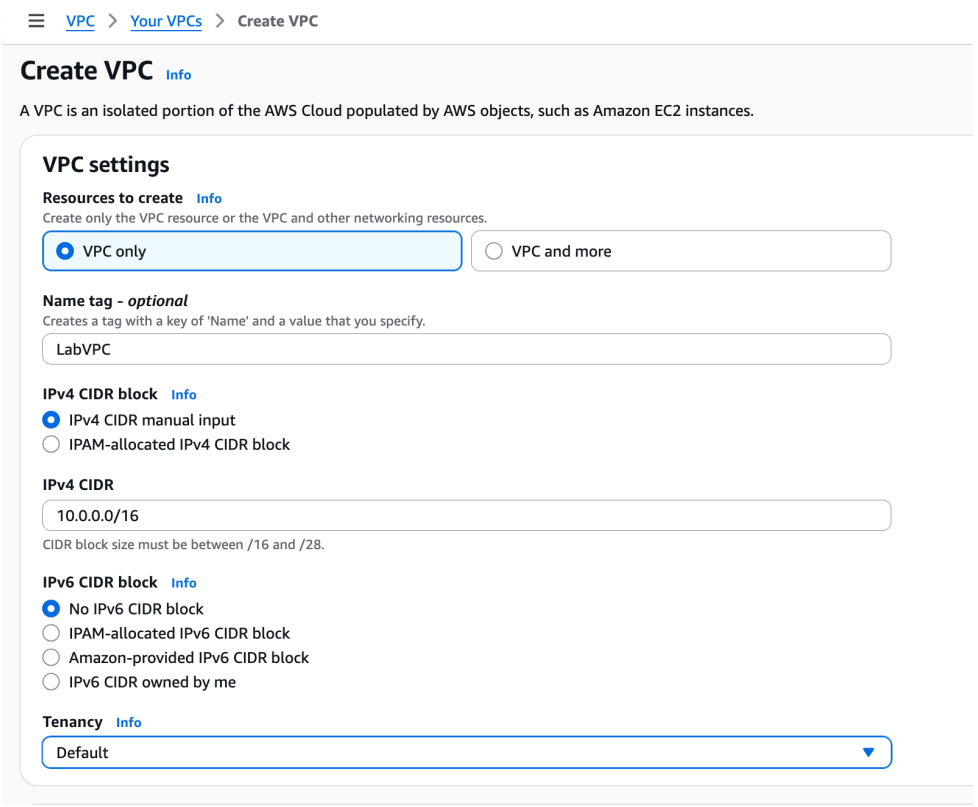
Sandbox Project: Two-tier VPC with Public EC2 and Private RDS (MySQL)

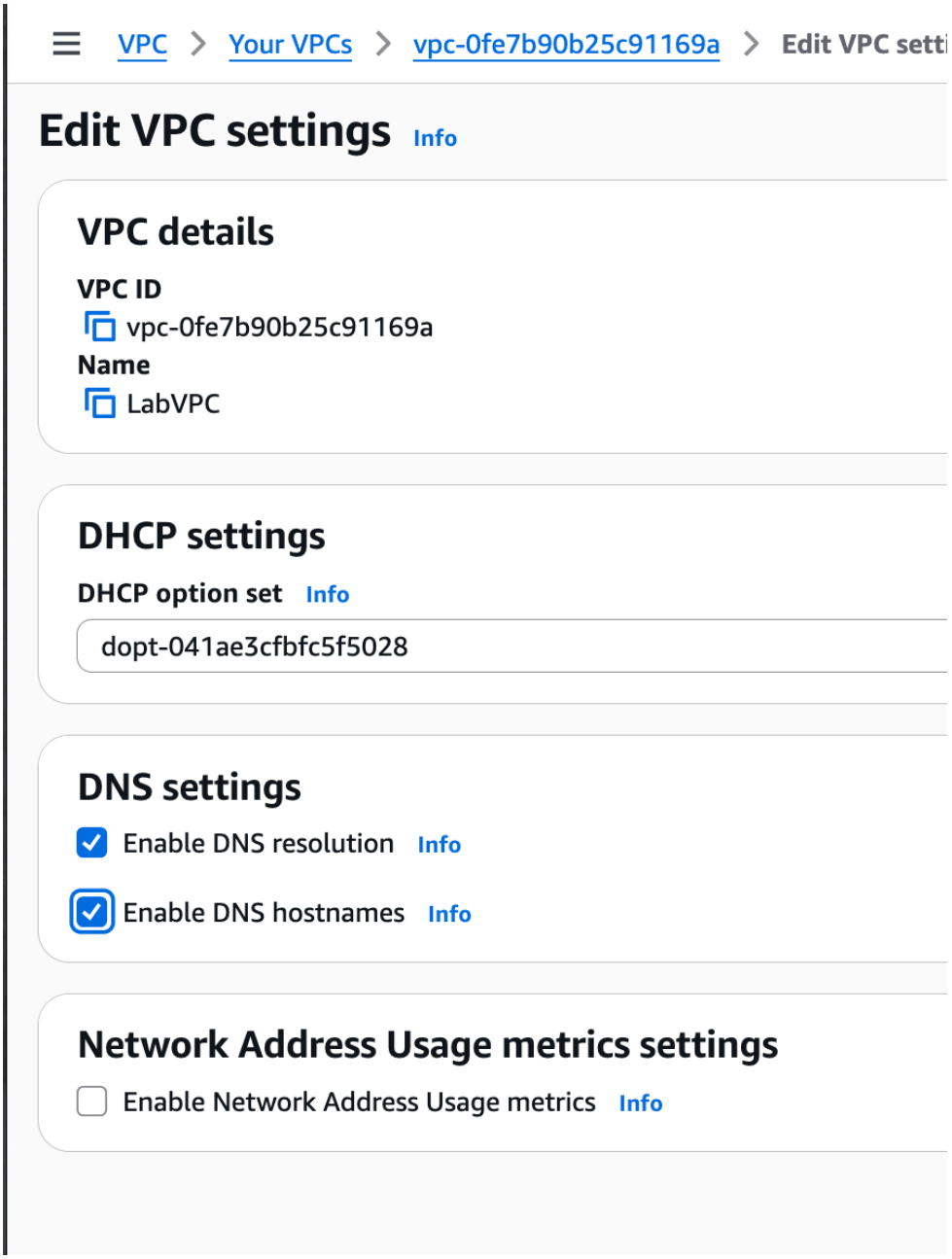
Prerequisites

- An AWS sandbox/lab account with permission to create VPC, EC2, RDS, IAM key pairs, and security groups
- A **key pair** to SSH into EC2 (create one during EC2 launch if you don't have it)
- Choose a **Region** (e.g., `us-east-1`) and stick to it throughout
- Your **public IP address** (for the EC2 SSH rule) — you can find it via an IP-check site if needed

> ❗ Cost guardrail: This build avoids a NAT Gateway (often the biggest surprise cost). Because your EC2 is in the public subnet (with an IGW) and RDS doesn't need outbound internet, a NAT GW is not required.

Step 1 — Create the VPC

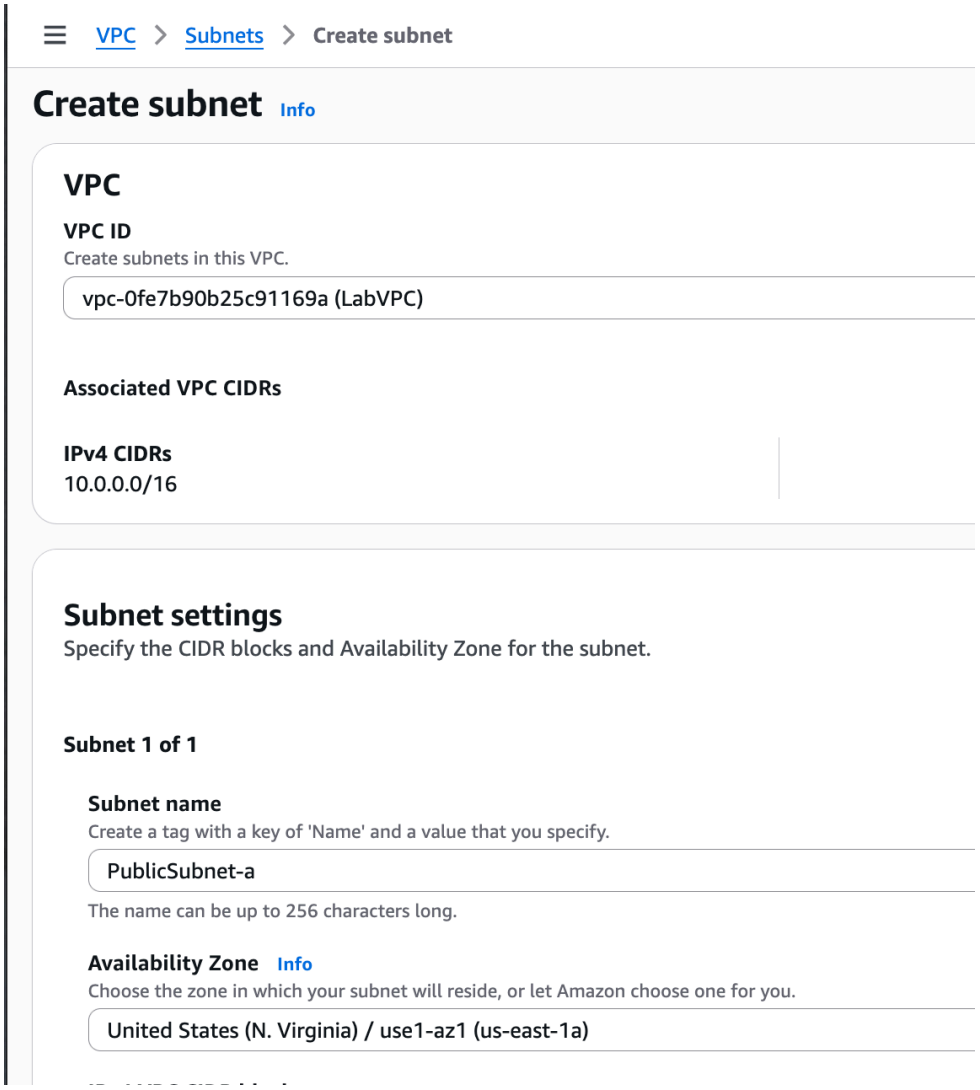

VPC > Your VPCs > Create VPC
Create VPC Info
A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.
VPC settings
Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
VPC only
VPC and more
Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.
LabVPC
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

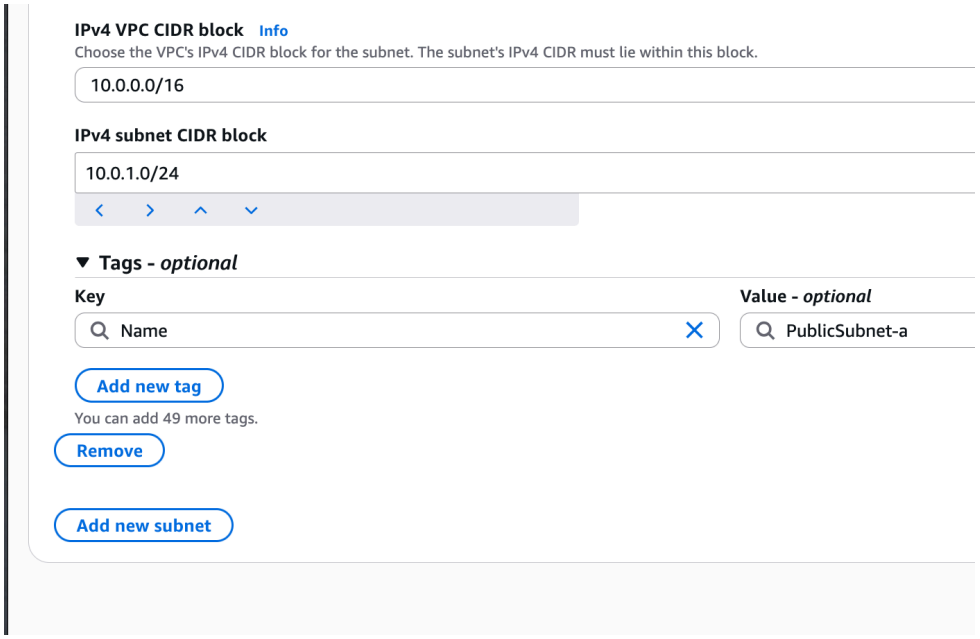


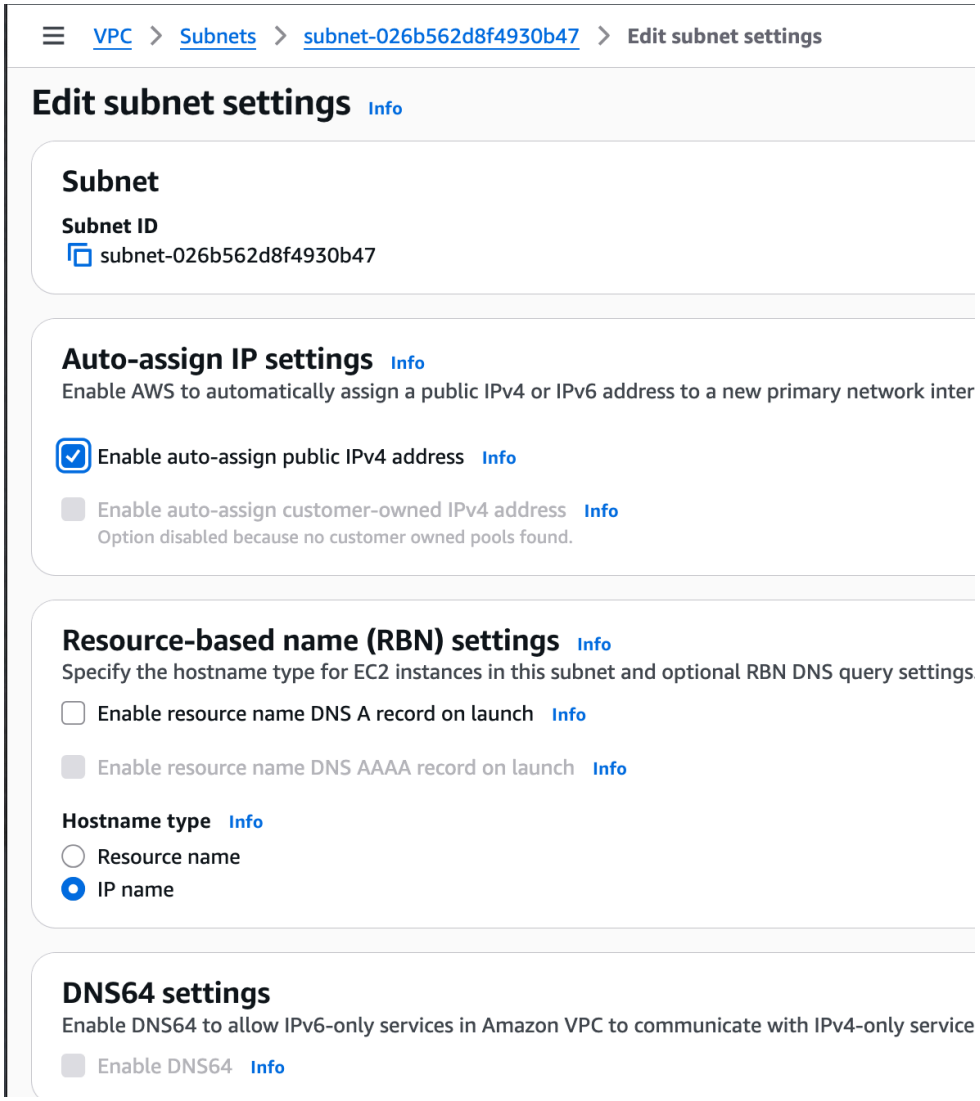
Step 2: Creating Subnets

Create **one public** and **two private** subnets (the second private is needed for the RDS subnet group).

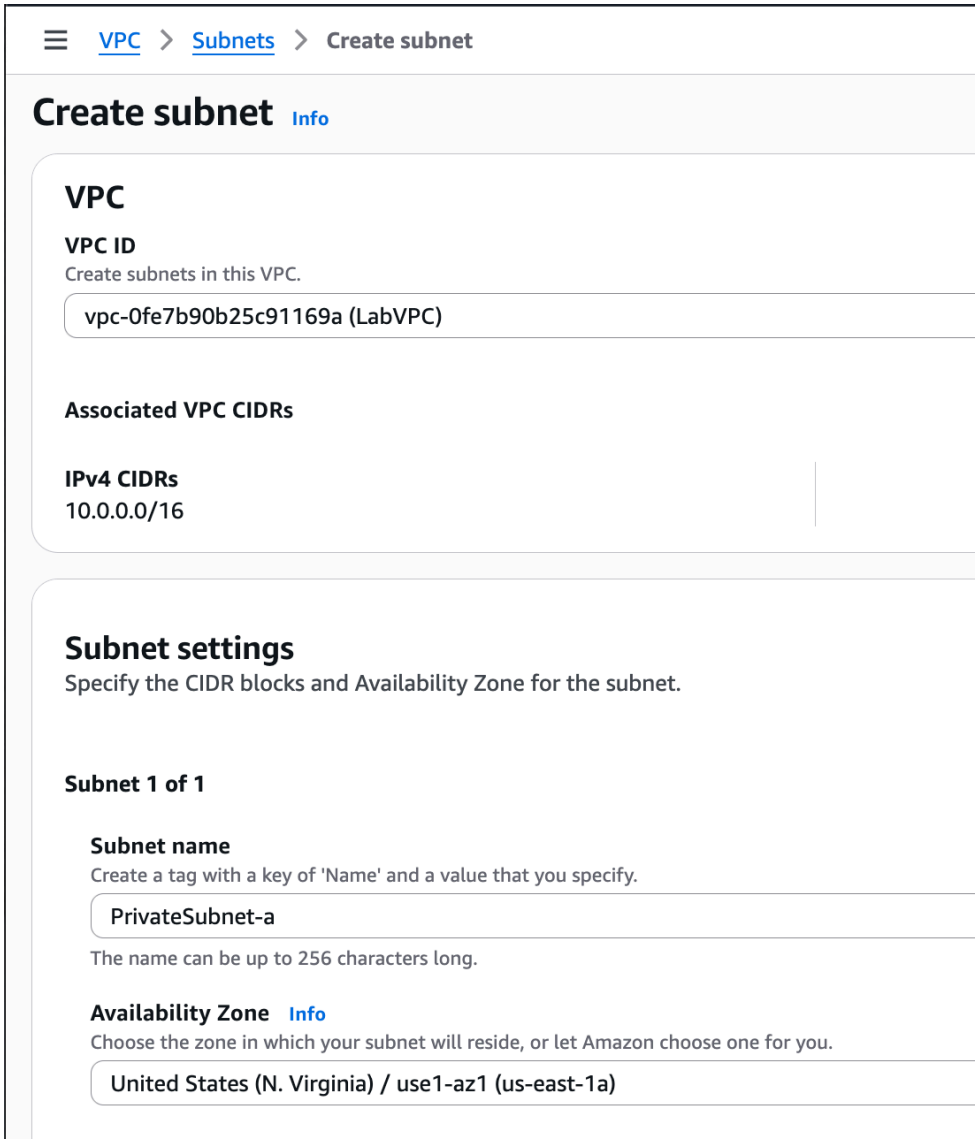
Public Subnet

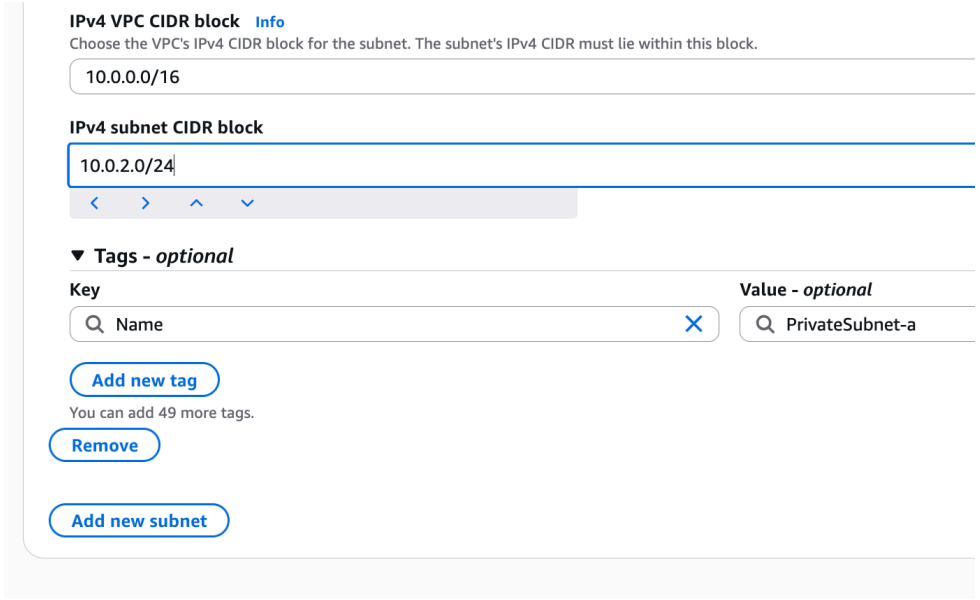

VPC > Subnets > Create subnet
Create subnet Info
VPC
VPC ID
Create subnets in this VPC.
vpc-0fe7b90b25c91169a (LabVPC)
Associated VPC CIDRs
IPv4 CIDRs
10.0.0.0/16
Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.
Subnet 1 of 1
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PublicSubnet-a
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.
United States (N. Virginia) / use1-az1 (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
Tags - optional
Key
Value - optional
Name
PublicSubnet-a
Add new tag
You can add 49 more tags.
Remove
Add new subnet



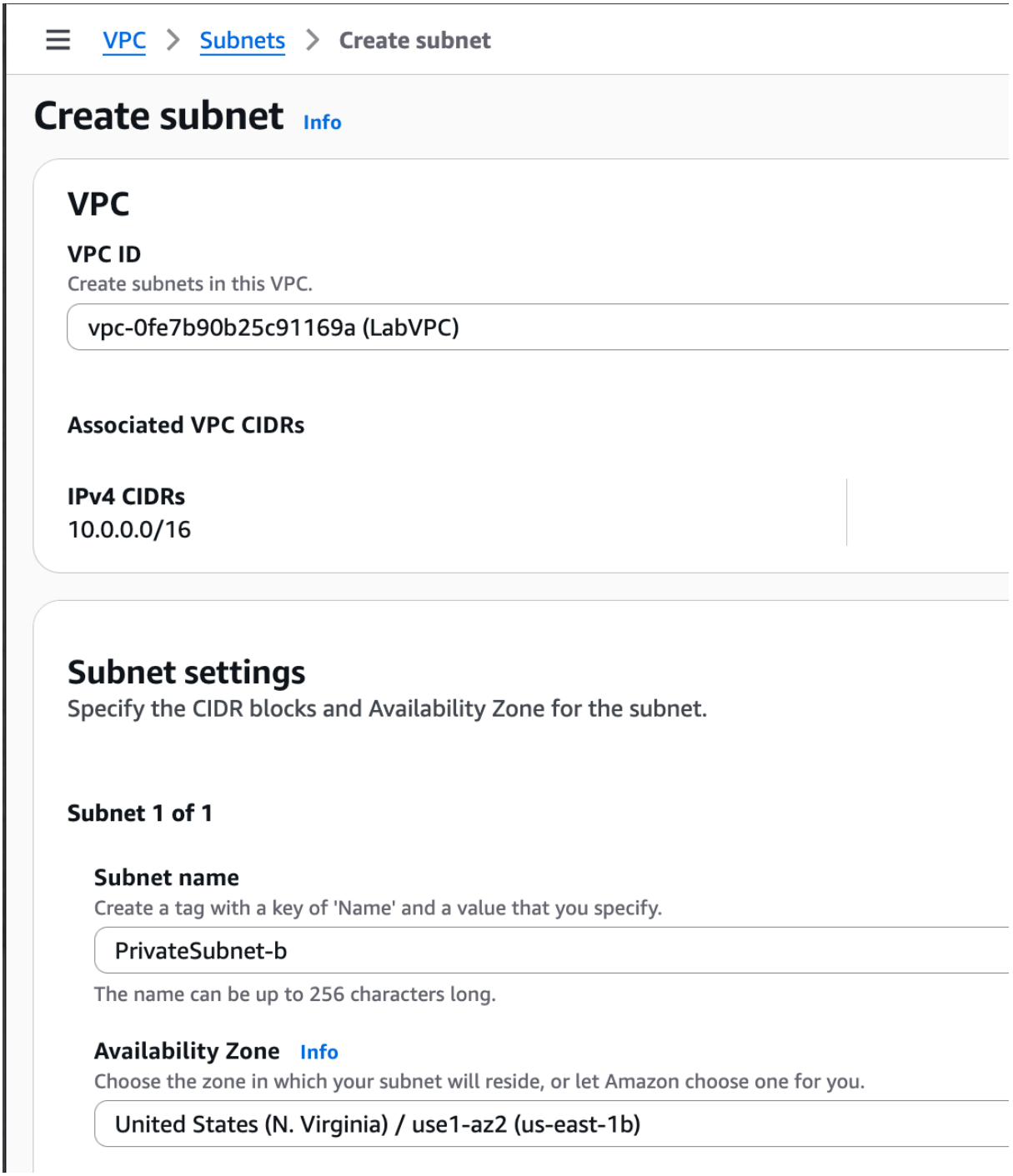
Private Subnet -a

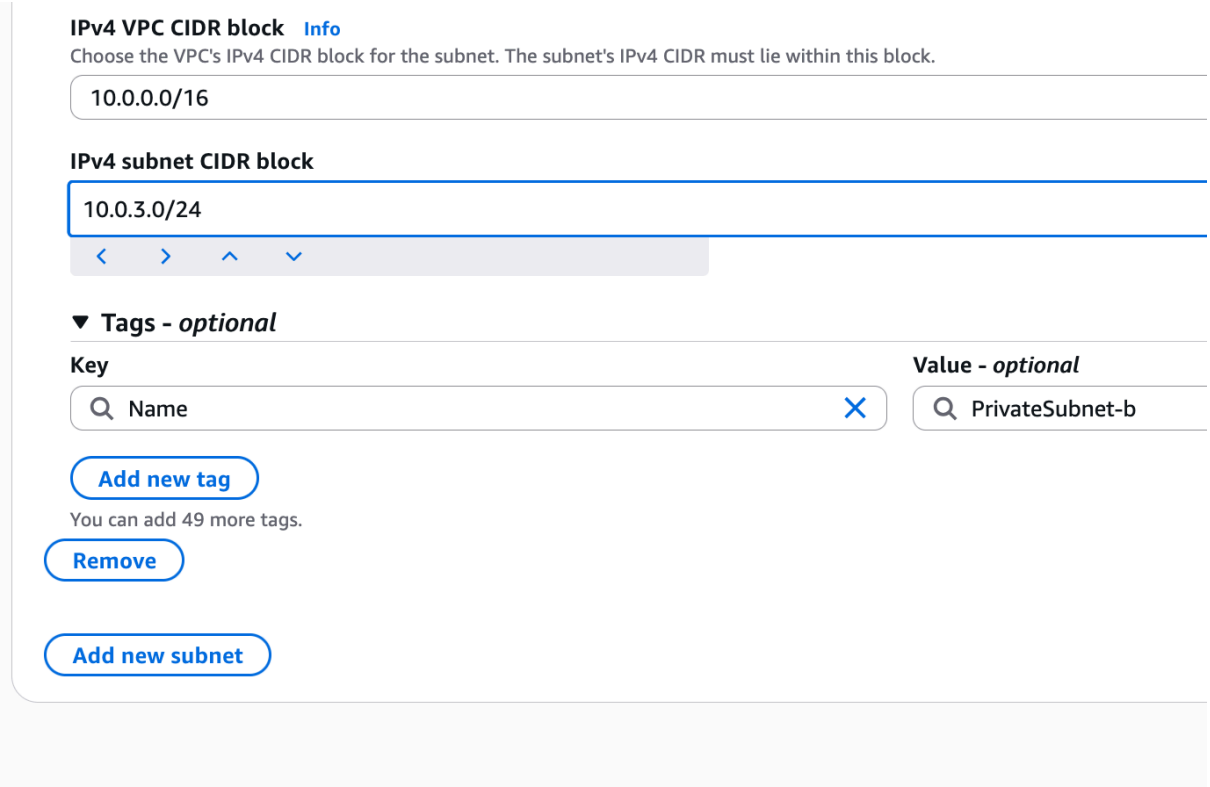

VPC > Subnets > Create subnet
Create subnet Info
VPC
VPC ID
Create subnets in this VPC.
vpc-0fe7b90b25c91169a (LabVPC)
Associated VPC CIDRs
IPv4 CIDRs
10.0.0.0/16
Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.
Subnet 1 of 1
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PrivateSubnet-a
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.
United States (N. Virginia) / use1-az1 (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
Tags - optional
Key
Name
Value - optional
PrivateSubnet-a
Add new tag
You can add 49 more tags.
Remove
Add new subnet

Private Subnet -b

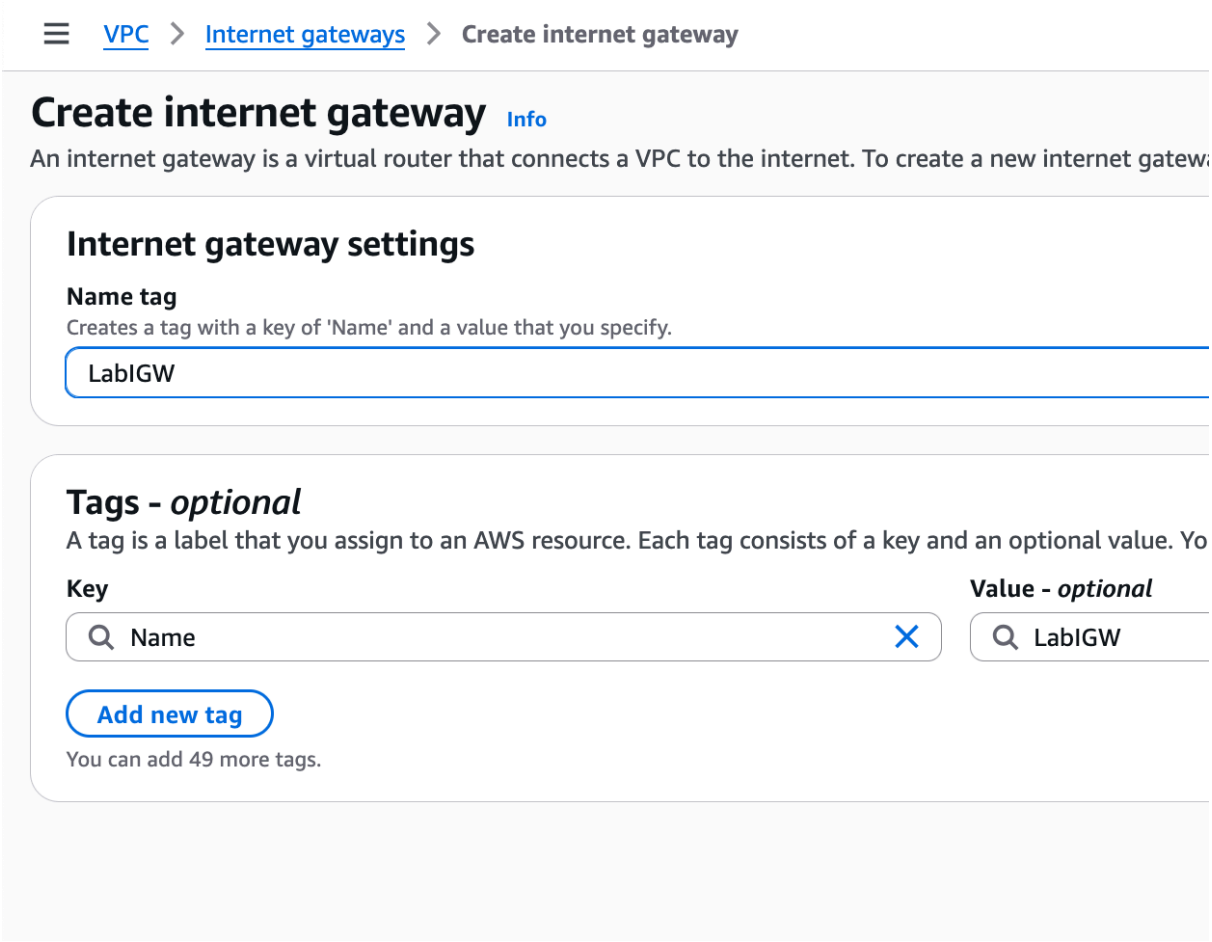
AZ is different than private subnet -a



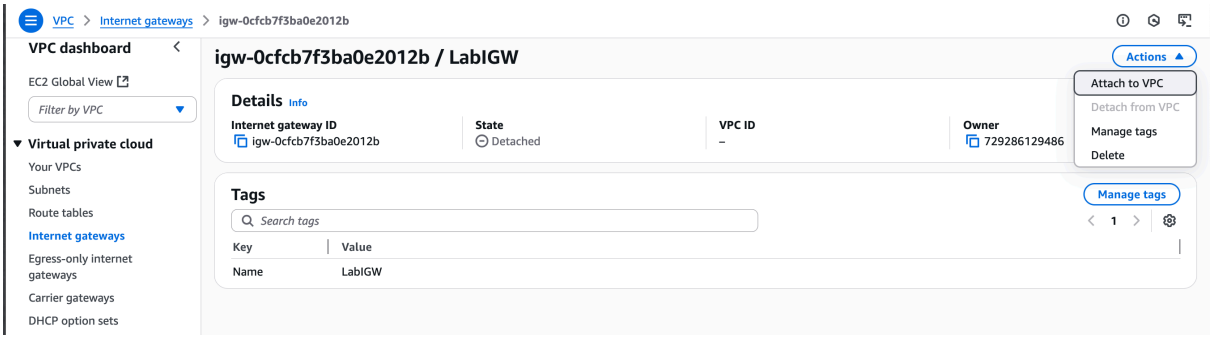


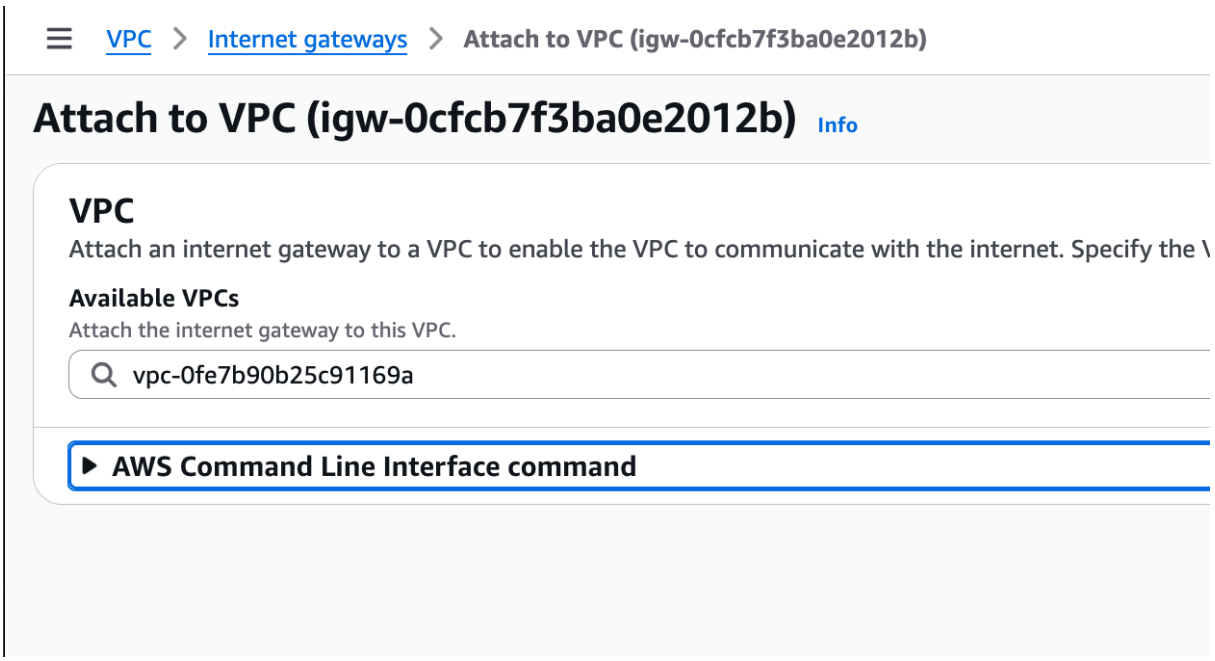
Step 3 — Internet Gateway (IGW)

The IGW provides internet egress/ingress for the public subnet.



Attach to VPC

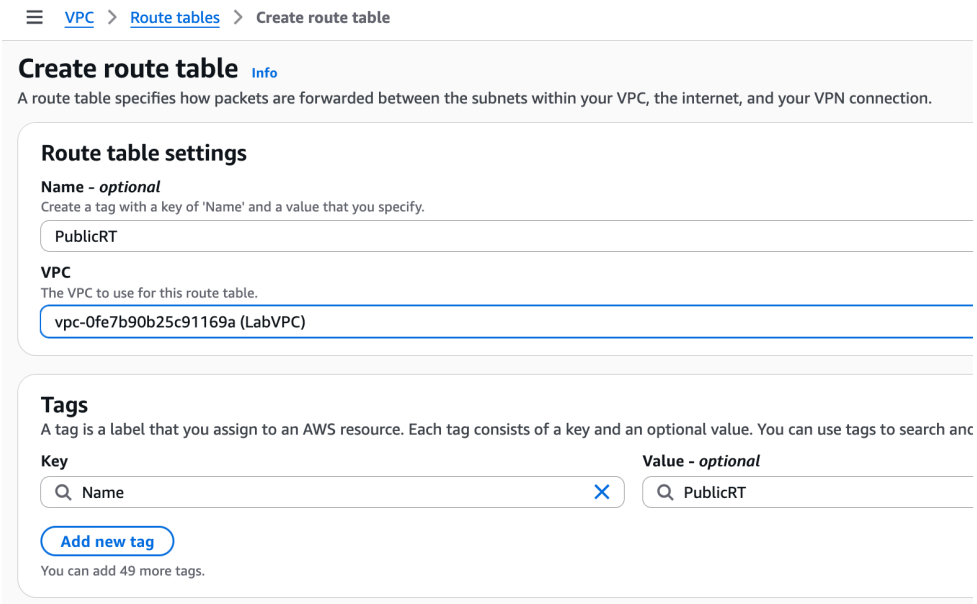




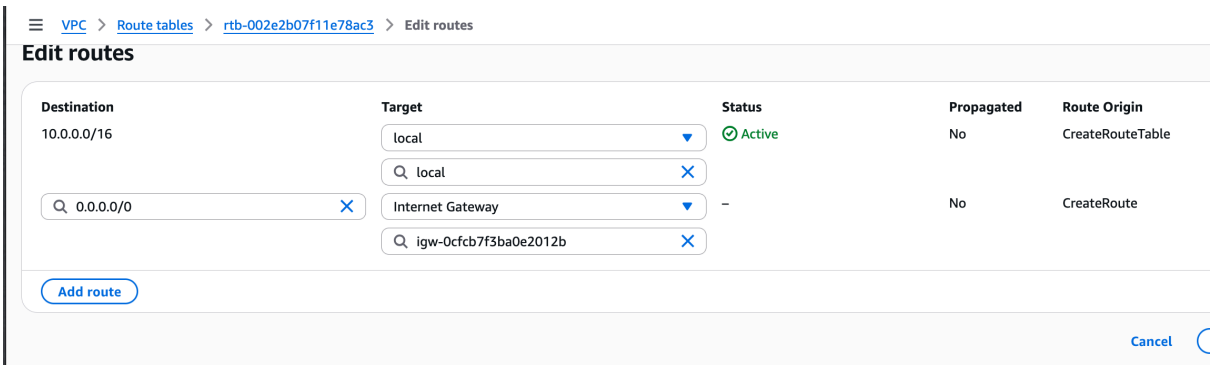
Step 4 — Route Tables

Create two route tables: one public (with IGW default route) and one private (local only).

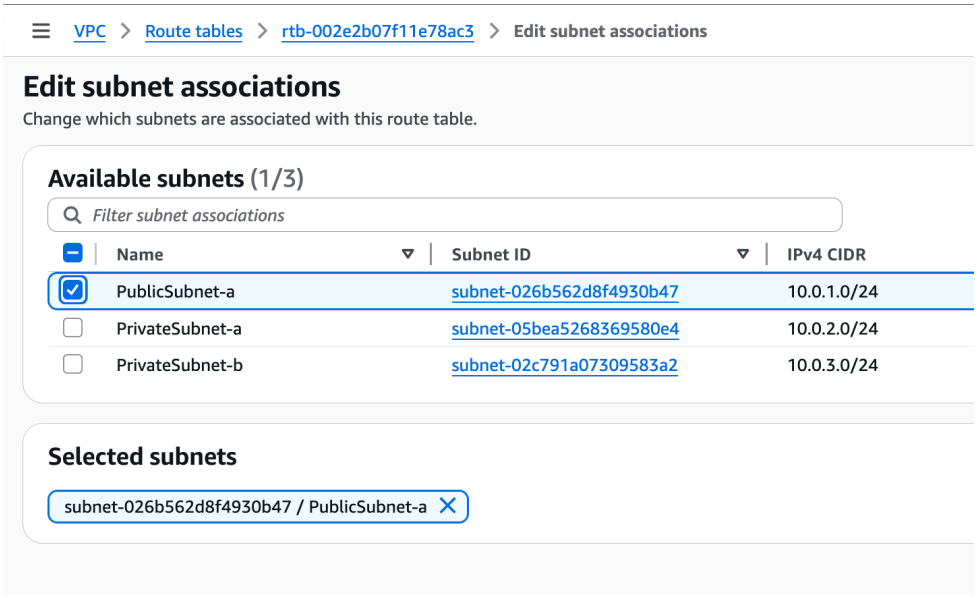
Public Route Table



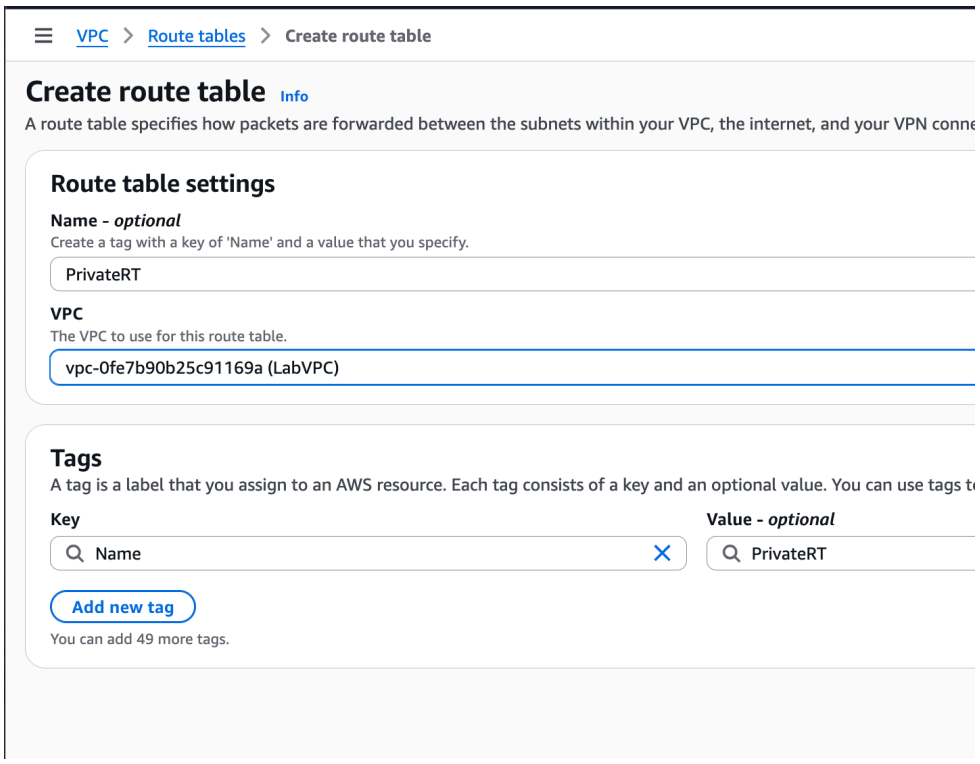
Edit routes.



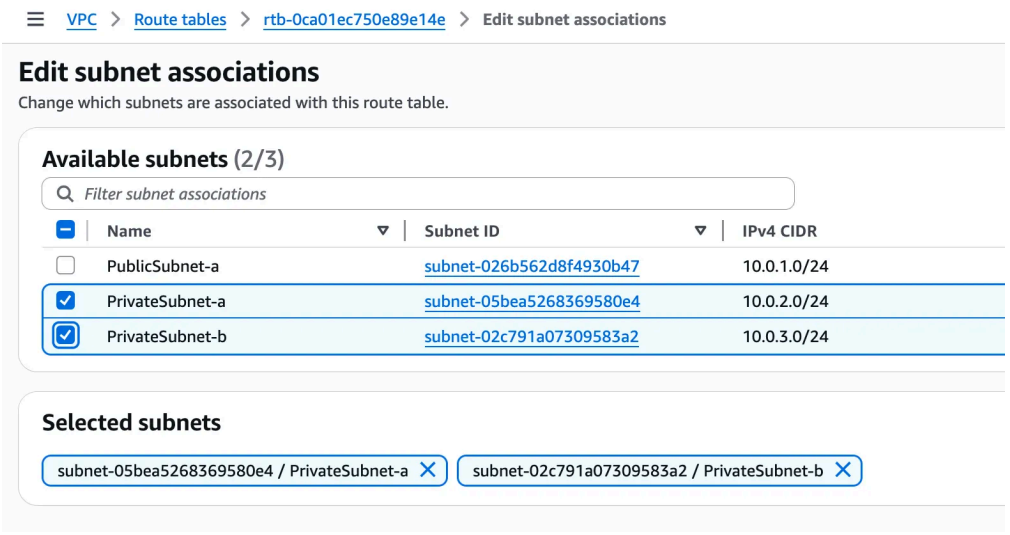
Associate the route table with public subnet



Private Route Table



Associate the route table with both private subnet a and b

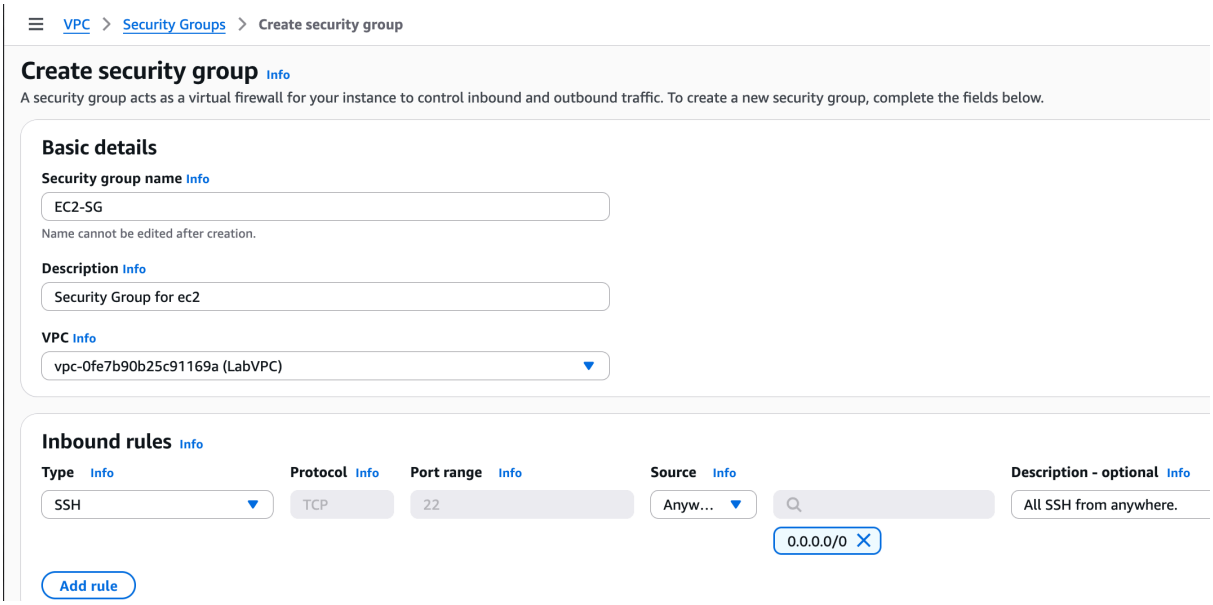


! Note: **Do not** add a NAT Gateway route. We don't need NAT because our EC2 is public and RDS doesn't need outbound internet.

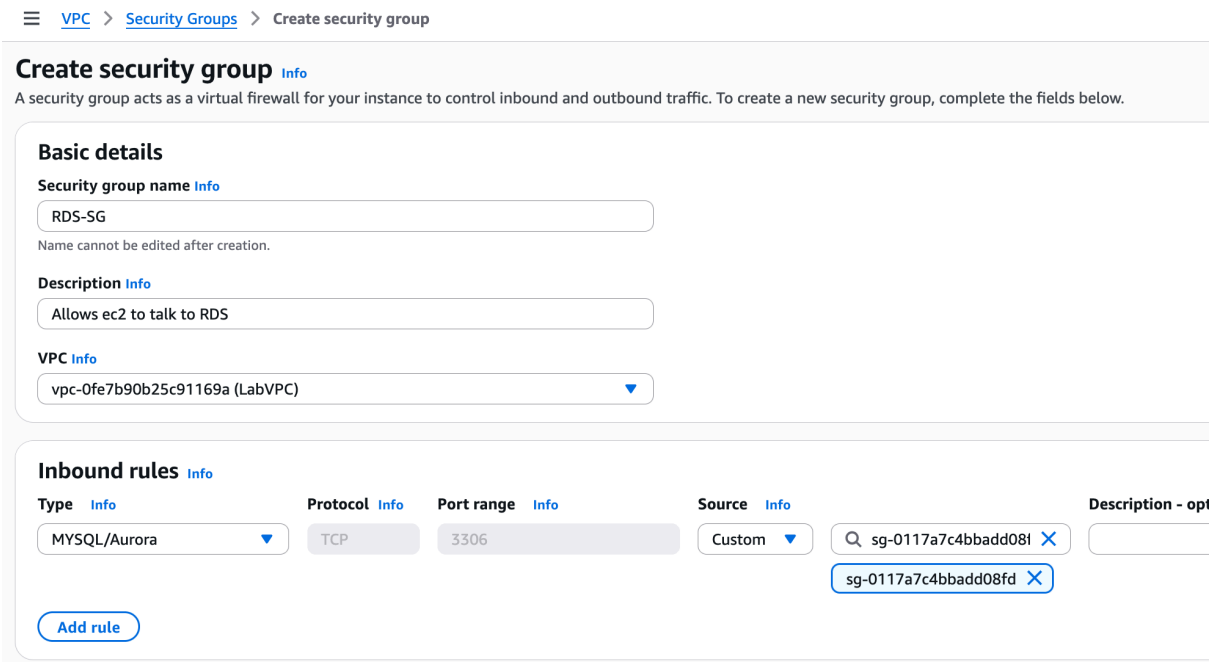
Step 5 — Security Groups

Create two groups: one for EC2, one for RDS. Keep Network ACLs at default (stateless changes often cause confusion in labs).

EC2-SG



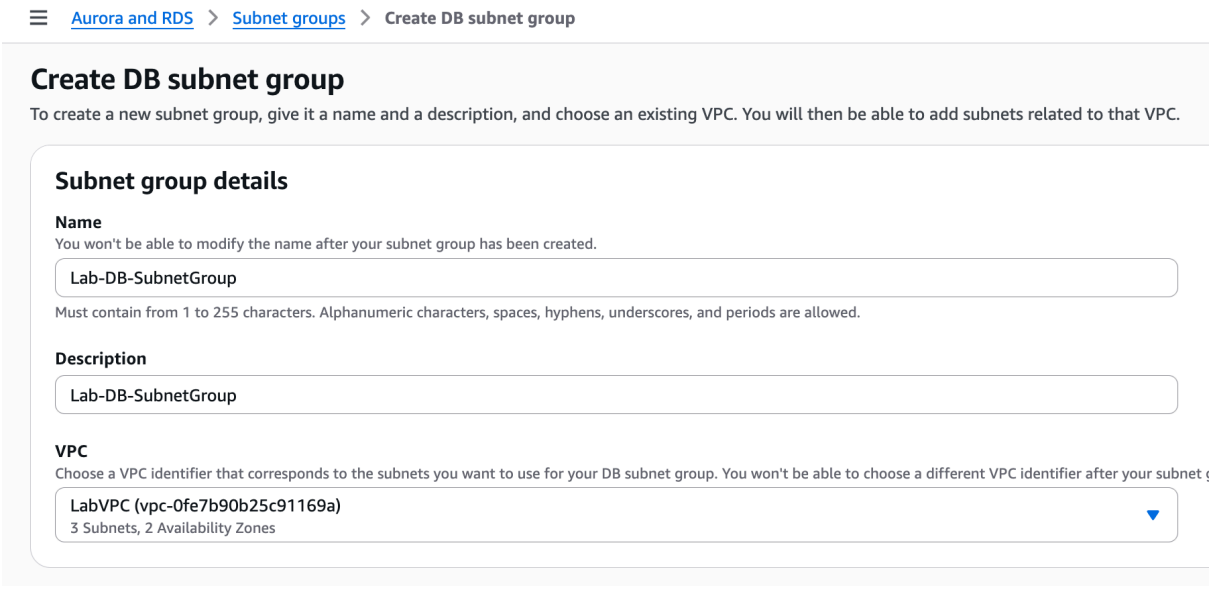
RDS-SG

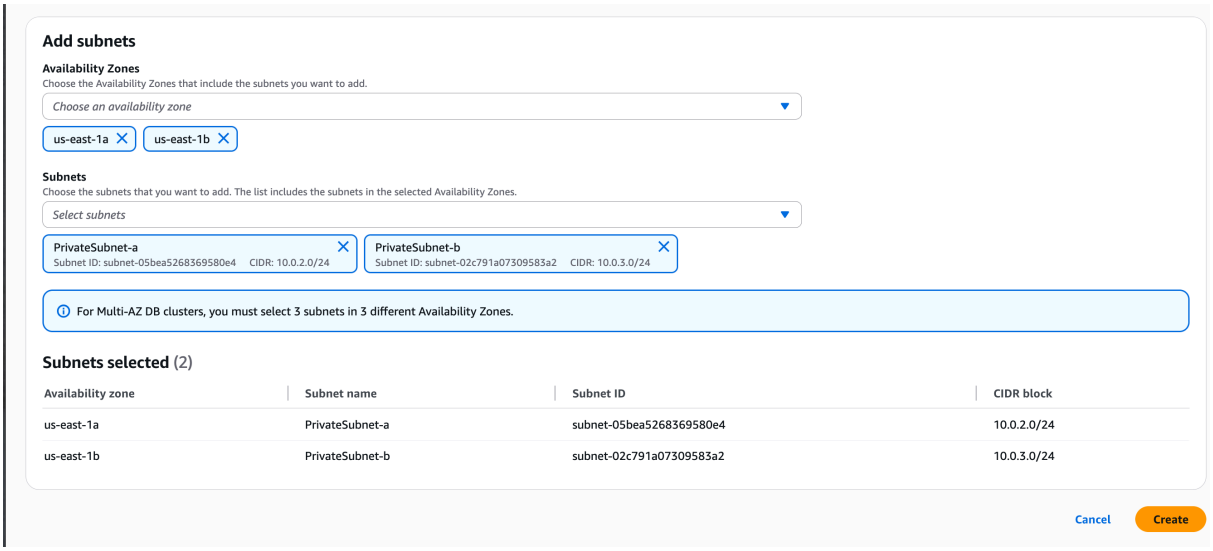


> ! Using **SG-to-SG** rules is best practice. Only the EC2 instances in `EC2-SG` can talk to MySQL on the RDS instance.

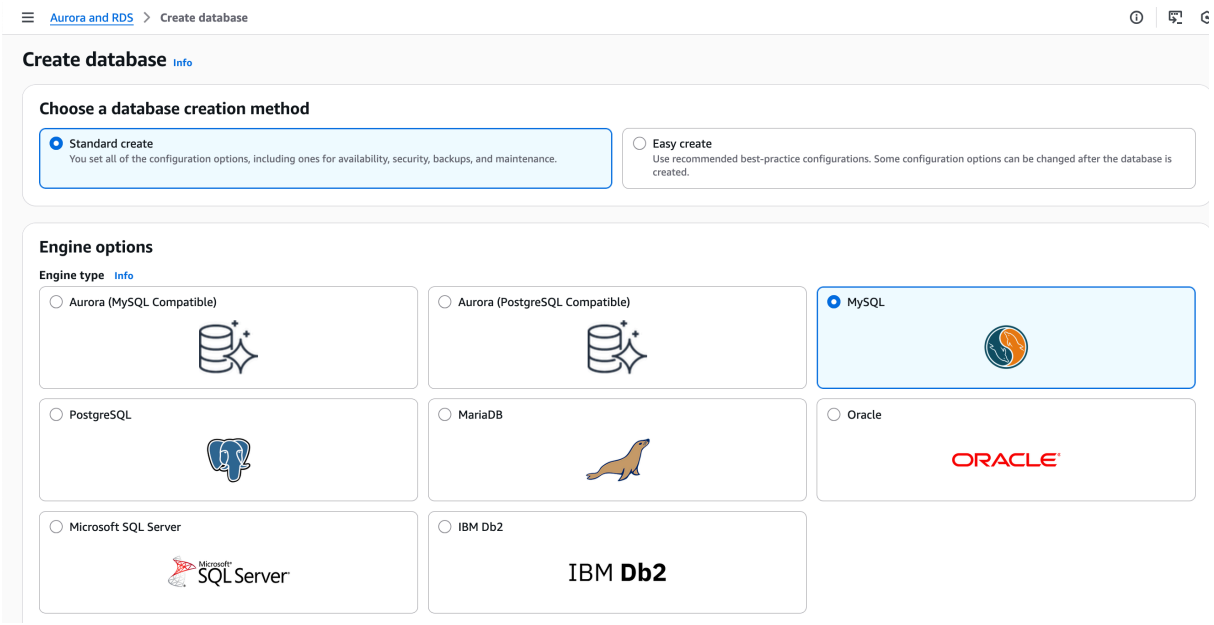
Step 6 — DB Subnet Group (for RDS)

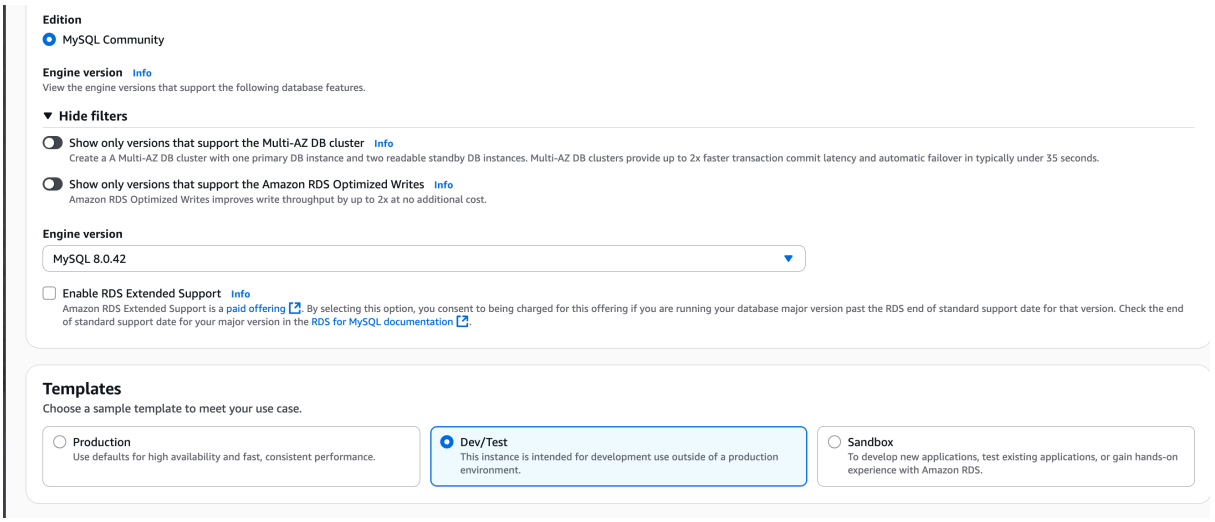
RDS needs a subnet group covering **at least two subnets in different AZs** in your VPC.

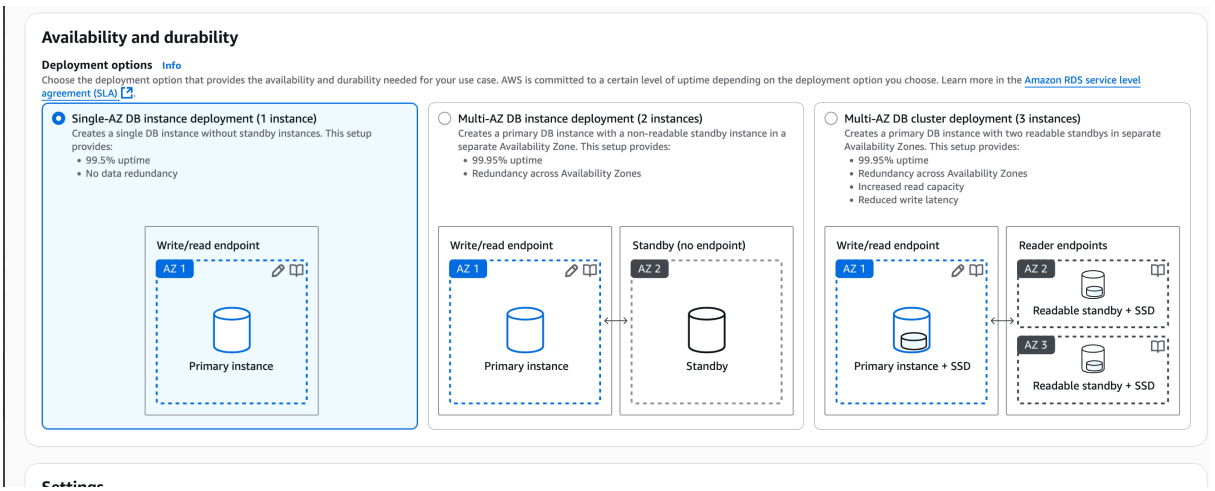


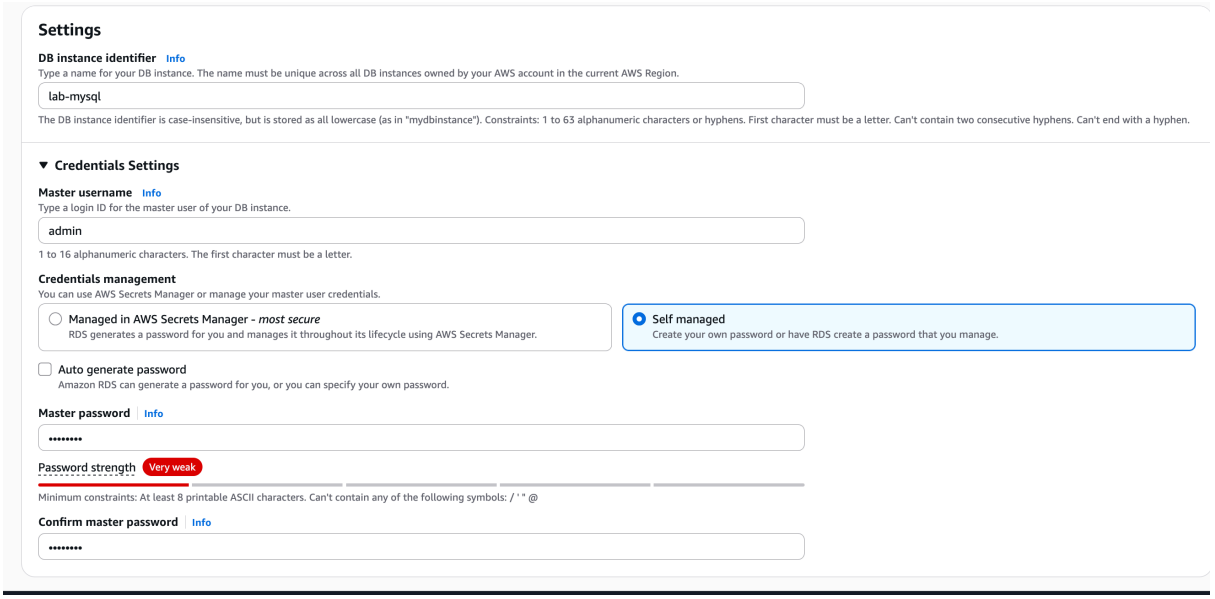


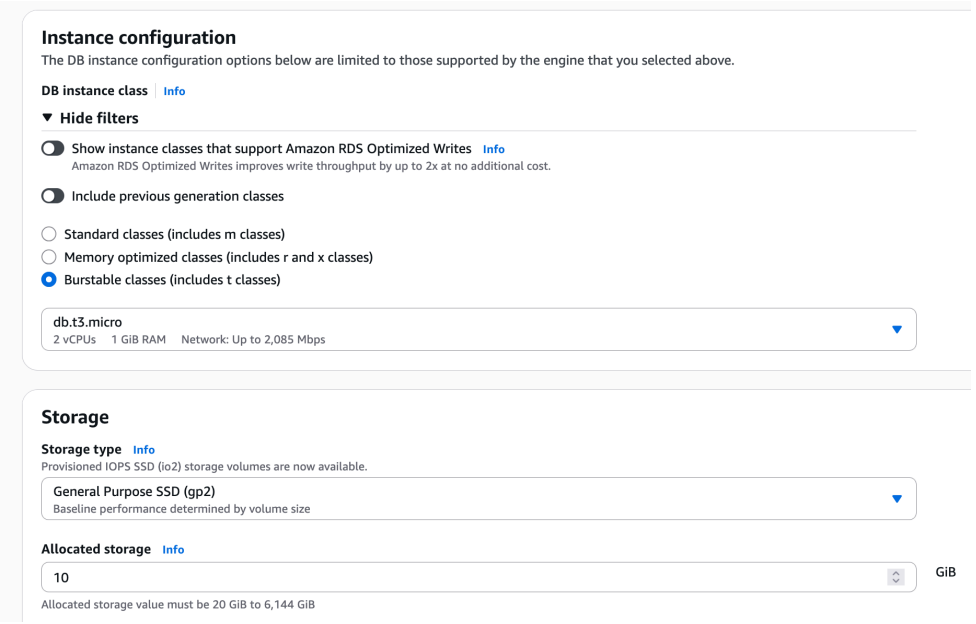
Step 7 — Create the RDS MySQL Instance (Private)


Aurora and RDS > Create database
Create database Info
Choose a database creation method
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
Aurora (MySQL Compatible)
Aurora (PostgreSQL Compatible)
MySQL
PostgreSQL
MariaDB
Oracle
ORACLE
Microsoft SQL Server
SQL Server
IBM Db2
IBM Db2


Edition
MySQL Community
Engine version Info
View the engine versions that support the following database features.
Hide filters
Show only versions that support the Multi-AZ DB cluster Info
Create a A Multi-AZ DB cluster with one primary DB instance and two readable standby DB instances. Multi-AZ DB clusters provide up to 2x faster transaction commit latency and automatic failover in typically under 35 seconds.
Show only versions that support the Amazon RDS Optimized Writes Info
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.
Engine version
MySQL 8.0.42
Enable RDS Extended Support Info
Amazon RDS Extended Support is a paid offering. By selecting this option, you consent to being charged for this offering if you are running your database major version past the RDS end of standard support date for that version. Check the end of standard support date for your major version in the RDS for MySQL documentation.
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.
Sandbox
To develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.


Availability and durability
Deployment options Info
Choose the deployment option that provides the availability and durability needed for your use case. AWS is committed to a certain level of uptime depending on the deployment option you choose. Learn more in the Amazon RDS service level agreement (SLA).
Single-AZ DB instance deployment (1 instance)
Creates a single DB instance without standby instances. This setup provides:
• 99.5% uptime
• No data redundancy
Write/read endpoint
AZ 1
Primary instance
Multi-AZ DB instance deployment (2 instances)
Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:
• 99.95% uptime
• Redundancy across Availability Zones
Write/read endpoint
AZ 1
Primary instance
Standby (no endpoint)
AZ 2
Standby
Multi-AZ DB cluster deployment (3 instances)
Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:
• 99.95% uptime
• Redundancy across Availability Zones
• Increased read capacity
• Reduced write latency
Write/read endpoint
AZ 1
Primary instance + SSD
Reader endpoints
AZ 2
Readable standby + SSD
AZ 3
Readable standby + SSD


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.
lab-mysql
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.
Master password Info
••••••••
Password strength
Very weak
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @
Confirm master password Info
••••••••


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
Storage
Storage type Info
Provisioned IOPS SSD (io2) storage volumes are now available.
General Purpose SSD (gp2)
Baseline performance determined by volume size
Allocated storage Info
10
GiB
Allocated storage value must be 20 GiB to 6,144 GiB

▼ Additional storage configuration

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.

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.

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.

LabVPC (vpc-0fe7b90b25c91169a)
3 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.

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.

lab-db-subnetgroup
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 conne
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 securit

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

Existing VPC security groups

Choose one or more options

RDS-SG ✕

Availability Zone Info

No preference

▼ Additional monitoring settings

Enhanced Monitoring, CloudWatch Logs and DevOps Guru

Enhanced Monitoring

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

Log exports

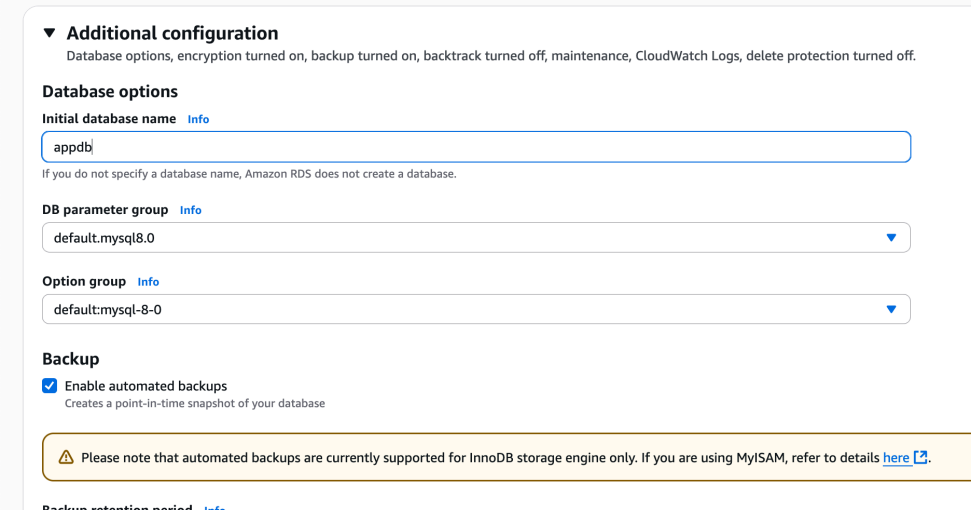
Select the log types to publish to Amazon CloudWatch Logs

☐ Audit log
☐ Error log
☐ General log
☐ iam-db-auth-error log
☐ Slow query log

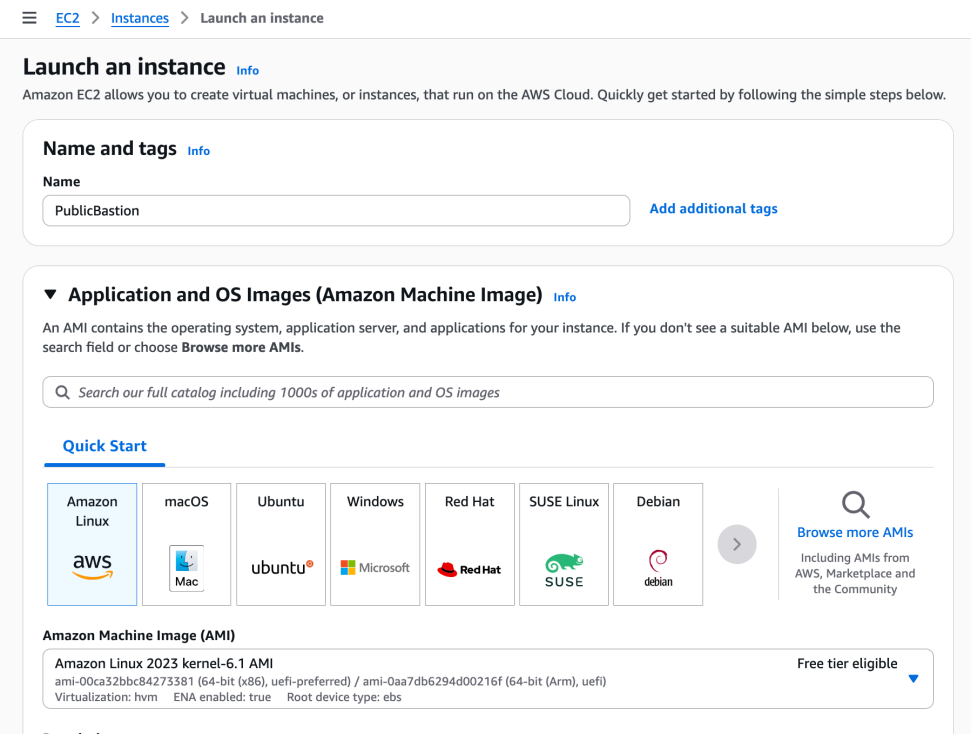
IAM role

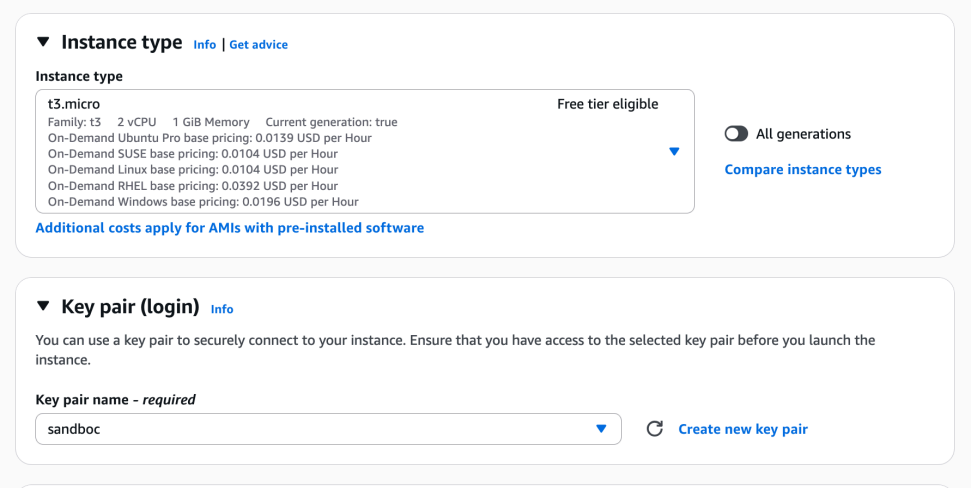
The following service-linked role is used for publishing logs to CloudWatch Logs.

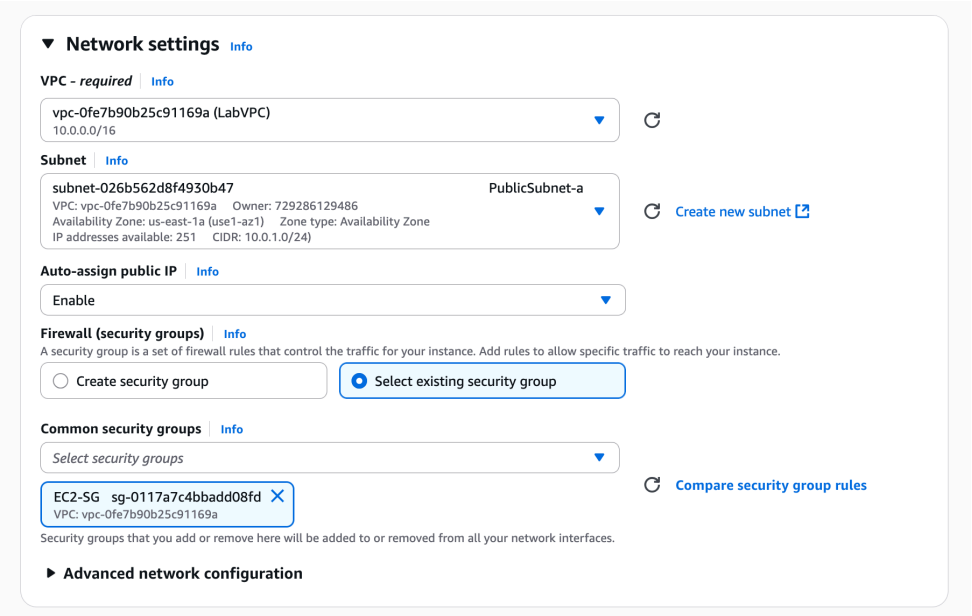
RDS service-linked role



Step 8 — Launch the EC2 Instance (Public Subnet)







User Data automates setup.

- `dnf update -y`
- `dnf install -y mariadb105`

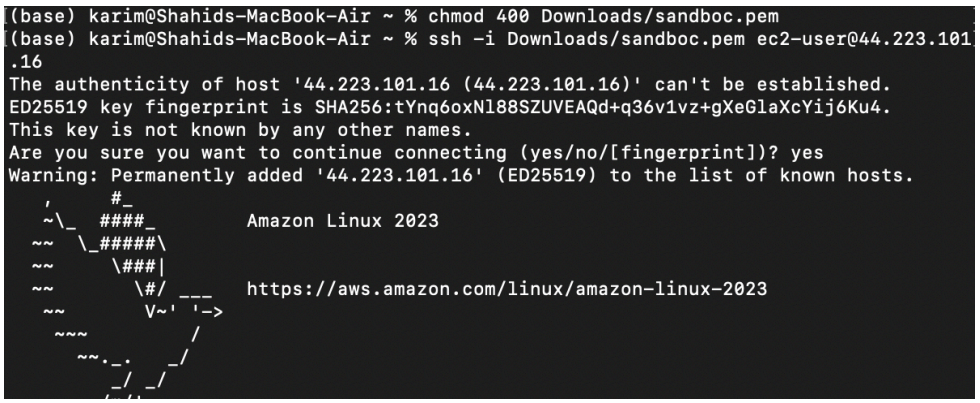
```
#!/bin/bash
# Update system packages
dnf update -y

# Install MySQL client only
dnf install -y mariadb105
```

☐ Input is already base64-encoded

Step 9 — Connect to EC2 and Test MySQL Connectivity

SSH to EC2

```
chmod 400 <.pemfile>
ssh -i <.pemfile> ec2-user@<public-ec2-ip>
```



Connect from EC2 to RDS

```
mysql -h <RDS-endpoint> -P 3306 -u <db-username> -p
```

```
[ec2-user@ip-10-0-1-222 ~]$ mysql -h lab-mysql.cxi8w882k9xj.us-east-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 32
Server version: 8.0.42 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)]>
MySQL [(none)]>
MySQL [(none)]>
```

Using the Database

```
MySQL [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| appdb              |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.005 sec)

MySQL [(none)]> use appdb;
Database changed
MySQL [appdb]> show tables;
Empty set (0.002 sec)
```

```
MySQL [appdb]> CREATE TABLE messages (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     msg VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.039 sec)

MySQL [appdb]> show tables;
+----------------+
| Tables_in_appdb |
+----------------+
| messages       |
+----------------+
1 row in set (0.002 sec)

MySQL [appdb]> INSERT INTO messages (msg) VALUES ('Hello World');
Query OK, 1 row affected (0.008 sec)

MySQL [appdb]> INSERT INTO messages (msg) VALUES ('AWS RDS Test');
Query OK, 1 row affected (0.005 sec)

MySQL [appdb]> INSERT INTO messages (msg) VALUES ('Connected via EC2');
Query OK, 1 row affected (0.005 sec)

MySQL [appdb]>
MySQL [appdb]> select * from messages;
+----+------------------+
| id | msg              |
+----+------------------+
|  1 | Hello World      |
|  2 | AWS RDS Test     |
|  3 | Connected via EC2 |
+----+------------------+
3 rows in set (0.001 sec)
```