want to create your VPC. This example uses the US East (N. Virginia) Region.

- In the navigation panel, choose **Elastic IPs**.

- Choose **Allocate Elastic IP address**.



- If the console shows the **Network Border Group** field, keep the default value for it.



Allocate Elastic IP address Info

Elastic IP address settings Info

Network Border Group Info

us-east-1

Public IPv4 address pool

● Amazon's pool of IPv4 addresses

○ Public IPv4 address that you bring to your AWS account (option disabled because no pools found) Learn more

○ Customer owned pool of IPv4 addresses (option disabled because no customer owned pools found) Learn more

Global static IP addresses

AWS Global Accelerator can provide global static IP addresses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using the Amazon global network. Learn more

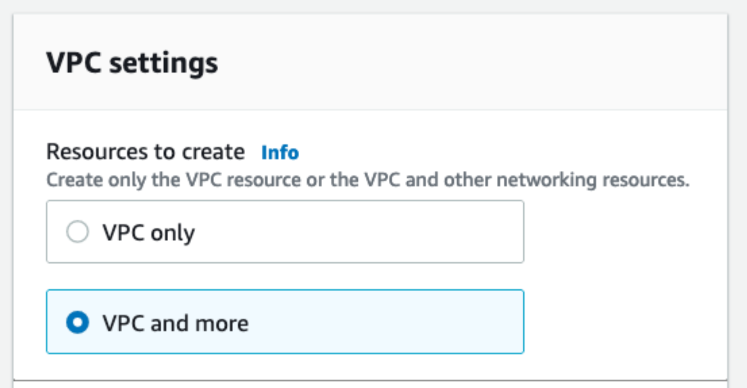
- For **Public IPv4 address pool**, choose **Amazon's pool of IPv4 addresses**.
- Choose **Allocate**.

  Note the allocation ID of the new Elastic IP address because you'll need this information when you create your VPC.

2. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/

   - In the top-right corner of the AWS Management Console, choose the Region to create your VPC in. This example uses the US East (N. Virginia) Region.
   - In the upper-left corner, choose **VPC Dashboard**. To begin creating a VPC, choose **Create VPC**.

   - On the **Top part: On the VPC settings section**, choose **VPC and more**, and then choose **Select**.



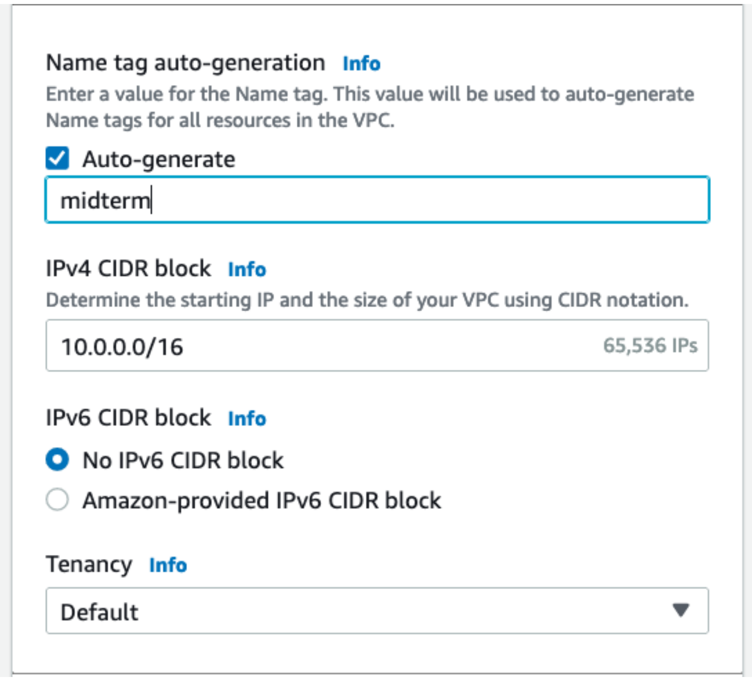
VPC settings

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

○ VPC only

● VPC and more

   - On the **Second part: Add a name to our new VPC**, set these values:



Name tag auto-generation

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☑ Auto-generate

midterm

IPv4 CIDR block

Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16 65,536 IPs

IPv6 CIDR block

● No IPv6 CIDR block

○ Amazon-provided IPv6 CIDR block

Tenancy

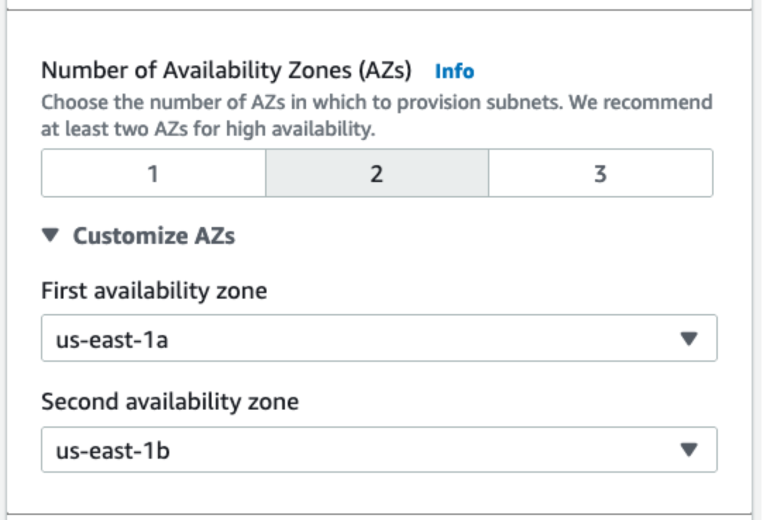
Default ▼

1. **IPv4 CIDR block:** 10.0.0.0/16

2. **IPv6 CIDR block:** No IPv6 CIDR Block

3. **VPC name:** midterm

- On the **Third part: Number of Availability Zones**, set these values:



Number of Availability Zones (AZs)

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1	2	3

▼ Customize AZs

First availability zone

us-east-1a ▼

Second availability zone

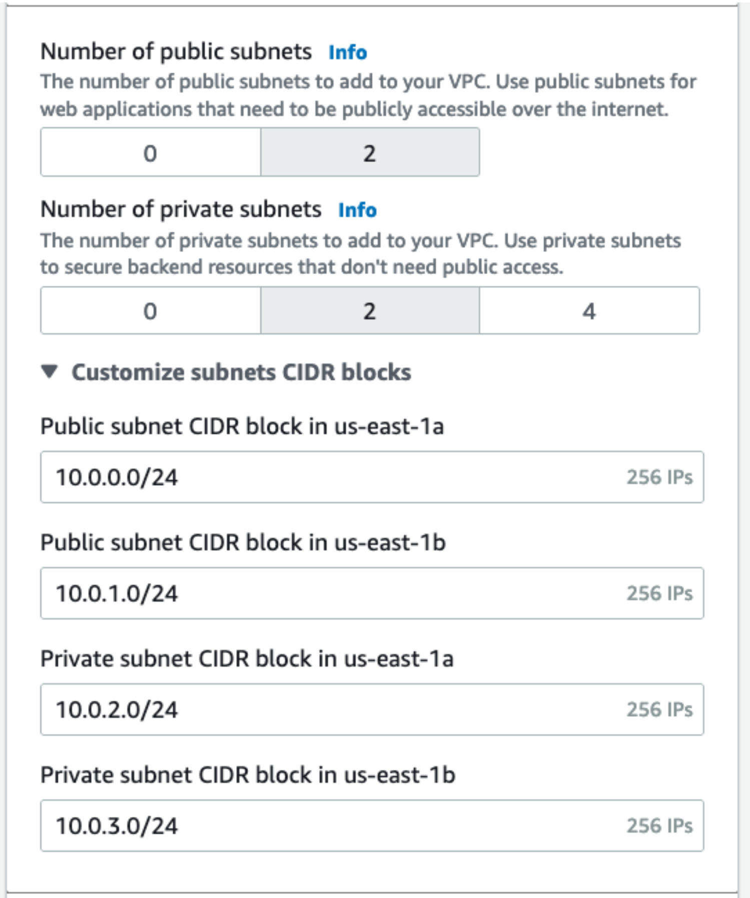
us-east-1b ▼

Number of Availability Zones (AZs): 2

Click on the drop down icon next to the title **Customize AZs** to select the availability zones you need for this tutorial.

1. **First Availability Zone:** us-east-1a
2. **Second Availability Zone:** us-east-1b

- On the **Next section you need to specify how many public subnets you need and also how many private subnets**, set these values:



Number of public subnets Info

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 | 2

Number of private subnets Info

The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 | 2 | 4

▼ Customize subnets CIDR blocks

Public subnet CIDR block in us-east-1a

10.0.0.0/24 256 IPs

Public subnet CIDR block in us-east-1b

10.0.1.0/24 256 IPs

Private subnet CIDR block in us-east-1a

10.0.2.0/24 256 IPs

Private subnet CIDR block in us-east-1b

10.0.3.0/24 256 IPs

-

1. **Number of public subnets:** 2

2. **Number of private subnets:** 2

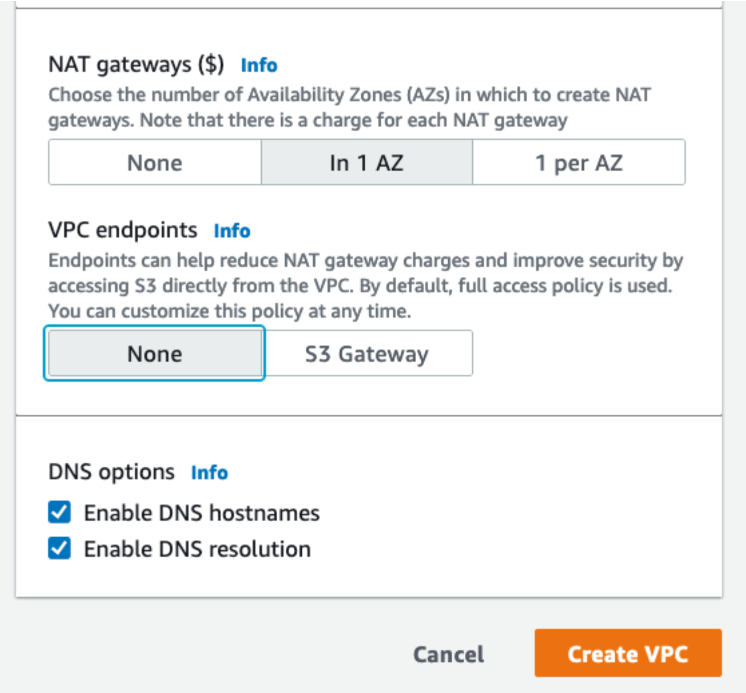
3. **Public subnet CIDR block in us-east-1a:** 10.0.0.0/24

4. **Public subnet CIDR block in us-east-1b** 10.0.1.0/24

5. **Private subnet CIDR block in us-east-1a**: 10.0.2.0/24

6. **Private subnet CIDR block in us-east-1b** 10.0.3.0/24

On the final step lets specify how many NAT Gateways we will use and also disable the VPC endpoints. For the DNS options on the bottom lets go with the default values.



NAT gateways ($) Info

Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway

None | [✓ In 1 AZ] | 1 per AZ

VPC endpoints Info

Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

[✓ None] | S3 Gateway

DNS options Info

☑ Enable DNS hostnames

☑ Enable DNS resolution

7. **NAT gateway:** In 1 AZ

8. **VPC endpoints:** None

9. **Enable DNS hostnames:** Yes

10. **Enable DNS resolution:** Yes

- Choose **Create VPC**.

- This wizard will set route tables and subnets associations for you, also it will attach the Nat Gateway to the private route tables and an Internet Gateway to your VPC with the route entry added to the public route table.

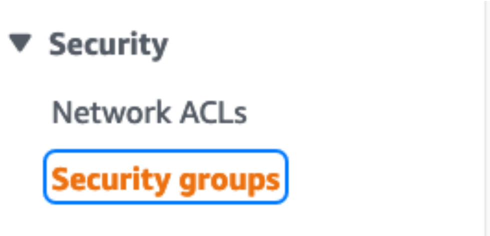
Create a VPC security group for a public web server

Next you create a security group for public access. To connect to public instances in your VPC, you add inbound rules to your VPC security group that allow traffic to connect from the internet.

To create a VPC security group

3. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/

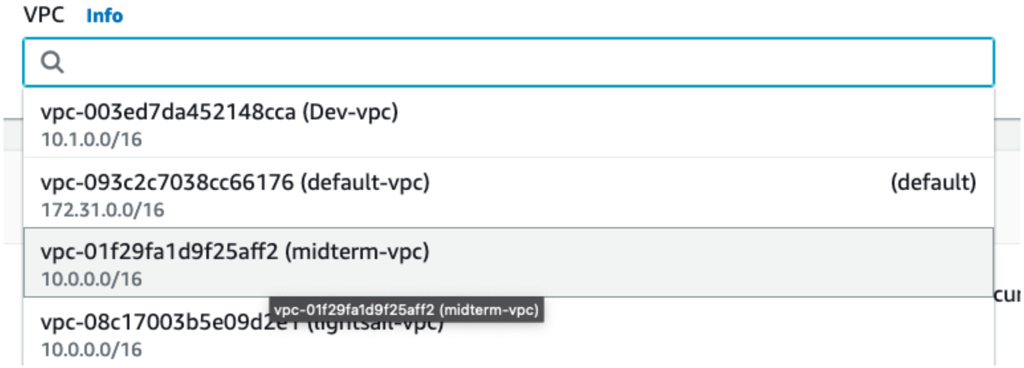
4. Choose **VPC Dashboard**, choose **Security Groups**:



5. Then choose **Create security group**.

6. On the **Create security group** page, set these values:

   ○ **Security group name:** tutorial-securitygroup
   ○ **Description:** Tutorial Security Group
   ○ **VPC:** Choose the VPC that you created earlier, for example: vpc-identifier (midterm-vpc)



vpc-003ed7da452148cca (Dev-vpc)
10.1.0.0/16

vpc-093c2c7038cc66176 (default-vpc) (default)
172.31.0.0/16

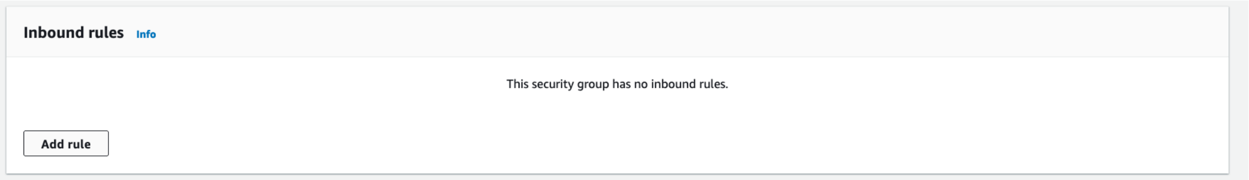
vpc-01f29fa1d9f25aff2 (midterm-vpc)
10.0.0.0/16

vpc-01f29fa1d9f25aff2 (midterm-vpc)

vpc-08c17003b5e09d2e1 (lightsail-vpc)
10.0.0.0/16

7. Add inbound rules to the security group.

   1. In the **Inbound rules** section, choose **Add rule**.



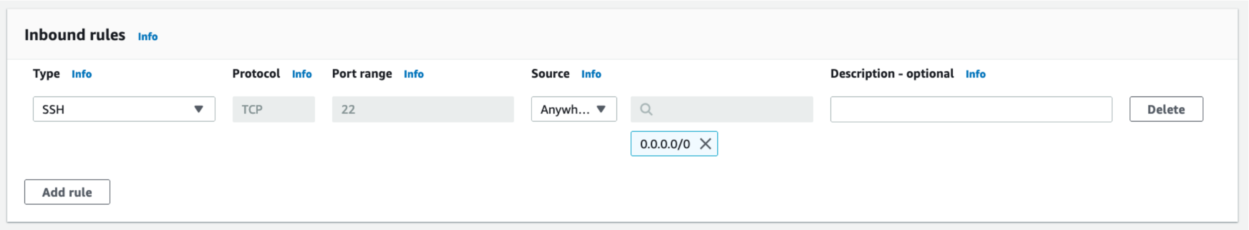
Inbound rules Info

This security group has no inbound rules.

Add rule

   2. Set the following values for your new inbound rule to allow Secure Shell (SSH) access to your EC2 instance. If you do this, you can connect to your EC2 instance to install the web server and other utilities, and to upload content for your web server.

      - **Type:** SSH
      - **Source:** 0.0.0.0/0 (Not recommended in production environments but for this lab is good enough).



Inbound rules Info

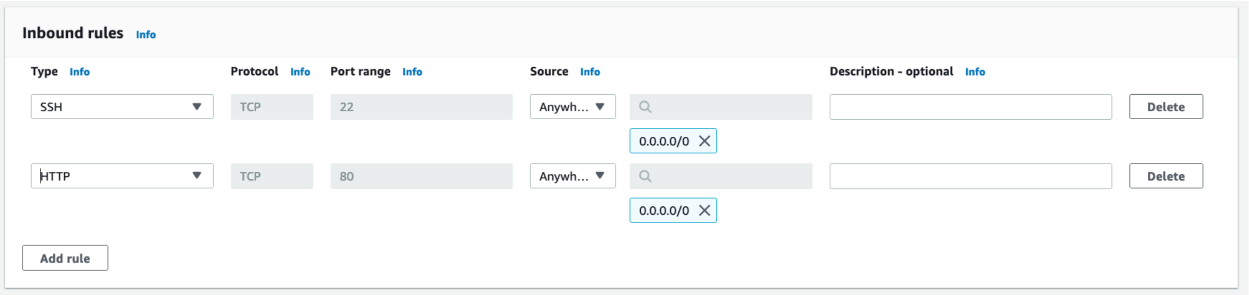
Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
SSH	TCP	22	Anywh...	0.0.0.0/0		Delete

Add rule

   3. Choose **Add rule**.

   4. Set the following values for your new inbound rule to allow HTTP access to your web server.

      - **Type:** HTTP
      - **Source:** 0.0.0.0/0



Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
SSH	TCP	22	Anywh...	0.0.0.0/0		Delete
HTTP	TCP	80	Anywh...	0.0.0.0/0		Delete

Add rule

NOTE: Since Security Groups are stateful you don't need to modify the outbound rules, there is only one rule there that will allow all traffic to go out, so don't delete or modify this rule.

- To create the security group, choose **Create security group**.

  Note the security group ID because you need it later in this tutorial.

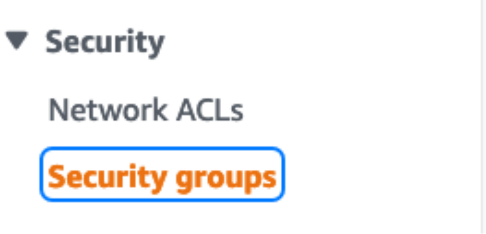
Create a VPC security group for a private DB instance

To keep your DB instance private, create a second security group for private access. To connect to private instances in your VPC, you add inbound rules to your VPC security group that allow traffic from your web server only.

To create a VPC security group

- Open the Amazon VPC console at https://console.aws.amazon.com/vpc/

- Choose **VPC Dashboard**, choose **Security Groups**.

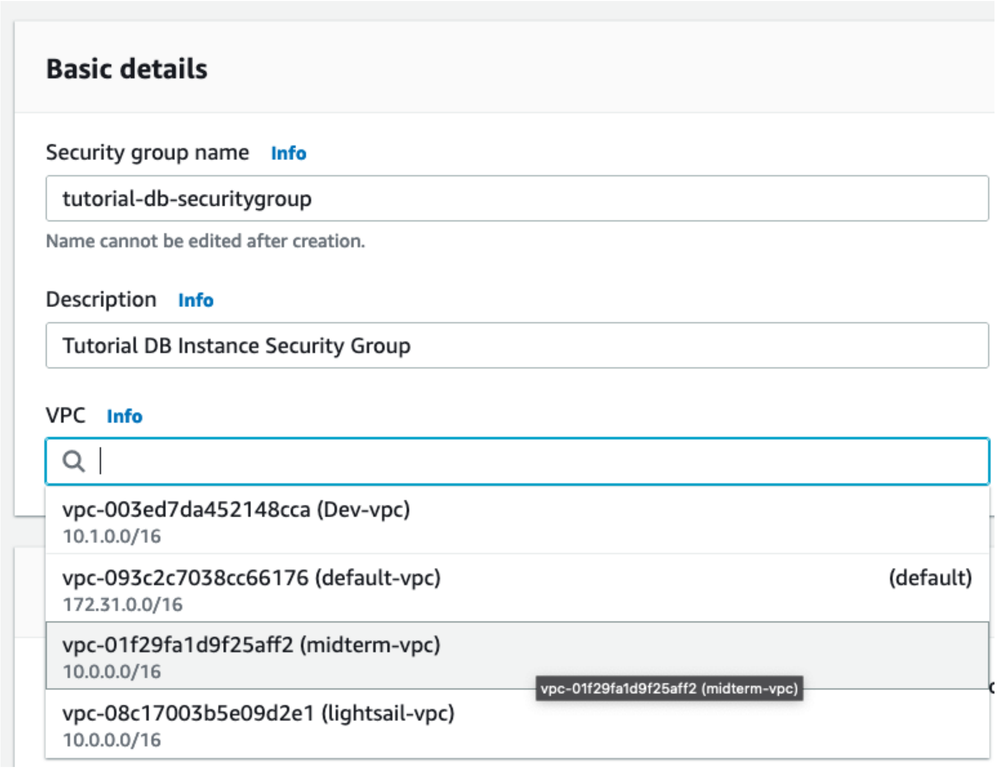
- Then choose **Create security group**.

- On the **Create security group** page, set these values:

  1. **Security group name:** tutorial-db-securitygroup

  2. **Description:** Tutorial DB Instance Security Group

  3. **VPC:** Choose the VPC that you created earlier, for example: vpc-identifier (midterm-vpc)



Basic details

Security group name Info

tutorial-db-securitygroup

Name cannot be edited after creation.

Description Info

Tutorial DB Instance Security Group

VPC Info

vpc-003ed7da452148cca (Dev-vpc)
10.1.0.0/16

vpc-093c2c7038cc66176 (default-vpc) (default)
172.31.0.0/16

vpc-01f29fa1d9f25aff2 (midterm-vpc)
10.0.0.0/16

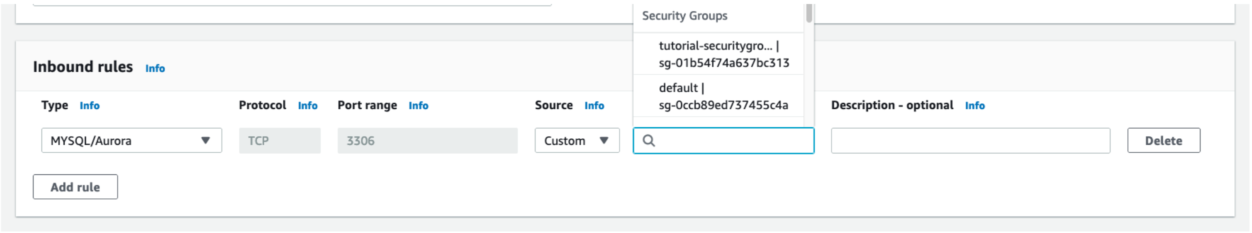
vpc-01f29fa1d9f25aff2 (midterm-vpc)

vpc-08c17003b5e09d2e1 (lightsail-vpc)
10.0.0.0/16

- ○ Add inbound rules to the security group.

    1. In the **Inbound rules** section, choose **Add rule**.
    2. Set the following values for your new inbound rule to allow MySQL/Aurora traffic on port 3306 from your EC2 instance. If you do this, you can connect from your web server to your DB instance to store and retrieve data from your web application to your database.

        - ■ **Type:** MySQL/Aurora
        - ■ **Source:** The identifier of the tutorial-securitygroup security group that you created previously in this tutorial



Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
MYSQL/Aurora ▼	TCP	3306	Custom ▼	🔍		Delete

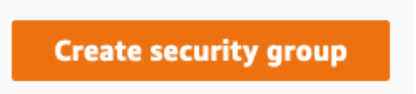
Security Groups

tutorial-securitygro... |
sg-01b54f74a637bc313

default |
sg-0ccb89ed737455c4a

Add rule

- ○ To create the security group, choose **Create security group**.



Create a DB subnet group

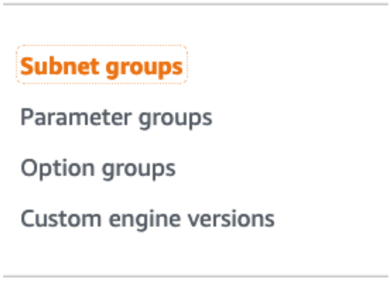
A DB subnet group is a collection of subnets that you create in a VPC and that you then designate for your DB instances. A DB subnet group allows you to specify a particular VPC when creating DB instances.

To create a DB subnet group

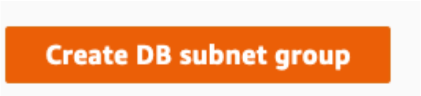
8. Open the Amazon RDS console at https://console.aws.amazon.com/rds/

   Note: Make sure you connect to the Amazon RDS console, not to the Amazon VPC console.

9. In the navigation pane, choose **Subnet groups**.

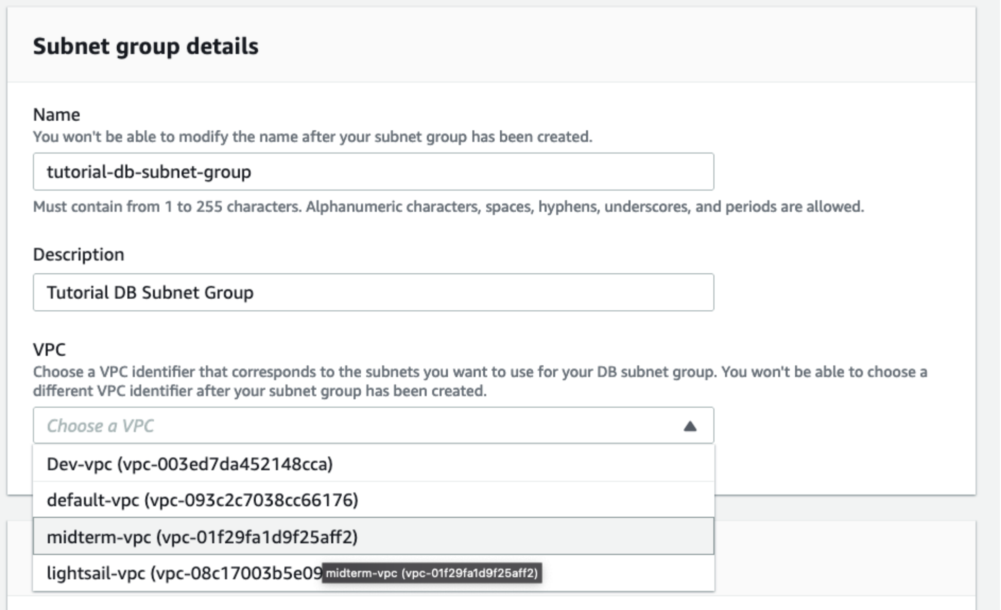


10. Choose **Create DB Subnet Group**.



11. On the **Create DB subnet group** page, set these values in **Subnet group details**:

    - **Name:** tutorial-db-subnet-group
    - **Description:** Tutorial DB Subnet Group
    - **VPC:** midterm-vpc (vpc-identifier)



Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.

tutorial-db-subnet-group

Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description

Tutorial DB Subnet Group

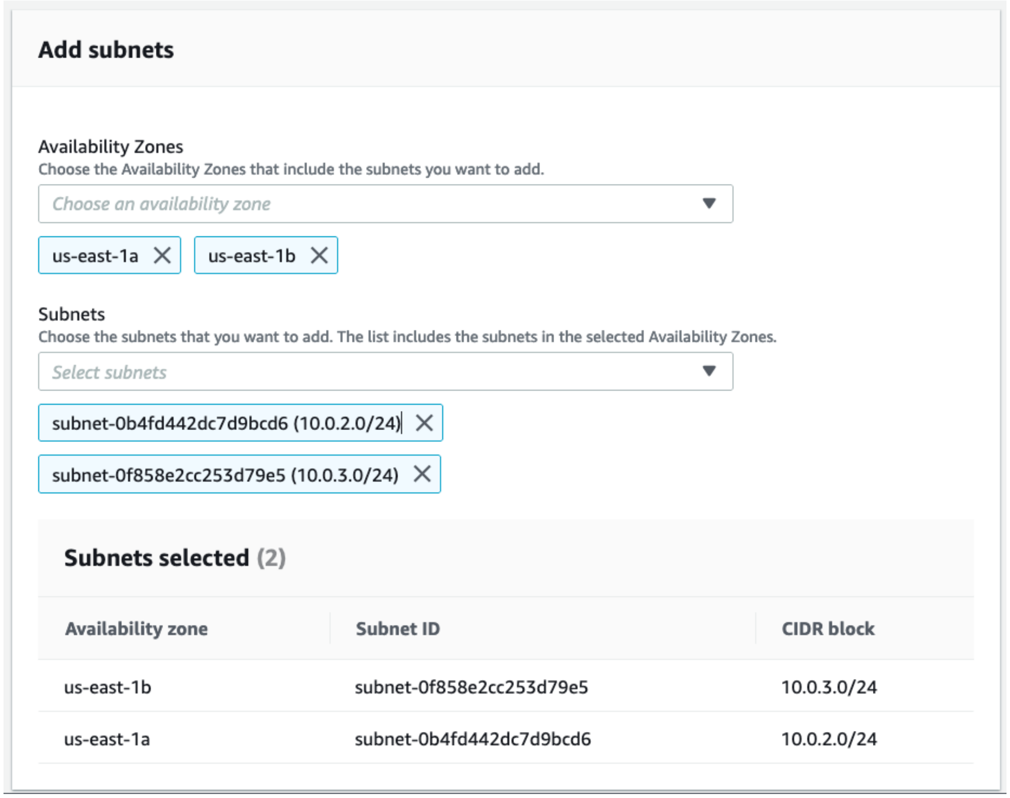
VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.

Choose a VPC

Dev-vpc (vpc-003ed7da452148cca)
default-vpc (vpc-093c2c7038cc66176)
midterm-vpc (vpc-01f29fa1d9f25aff2)
lightsail-vpc (vpc-08c17003b5e09

12. In the **Add subnets** section, choose the **Availability Zones** and **Subnets**.

    For this tutorial, choose us-east-1a and us-east-1b for the **Availability Zones**. Next, for **Subnets**, choose the subnets for IPv4 CIDR block 10.0.2.0/24 and 10.0.3.0/24.



Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.

Choose an availability zone

us-east-1a ✕ us-east-1b ✕

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.

Select subnets

subnet-0b4fd442dc7d9bcd6 (10.0.2.0/24) ✕

subnet-0f858e2cc253d79e5 (10.0.3.0/24) ✕

Subnets selected (2)

Availability zone	Subnet ID	CIDR block
us-east-1b	subnet-0f858e2cc253d79e5	10.0.3.0/24
us-east-1a	subnet-0b4fd442dc7d9bcd6	10.0.2.0/24

13. Choose **Create**.

Your new DB subnet group appears in the DB subnet groups list on the RDS console. You can click the DB subnet group to see details, including all of the subnets associated with the group, in the details pane at the bottom of the window.

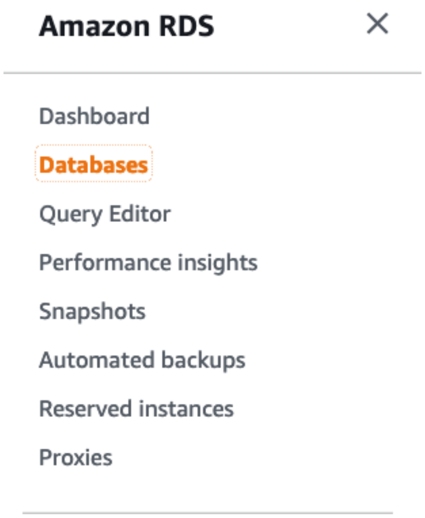
Now lets create a DB instance

1. **To create a MySQL DB instance**

   Sign in to the AWS Management Console and open the Amazon RDS console at https://console.aws.amazon.com/rds/

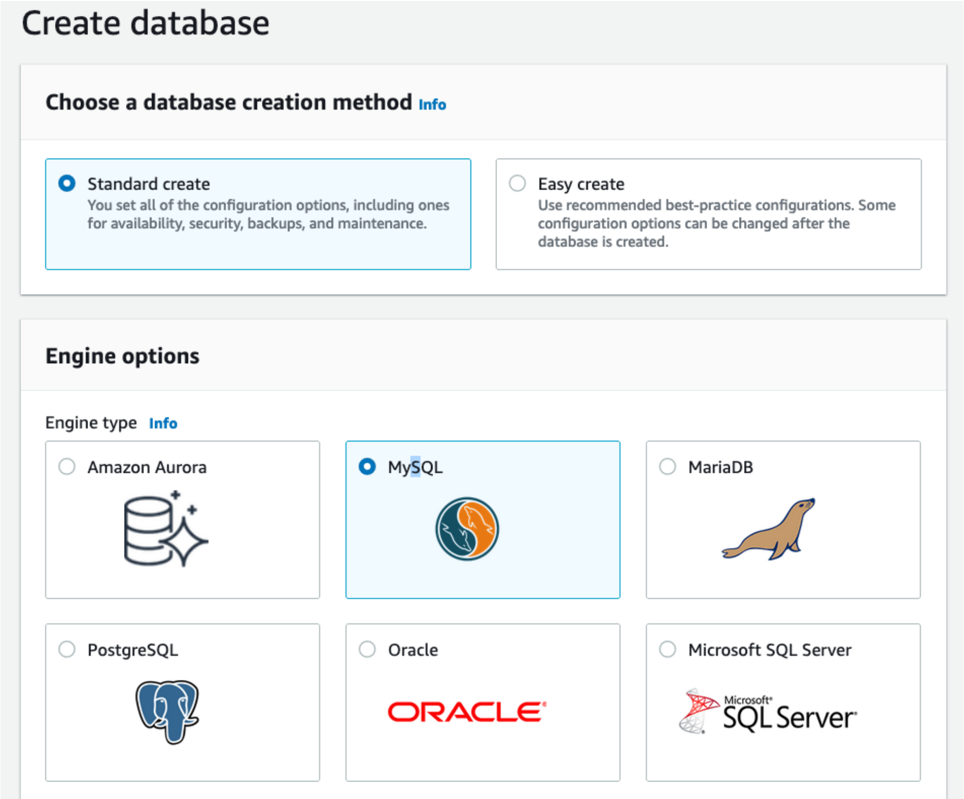
2. In the upper-right corner of the AWS Management Console, choose the AWS Region where you want to create the DB instance. This example uses the US East (N. Virginia) Region.

3. In the navigation pane, choose **Databases**.



4. Choose **Create database**.

5. On the **Create database** page, shown following, make sure that the **Standard create** option is chosen, and then choose **MySQL**.



Create database

Choose a database creation method Info

● Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

○ Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type Info

○ Amazon Aurora

● MySQL

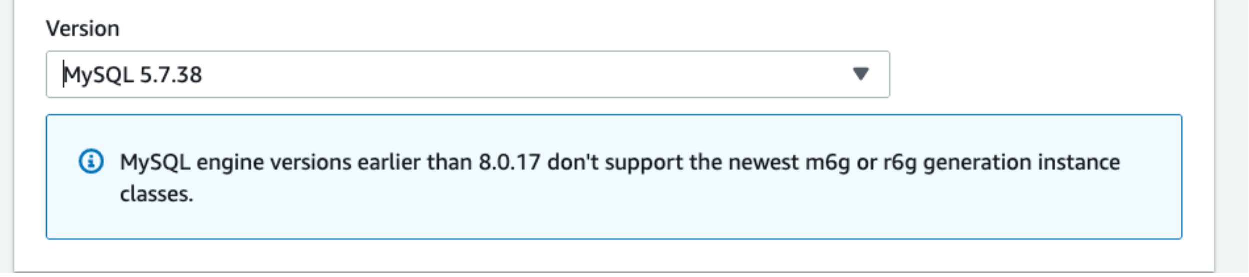
○ MariaDB

○ PostgreSQL

○ Oracle

○ Microsoft SQL Server

Make sure that for the **Version** you pick **5.7.38**

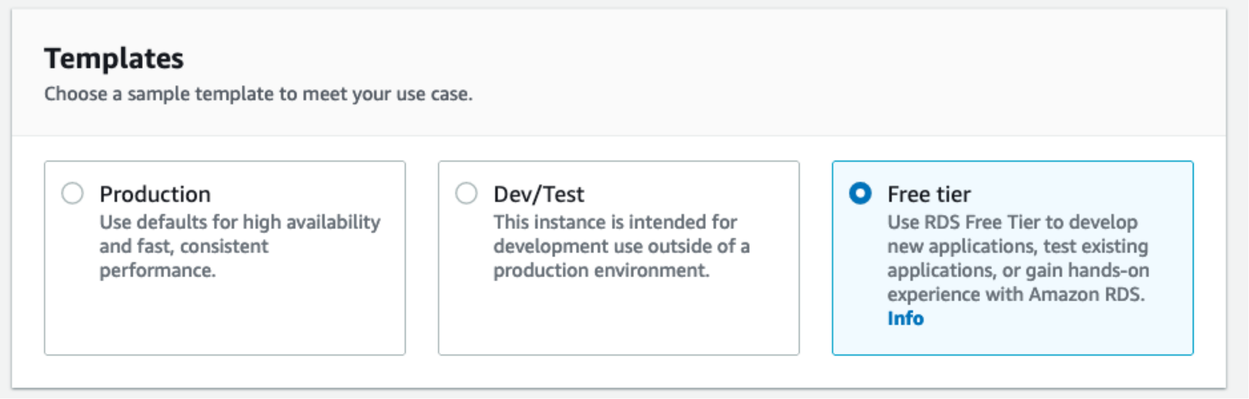


Version

MySQL 5.7.38 ▼

ⓘ MySQL engine versions earlier than 8.0.17 don't support the newest m6g or r6g generation instance classes.

6. In the **Templates** section, choose **Free tier**.



Templates

Choose a sample template to meet your use case.

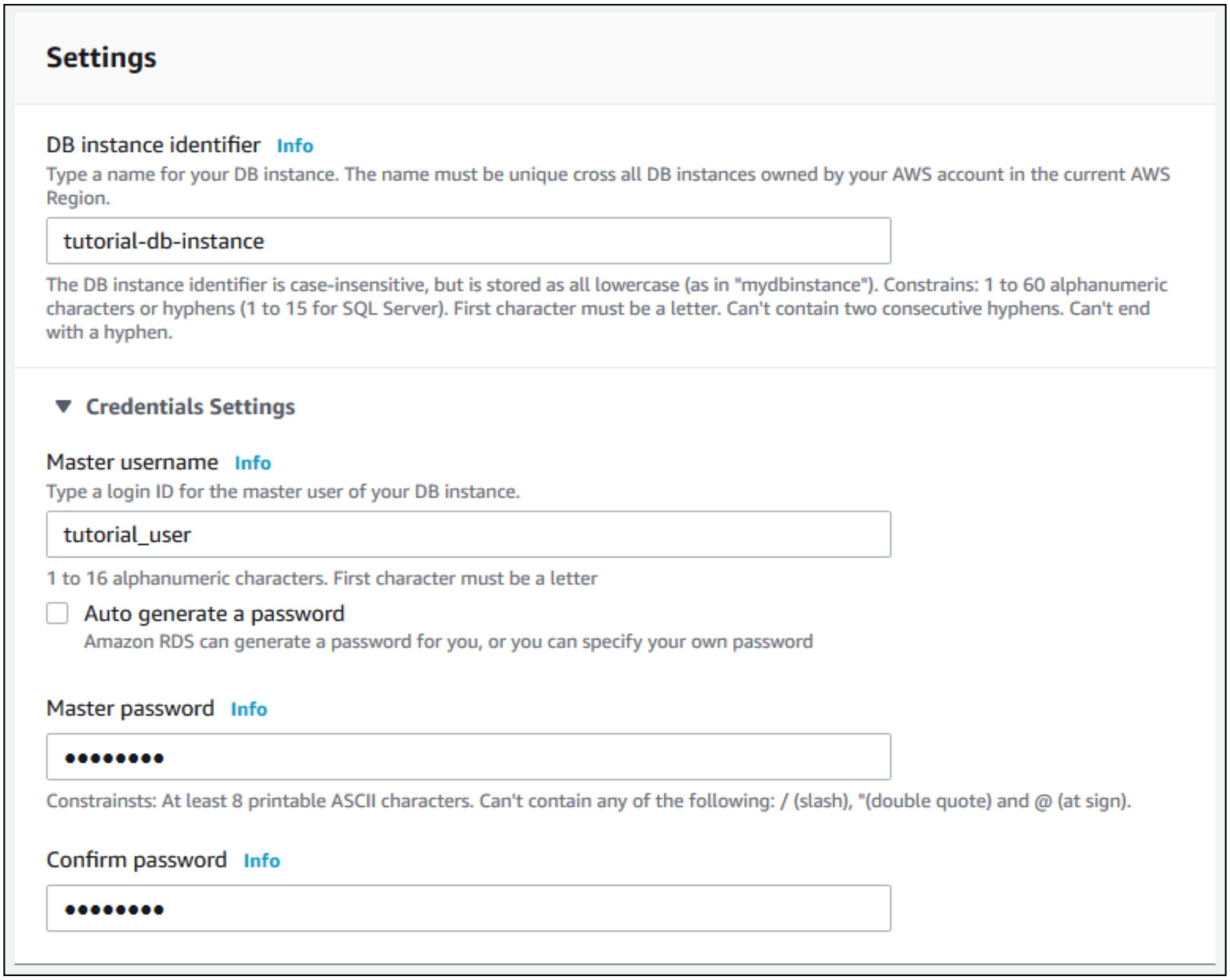
○ Production
Use defaults for high availability and fast, consistent performance.

○ 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

7. In the **Settings** section, set these values:

    - **DB instance identifier** – tutorial-db-instance
    - **Master username** – tutorial_user
    - **Auto generate a password** – Disable the option.
    - **Master password** – Choose a password.
    - **Confirm password** – Retype the password.



Settings

DB instance identifier Info

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

tutorial-db-instance

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constrains: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). 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.

tutorial_user

1 to 16 alphanumeric characters. First character must be a letter

□ Auto generate a password

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

Master password Info

••••••••

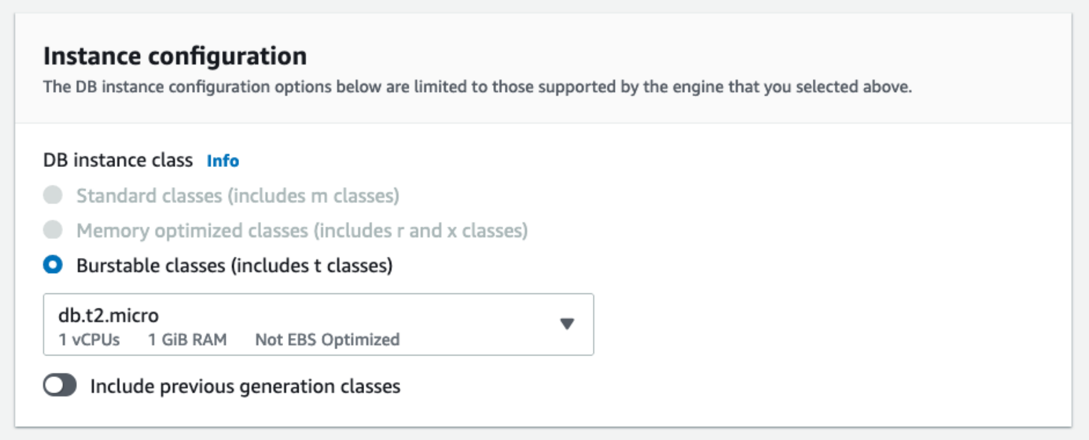
Constrainsts: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), "(double quote) and @ (at sign).

Confirm password Info

••••••••

In the **DB instance class** section, enable **Include previous generation classes**, and set these values:

- **Burstable classes (includes t classes)**
- **db.t2.micro**



Instance configuration

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

DB instance class Info

○ Standard classes (includes m classes)

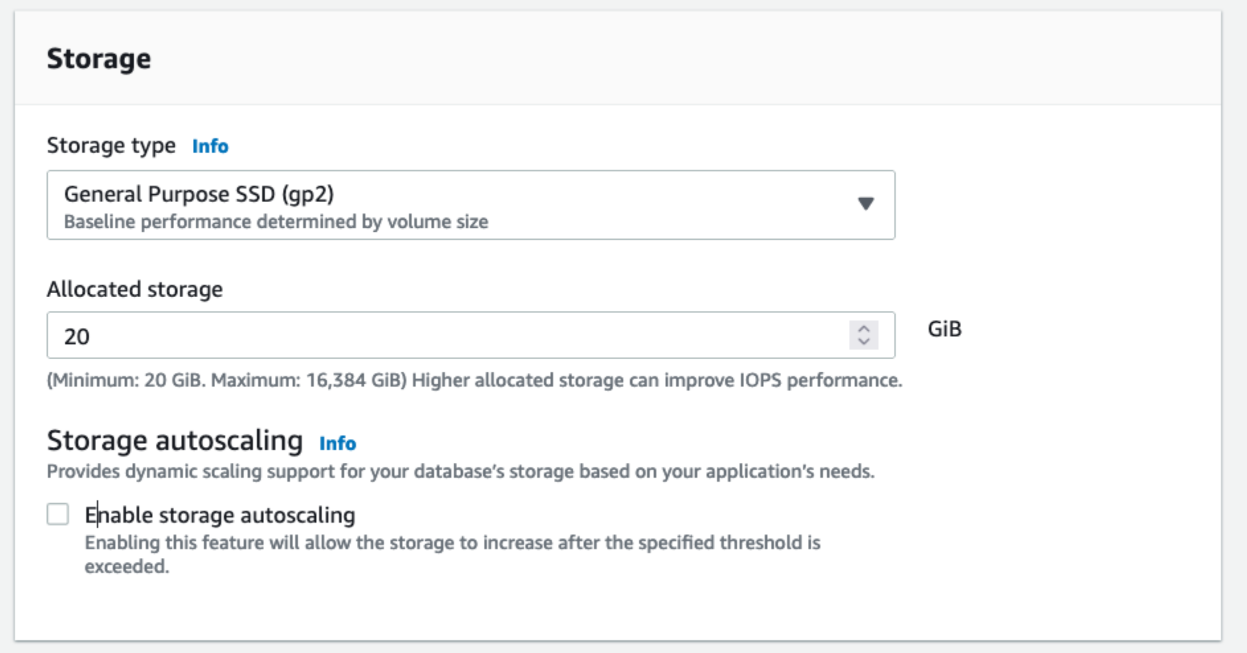
○ Memory optimized classes (includes r and x classes)

● Burstable classes (includes t classes)

db.t2.micro
1 vCPUs 1 GiB RAM Not EBS Optimized

Include previous generation classes

- In the Storage and Availability & durability sections, use these values.



Storage

Storage type Info

General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage

20 GiB

(Minimum: 20 GiB. Maximum: 16,384 GiB) Higher allocated storage can improve IOPS performance.

Storage autoscaling Info

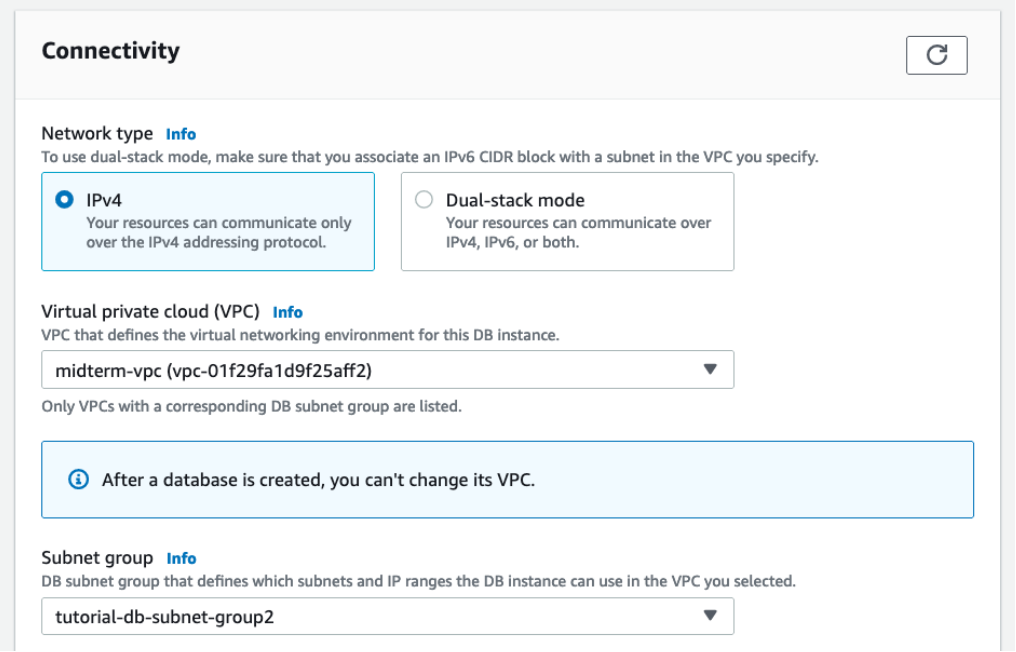
Provides dynamic scaling support for your database's storage based on your application's needs.

□ Enable storage autoscaling
Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

- **Storage type: General Purpose SSD (gp2)**
- **Allocated storage: 20**
- **Disable storage autoscaling**

- In the Connectivity section, set these values:
  Network Type: **IPv4**
- Virtual private cloud (VPC) – Choose an existing VPC with both public and private subnets, such as the **midterm-vpc** (vpc-identifier) created.

- Subnet group – The DB subnet group for the VPC, such as the **tutorial-db-subnet-group** you created.



Connectivity

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

VPC that defines the virtual networking environment for this DB instance.

midterm-vpc (vpc-01f29fa1d9f25aff2)

Only VPCs with a corresponding DB subnet group are listed.

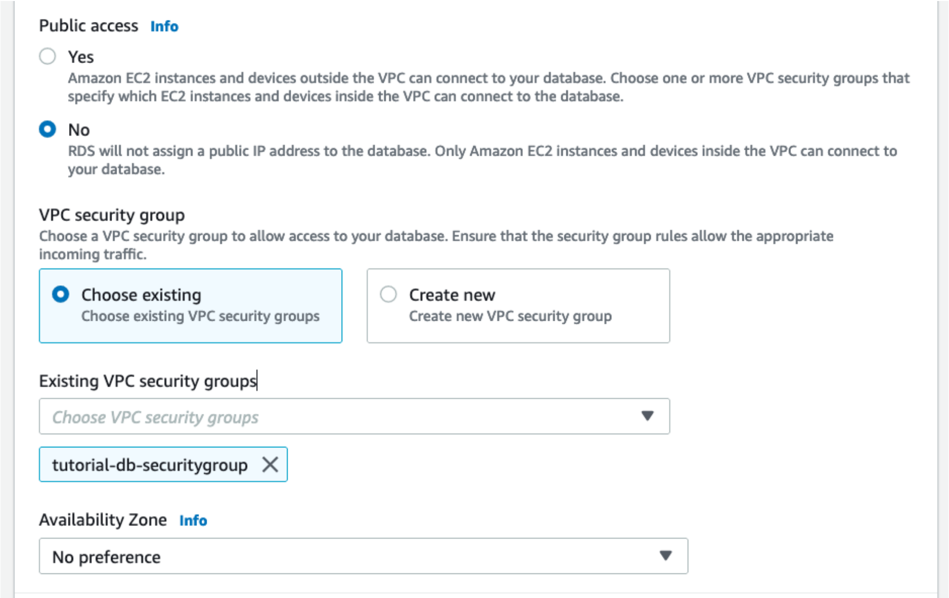
After a database is created, you can't change its VPC.

Subnet group Info

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

tutorial-db-subnet-group2

- Public access – No
- VPC security group – Choose existing
- Existing VPC security groups – Choose an existing VPC security group that is configured for private access, such as the **tutorial-db-securitygroup** you created.
- Remove other security groups, such as the default security group, by choosing the X associated with each.
- Availability Zone – No preference



Public access Info

○ Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

● No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

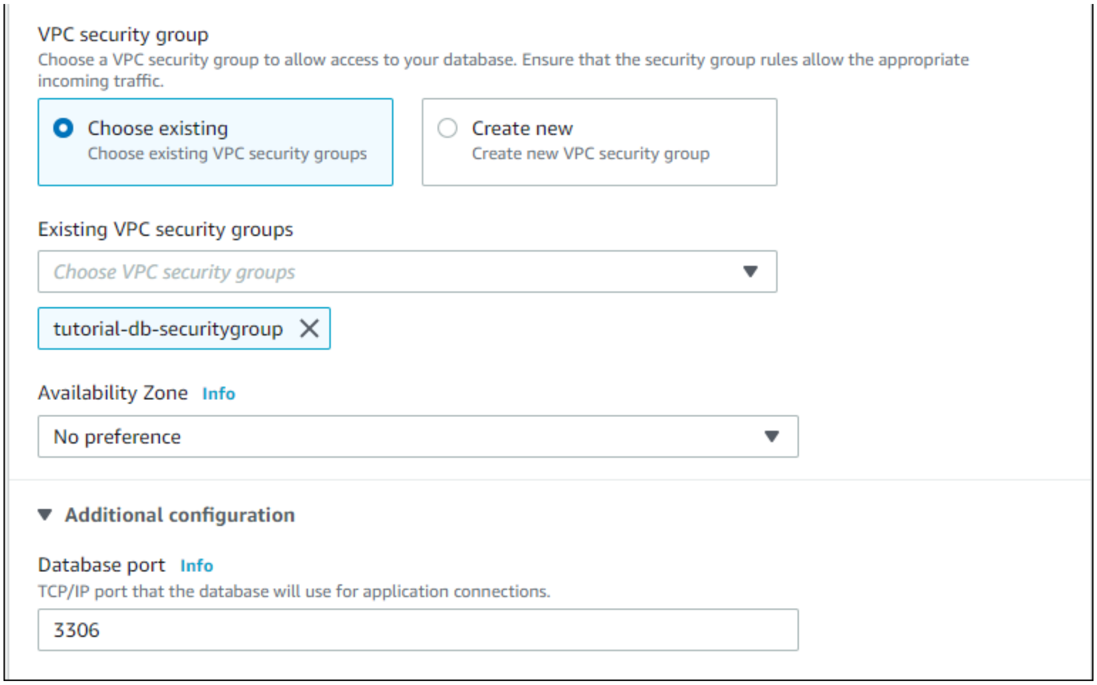
● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

Existing VPC security groups
Choose VPC security groups
tutorial-db-securitygroup ✕

Availability Zone Info
No preference

- Open Additional configuration, and make sure the Database port uses the default value 3306.



VPC security group
Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

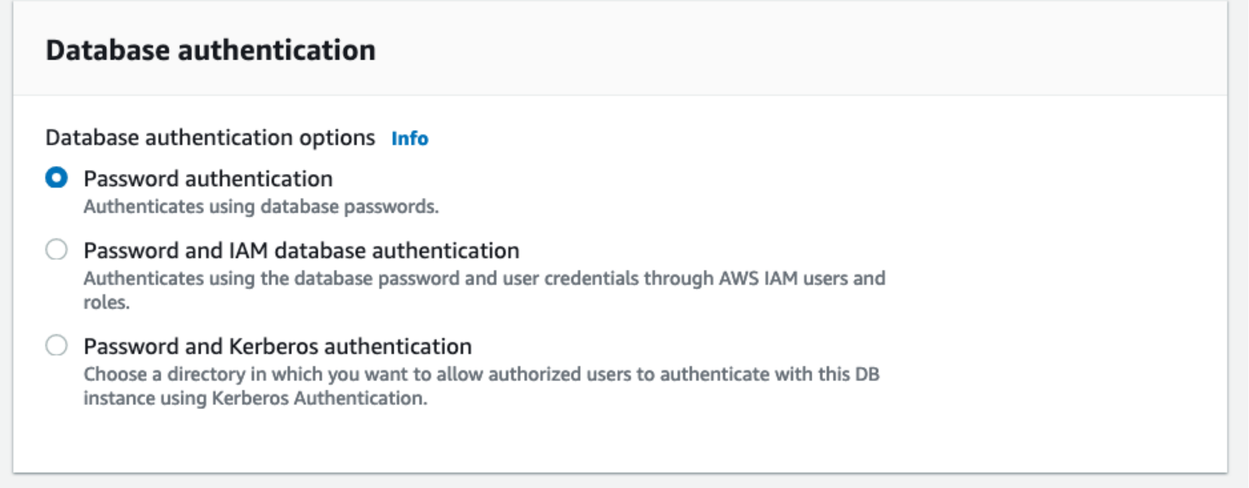
Existing VPC security groups
Choose VPC security groups
tutorial-db-securitygroup ✕

Availability Zone Info
No preference

▼ Additional configuration

Database port Info
TCP/IP port that the database will use for application connections.
3306

1. In the **Database authentication** section, make sure **Password authentication** is selected.



Database authentication

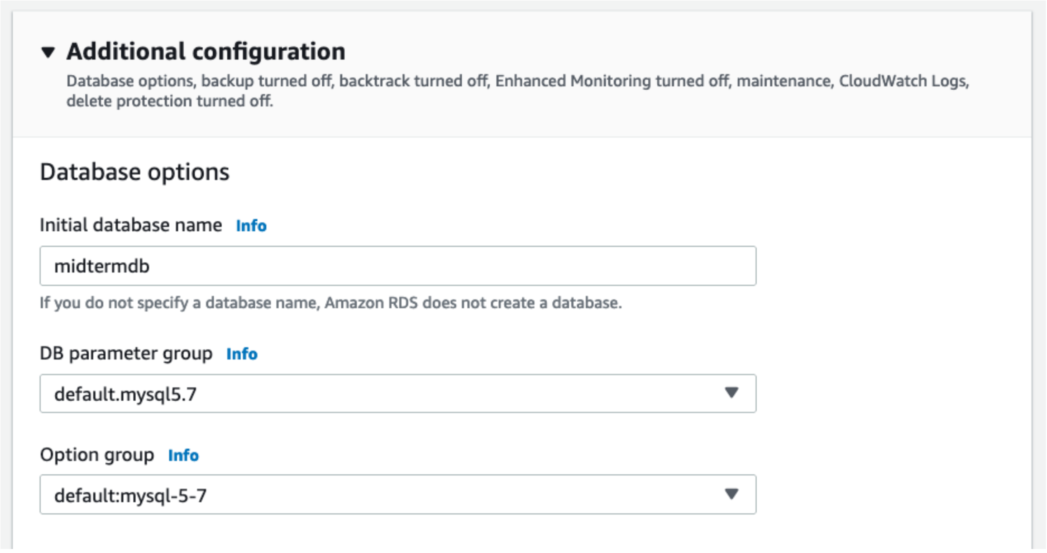
Database authentication options Info

● Password authentication
Authenticates using database passwords.

○ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

○ Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

2. Open the **Additional configuration** section, and enter **midtermdb** for **Initial database name**. Keep the default settings for the other options.



▼ **Additional configuration**
Database options, backup turned off, backtrack turned off, Enhanced Monitoring turned off, maintenance, CloudWatch Logs, delete protection turned off.

Database options

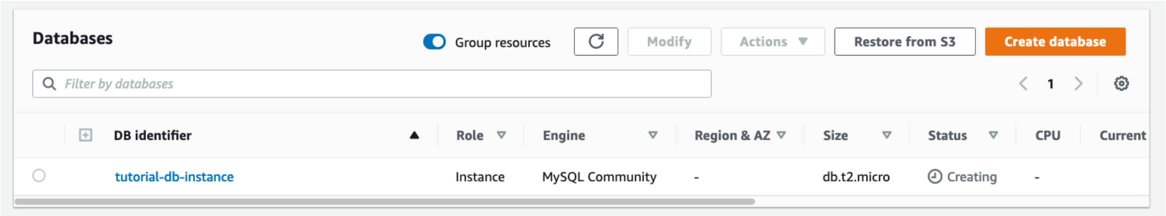
Initial database name Info
midtermdb
If you do not specify a database name, Amazon RDS does not create a database.

DB parameter group Info
default.mysql5.7 ▼

Option group Info
default:mysql-5-7 ▼

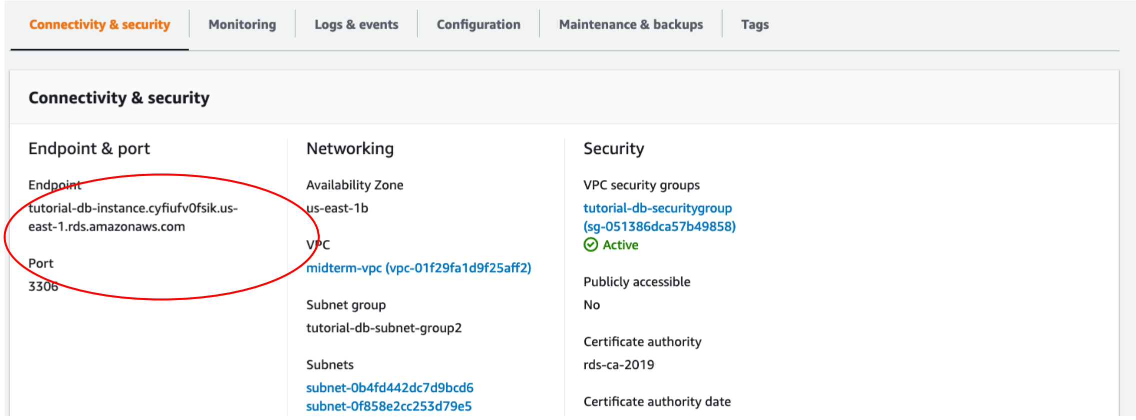
3. To create your MySQL DB instance, choose **Create database**.

Your new DB instance appears in the **Databases** list with the status **Creating**.



	DB identifier ▲	Role	Engine	Region & AZ	Size	Status	CPU	Current
○	tutorial-db-instance	Instance	MySQL Community	-	db.t2.micro	Creating	-	

4. Wait for the **Status** of your new DB instance to show as **Available**. Then choose the DB instance name to show its details.
5. In the **Connectivity & security** section, view the **Endpoint** and **Port** of the DB instance.



Connectivity & security

Endpoint & port

Endpoint
tutorial-db-instance.cyfiufv0fsik.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1b

VPC
midterm-vpc (vpc-01f29fa1d9f25aff2)

Subnet group
tutorial-db-subnet-group2

Subnets
subnet-0b4fd442dc7d9bcd6
subnet-0f858e2cc253d79e5

Security

VPC security groups
tutorial-db-securitygroup (sg-051386dca57b49858)
✓ Active

Publicly accessible
No

Certificate authority
rds-ca-2019

Certificate authority date

Note the endpoint and port for your DB instance. You use this information to connect your web server to your DB instance.

Now Create an EC2 instance and install a web server

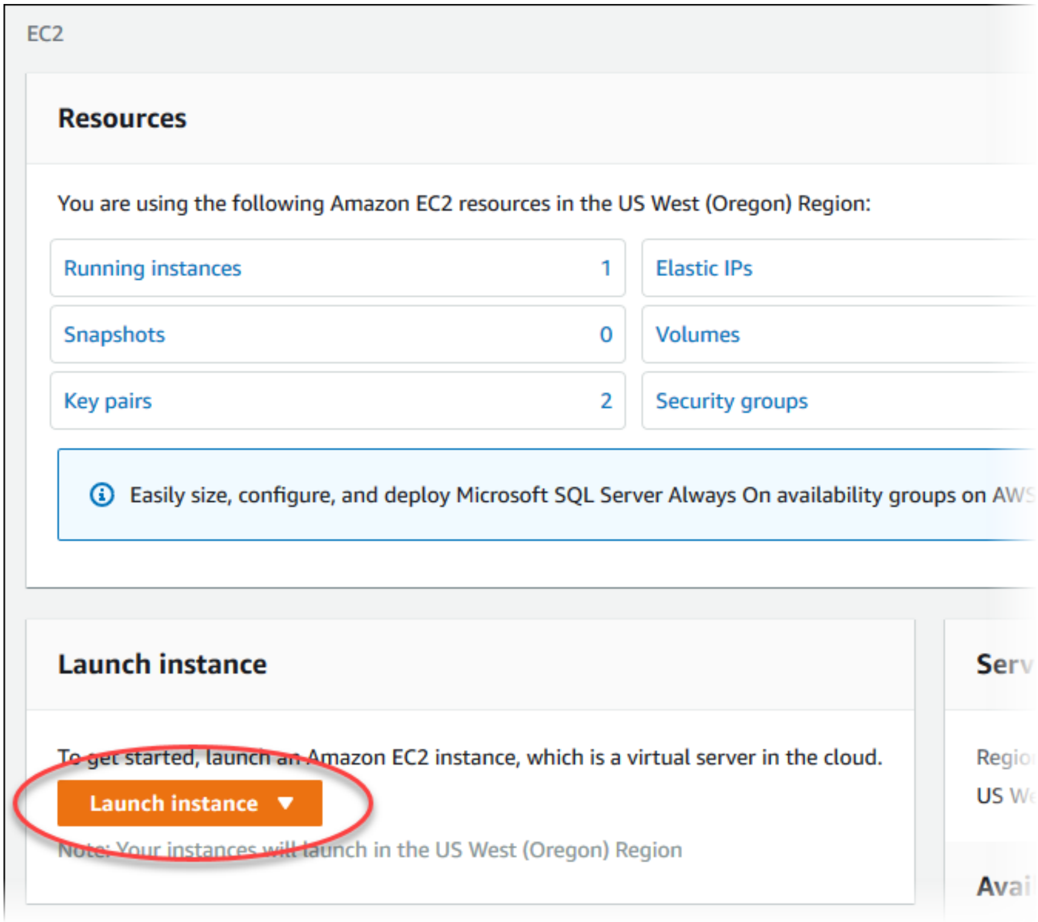
In this step, you create a web server to connect to the Amazon RDS DB instance that you created in **Create a DB instance.**

Launch an EC2 instance

First, you create an Amazon EC2 instance in the public subnet of your VPC.

To launch an EC2 instance

1. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/
2. Choose EC2 Dashboard, and then choose Launch instance, as shown below.



EC2

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) Region:

Running instances	1	Elastic IPs
Snapshots	0	Volumes
Key pairs	2	Security groups

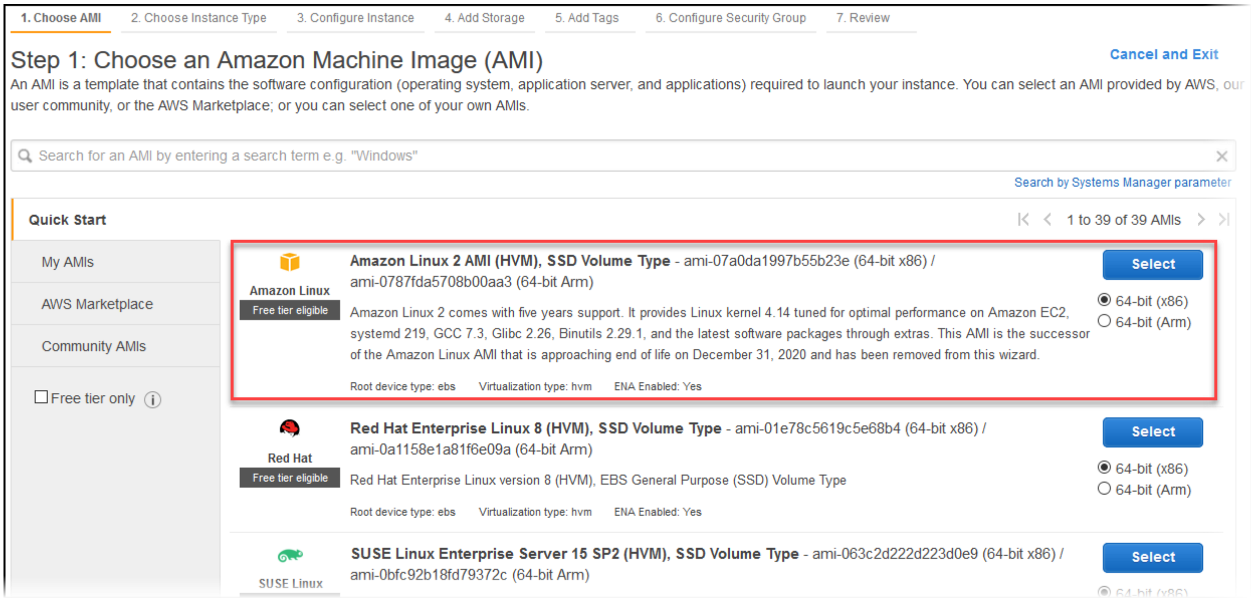
ⓘ Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Note: Your instances will launch in the US West (Oregon) Region

3. Choose the Amazon Linux 2 AMI.



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Amazon Linux — Free tier eligible

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-07a0da1997b55b23e (64-bit x86) / ami-0787fda5708b00aa3 (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 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 approaching end of life on December 31, 2020 and has been removed from this wizard.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select ● 64-bit (x86) ○ 64-bit (Arm)

Red Hat — Free tier eligible

Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-01e78c5619c5e68b4 (64-bit x86) / ami-0a1158e1a81f6e09a (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

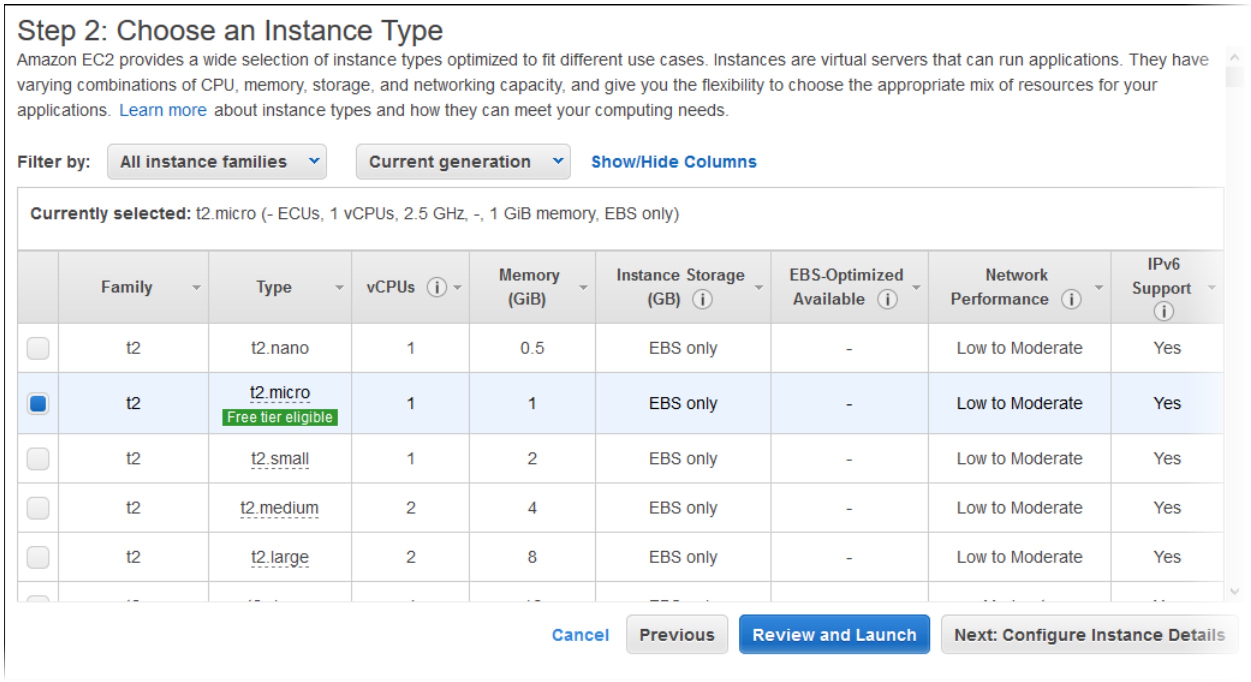
Select ● 64-bit (x86) ○ 64-bit (Arm)

SUSE Linux

SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type - ami-063c2d222d223d0e9 (64-bit x86) / ami-0bfc92b18fd79372c (64-bit Arm)

Select

4. Choose the t2.micro instance type, as shown following, and then choose Next: Configure Instance Details.



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

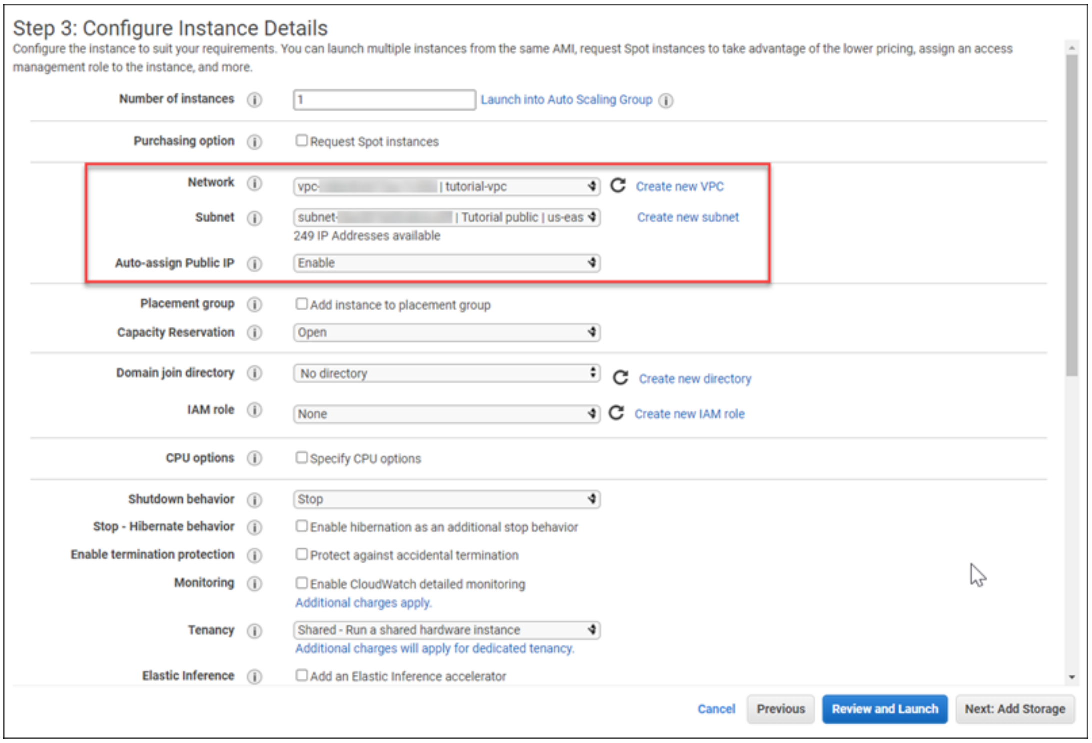
Filter by: All instance families · Current generation · Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
☐	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
☐	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
☐	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
☐	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes

5. On the Configure Instance Details page, shown following, set these values and keep the other values as their defaults:

   - Network: Choose the VPC with both public and private subnets that you chose for the DB instance, such as the vpc-identifier | **midterm-vpc.**
   - Subnet: Choose an existing public subnet, such as subnet-identifier | **Tutorial public | us-east-1a**
   - Auto-assign Public IP: **Choose Enable.**



Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 — Launch into Auto Scaling Group

Purchasing option: □ Request Spot instances

Network: vpc-[redacted] | tutorial-vpc — Create new VPC

Subnet: subnet-[redacted] | Tutorial public | us-eas — Create new subnet
249 IP Addresses available

Auto-assign Public IP: Enable

Placement group: □ Add instance to placement group

Capacity Reservation: Open

Domain join directory: No directory — Create new directory

IAM role: None — Create new IAM role

CPU options: □ Specify CPU options

Shutdown behavior: Stop

Stop - Hibernate behavior: □ Enable hibernation as an additional stop behavior

Enable termination protection: □ Protect against accidental termination

Monitoring: □ Enable CloudWatch detailed monitoring
Additional charges apply.

Tenancy: Shared - Run a shared hardware instance
Additional charges will apply for dedicated tenancy.

Elastic Inference: □ Add an Elastic Inference accelerator

Attach the Role you created at the beginning of this assignment.

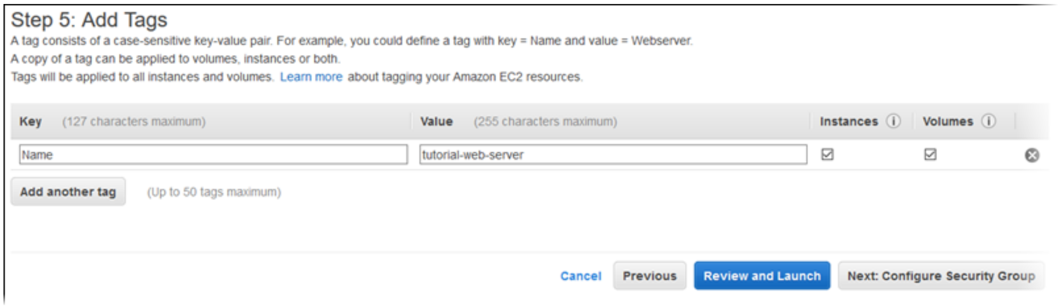
Or you can attach it after the instance is created.

The name of the role is: **midterm-ec2-s3-role**

Add the following script to user-data:

```
#!/bin/bash
sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
sudo yum install -y httpd
sudo systemctl start httpd
sudo systemctl enable httpd
```

- Choose Next: Add Storage.
- On the Add Storage page, keep the default values and choose Next: Add Tags.
- On the Add Tags page, shown following, choose Add Tag, then enter Name for Key and enter **tutorial-web-server** for Value.



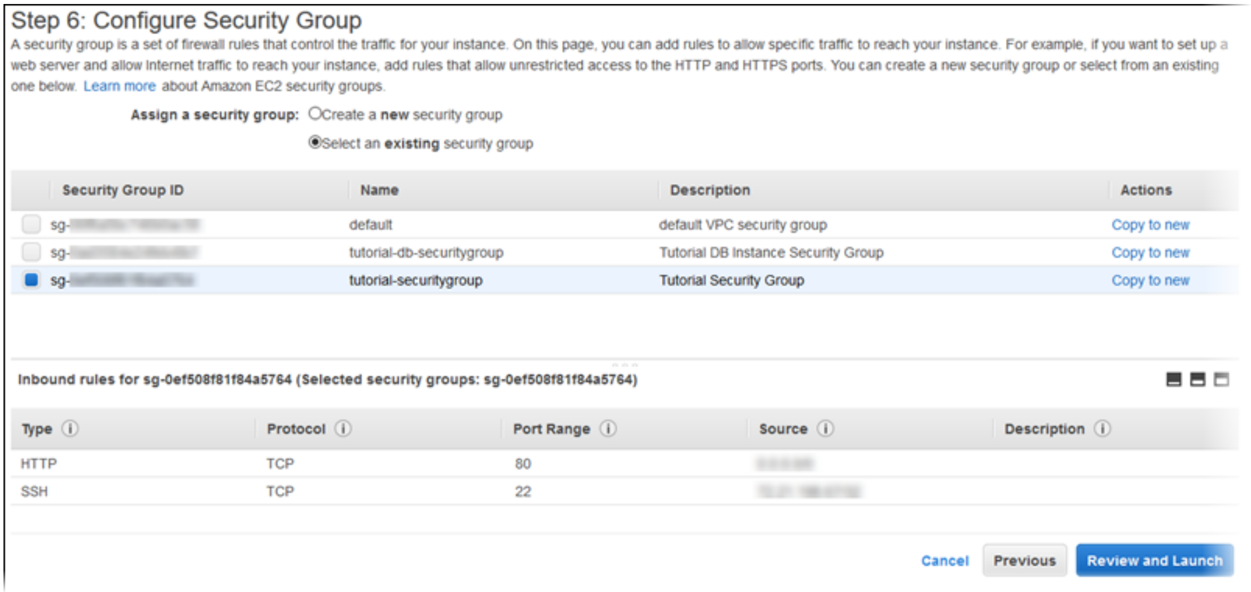
Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)	Instances	Volumes	
Name	tutorial-web-server	☑	☑	⊗

- Choose Next: Configure Security Group.

On the Configure Security Group page, shown following, choose Select an existing security group. Then choose an existing security group, such as the **tutorial-securitygroup** created before. Make sure that the security group that you choose includes inbound rules for Secure Shell (SSH) and HTTP access.



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ○ Create a **new** security group
● Select an **existing** security group

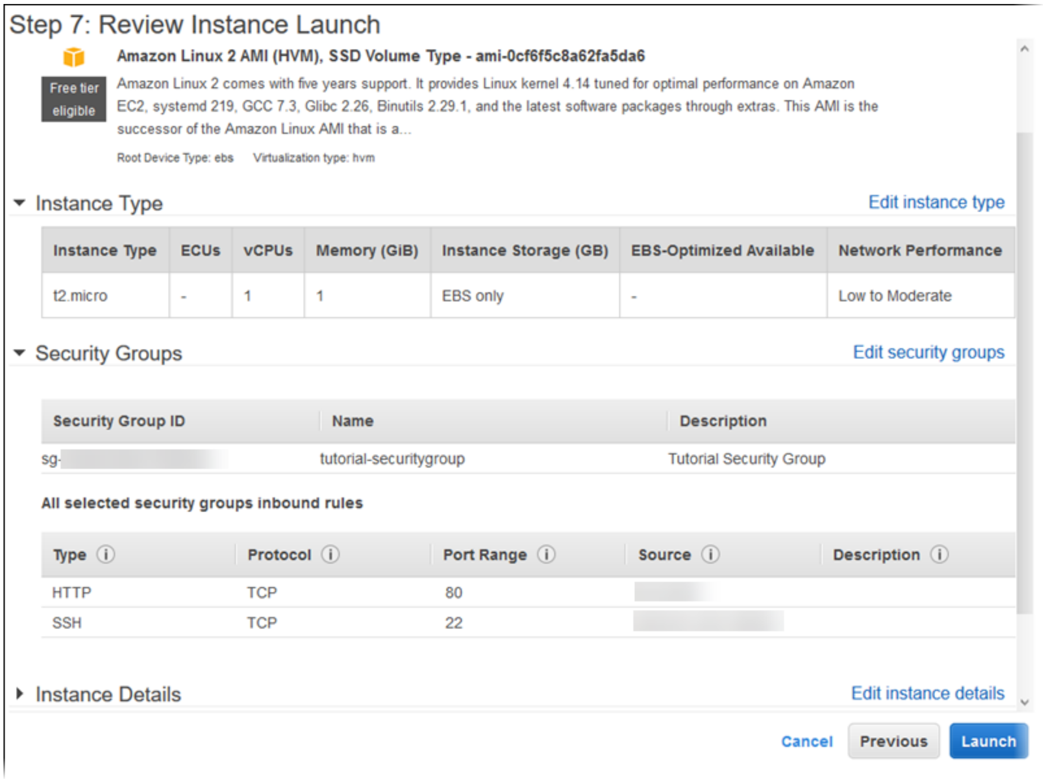
	Security Group ID	Name	Description	Actions
□	sg-[redacted]	default	default VPC security group	Copy to new
□	sg-[redacted]	tutorial-db-securitygroup	Tutorial DB Instance Security Group	Copy to new
■	sg-[redacted]	tutorial-securitygroup	Tutorial Security Group	Copy to new

Inbound rules for sg-0ef508f81f84a5764 (Selected security groups: sg-0ef508f81f84a5764)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	[redacted]	
SSH	TCP	22	[redacted]	

- Choose Review and Launch.

On the Review Instance Launch page, shown following, verify your settings and then choose Launch.



Step 7: Review Instance Launch

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0cf6f5c8a62fa5da6

Free tier eligible

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 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 a...

Root Device Type: ebs   Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

Security Groups

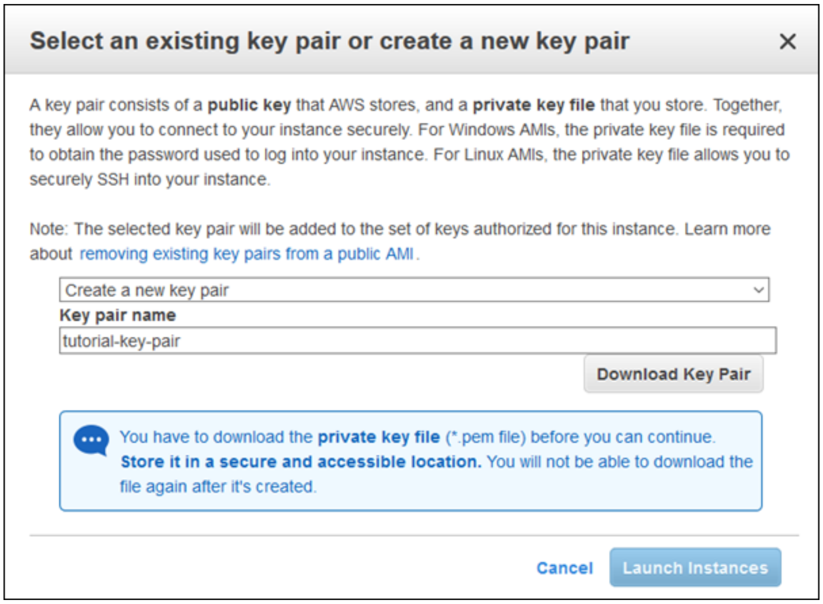
Security Group ID	Name	Description
sg-[redacted]	tutorial-securitygroup	Tutorial Security Group

All selected security groups inbound rules

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	[redacted]	
SSH	TCP	22	[redacted]	

Instance Details

- On the Select an existing key pair or create a new key pair page, shown following, choose Create a new key pair and set Key pair name to tutorial-key-pair. Choose Download Key Pair, and then save the key pair file on your local machine. You use this key pair file to connect to your EC2 instance.



Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

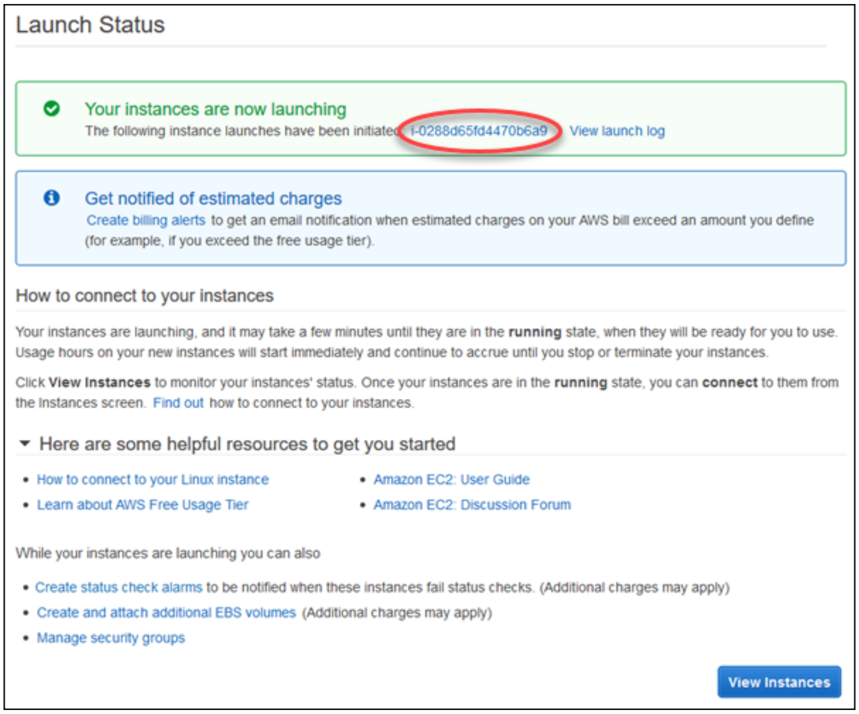
Create a new key pair

Key pair name

tutorial-key-pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

- To launch your EC2 instance, choose Launch Instances. On the Launch Status page, shown following, note the identifier for your new EC2 instance, for example: i-0288d65fd4470b6a9.



Launch Status

Your instances are now launching

The following instance launches have been initiated: i-0288d65fd4470b6a9 View launch log

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click **View Instances** to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. Find out how to connect to your instances.

Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2: User Guide
- Amazon EC2: Discussion Forum

While your instances are launching you can also

- Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)
- Create and attach additional EBS volumes (Additional charges may apply)
- Manage security groups

- Choose View Instances to find your instance.

- Wait until Instance Status for your instance reads as Running before continuing.

Install an Apache web server with PHP

1. Connect to the EC2 instance that you created earlier.

   To allow ec2-user to manage files in the default root directory for your Apache web server, modify the ownership and permissions of the /var/www directory. There are many ways to accomplish this task. In this tutorial, you add ec2-user to the apache group, to give the apache group ownership of the /var/www directory and assign write permissions to the group.

To set file permissions for the Apache web server

- Add the ec2-user user to the apache group.

```
sudo usermod -a -G apache ec2-user
```

- Log out to refresh your permissions and include the new apache group.

```
exit
```

- Log back in again and verify that the apache group exists with the groups command.

```
groups
```

- Your output looks similar to the following:

```
ec2-user adm wheel apache systemd-journal
```

- Change the group ownership of the /var/www directory and its contents to the apache group.

```
sudo chown -R ec2-user:apache /var/www
```

Change the directory permissions of /var/www and its subdirectories to add group write permissions and set the group ID on subdirectories created in the future.

```
sudo chmod 2775 /var/www

find /var/www -type d -exec sudo chmod 2775 {} \;
```

Recursively change the permissions for files in the /var/www directory and its subdirectories to add group write permissions.

```
find /var/www -type f -exec sudo chmod 0664 {} \;
```

Now, ec2-user (and any future members of the apache group) can add, delete, and edit files in the Apache document root, enabling you to add content, such as a static website or a PHP application.

Connect your Apache web server to your DB instance

Next, you add content to your Apache web server that connects to your Amazon RDS DB instance.

To add content to the Apache web server that connects to your DB instance

- While still connected to your EC2 instance, change the directory to /var/www and create a new subdirectory named inc.

```
cd /var/www

mkdir inc

cd inc
```

- Create a new file in the inc directory named dbinfo.inc, and then edit the file.

  ```
  vi dbinfo.inc
  ```

- Add the following contents to the dbinfo.inc file. Here, db_instance_endpoint is your DB instance endpoint, without the port, and master password is the master password for your DB instance.

  ```
  <?php

  define('DB_SERVER', 'db_instance_endpoint');
  define('DB_USERNAME', 'tutorial_user');
  define('DB_PASSWORD', 'master password');
  define('DB_DATABASE', 'sample');

  ?>
  ```

- Save and close the dbinfo.inc file.

- Change the directory to /var/www/html.

  ```
  cd /var/www/html
  ```

- **Download the SamplePage.php file from your s3 bucket using the CLI.**

Verify that your web server successfully connects to your DB instance by opening a web browser and browsing to **http://EC2 instance endpoint/SamplePage.php, or http://public-ip/SamplePage.php**

You can use SamplePage.php to add data to your DB instance. The data that you add is then displayed on the page.

After you have finished testing your web server and your database, you should delete your DB instance and your Amazon EC2 instance.