

AMAZON RDS: LANZAMIENTO DE UNA INSTANCIA DE BASE DE DATOS Y CONEXIÓN REMOTA

En el mundo actual de la informática en la nube, la gestión eficiente de bases de datos es esencial para el éxito de muchas aplicaciones y servicios en línea. Amazon Web Services (AWS) ofrece una amplia gama de servicios para satisfacer estas necesidades, y uno de los componentes fundamentales es Amazon Relational Database Service (RDS).

En esta práctica, exploraremos cómo configurar una base de datos privada utilizando Amazon RDS con el motor de base de datos MariaDB, una popular alternativa de código abierto a MySQL. Además, aprenderemos a establecer una conexión remota segura desde una instancia de Amazon EC2 pública hacia nuestra base de datos privada en Amazon RDS.

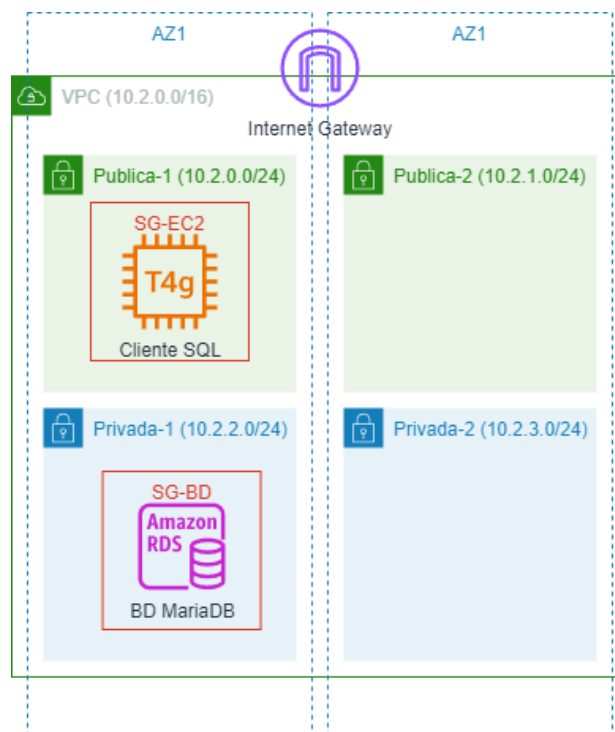
La combinación de RDS y EC2 proporciona una solución escalable y altamente disponible para la gestión de bases de datos relacionales, permitiendo a los desarrolladores centrarse en sus aplicaciones sin preocuparse por la infraestructura subyacente.

A lo largo de esta práctica, guiaremos paso a paso a través de la creación de la base de datos en Amazon RDS, la configuración de la instancia de EC2 pública y la conexión segura entre ambas.

Requerimientos:

- Disponer de acceso a los recursos de AWS a través de un *sandbox* de AWS Academy

Arquitectura propuesta:

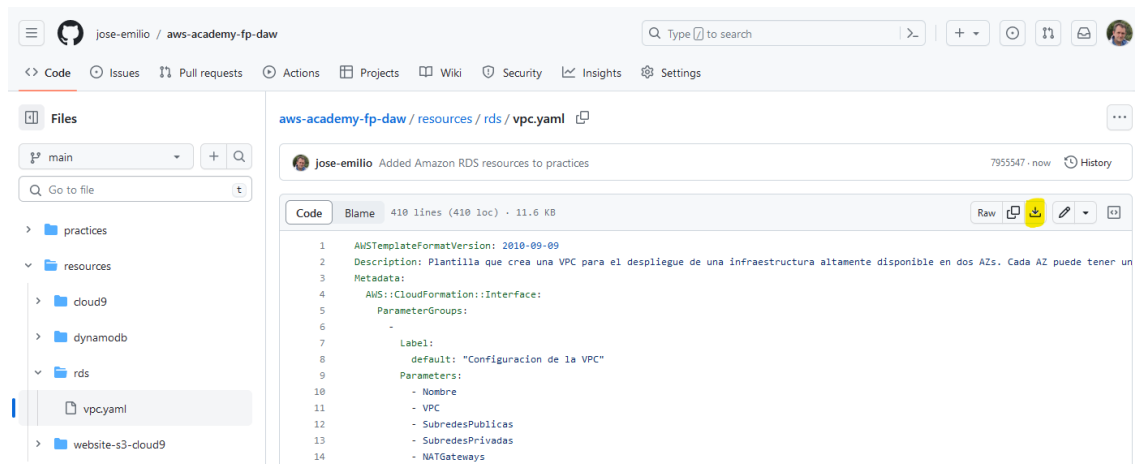


Realización:

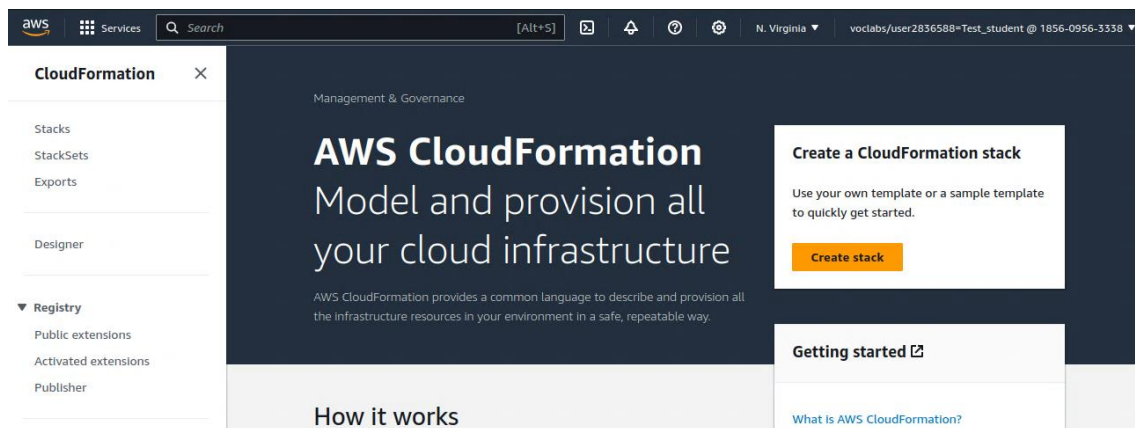
CREACIÓN DE LA INFRAESTRUCTURA DE RED

- 1) Para poder desplegar la arquitectura propuesta es necesario crear la infraestructura de red anterior, compuesta por una VPC con dos subredes privadas y dos subredes públicas en cada zona de disponibilidad. Debido a que el despliegue de dicha infraestructura no es el objetivo de esta práctica, se proporciona una plantilla de AWS CloudFormation para tal fin. Dicha plantilla, llamada `vpc.yaml`, la podemos descargar desde:

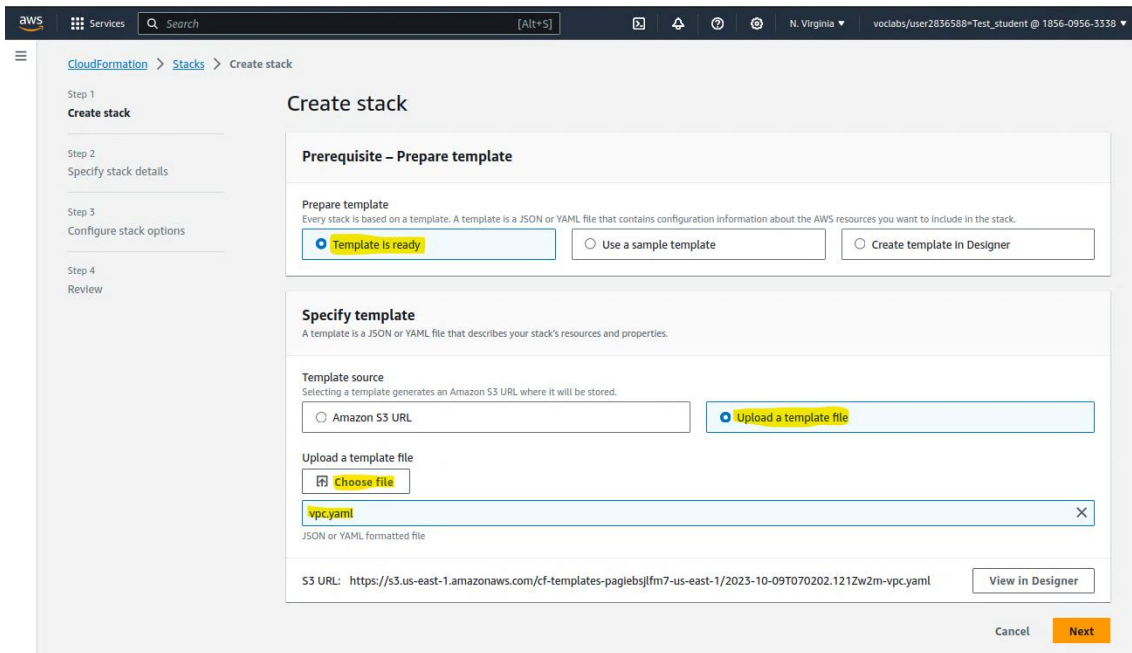
<https://github.com/jose-emilio/aws-academy-fp-daw/blob/main/resources/rds/vpc.yaml>



- 2) Una vez descargada la plantilla anterior, desde la consola de administración de AWS, buscamos el servicio de AWS CloudFormation y accedemos a su consola y presionamos el botón **Create stack**:

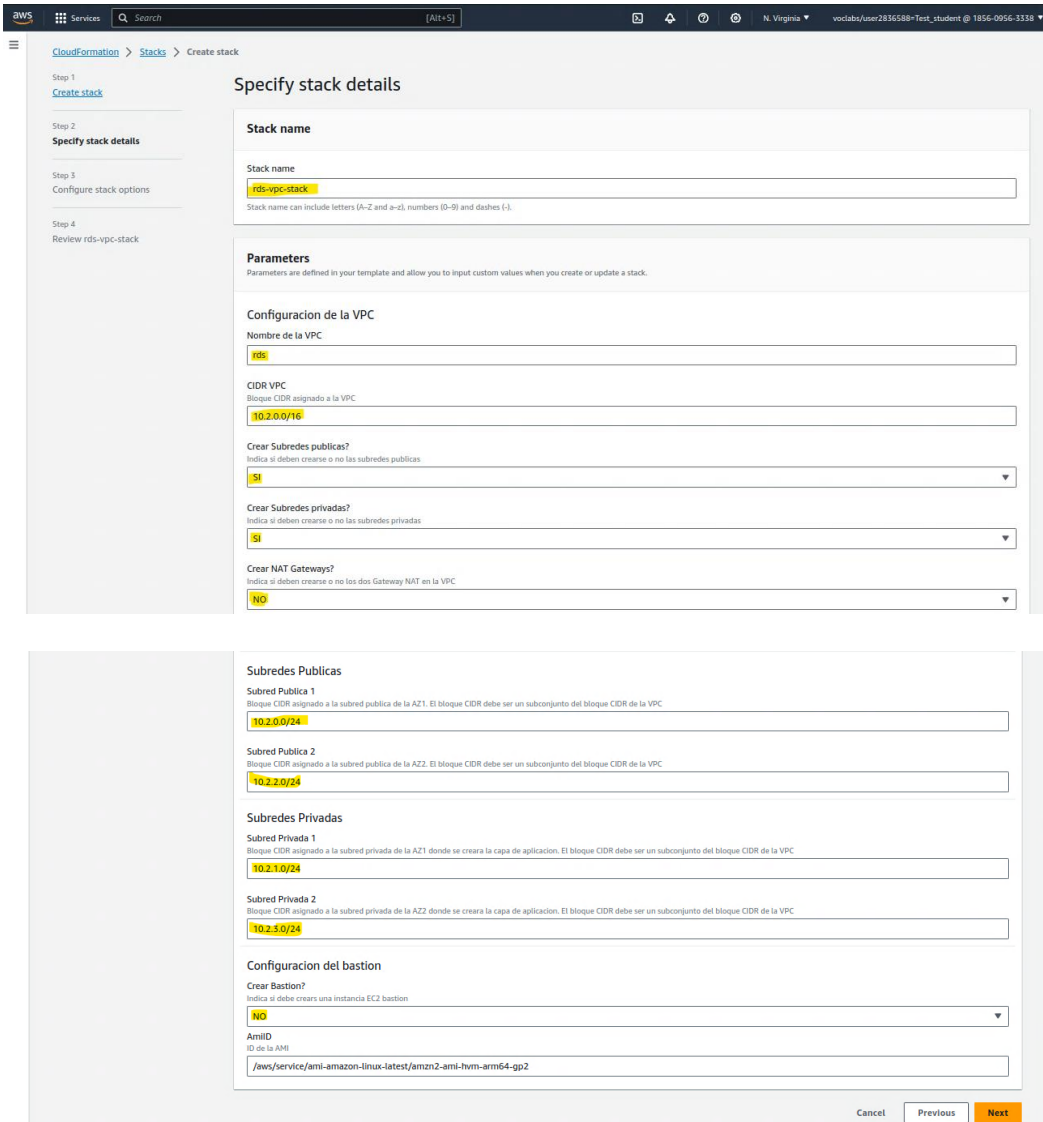


- 3) A continuación, seleccionamos dentro del grupo de opciones **Prepare template** la opción *Template is ready*. En el grupo de opciones **Specify template** elegimos la opción *Upload a template file*, presionamos el botón **Choose file** y cargamos la plantilla `vpc.yaml`. Por último, presionamos el botón **Next**:



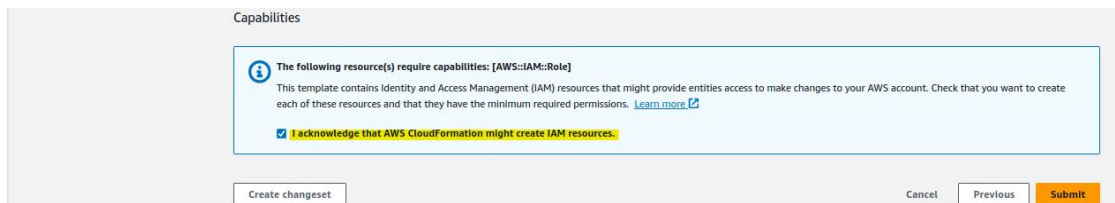
4) En el siguiente formulario, introducimos los siguientes valores:

- **Stack name:** *rds-vpc-stack*
- **Parameters:**
 - **Configuración de la VPC**
 - **Nombre de la VPC:** *rds*
 - **CIDR VPC:** *Por defecto*
 - **Crear subredes públicas:** *Por defecto*
 - **Crear subredes privadas:** *Por defecto*
 - **Crear NAT Gateways:** *NO*
 - **Subredes Públicas**
 - **Subred Pública 1:** *Por defecto*
 - **Subred Pública 2:** *Por defecto*
 - **Subredes Privadas**
 - **Subred Privada 1:** *Por defecto*
 - **Subred Privada 2:** *Por defecto*
 - **Configuración del bastión**
 - **Crear Bastión:** *NO*
 - **AmiID:** *Por defecto*

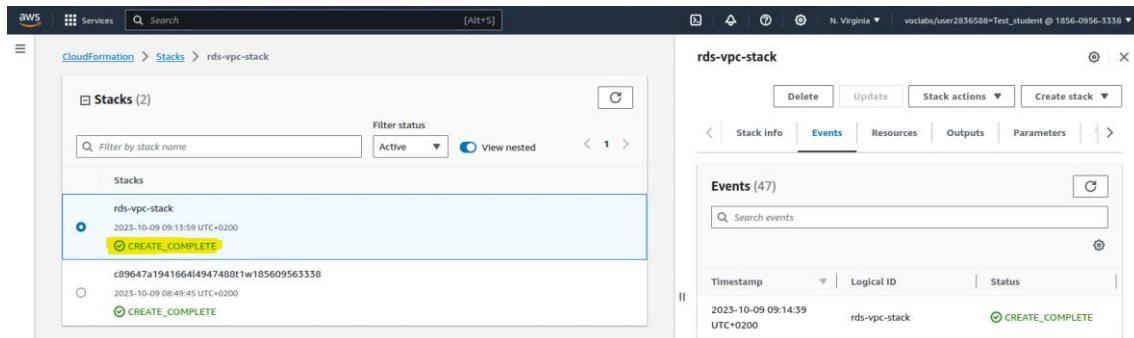


A continuación, presionamos el botón **Next**.

- 5) En la siguiente ventana, dejamos los valores por defecto y presionamos de nuevo el botón **Next**.
- 6) En el último paso, marcamos la casilla de verificación **I acknowledge that AWS CloudFormation might create IAM resources** y presionamos el botón **Submit**.



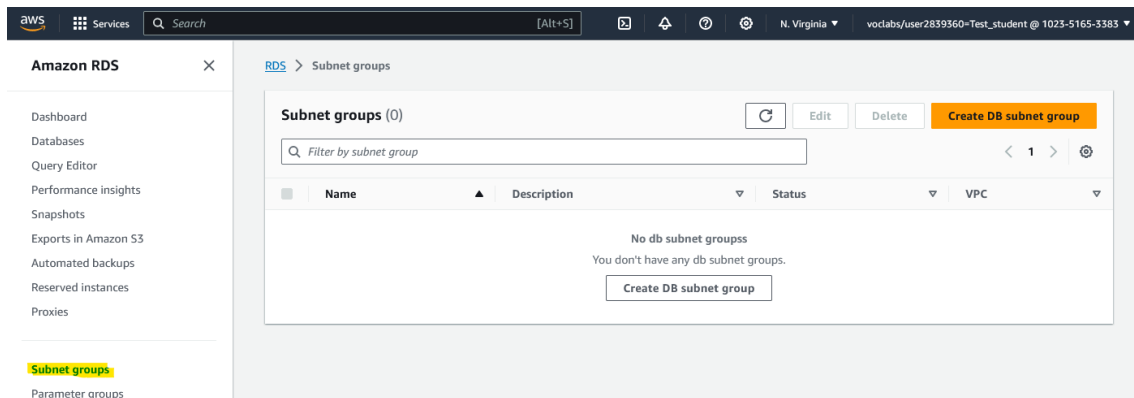
- 7) Tras el proceso anterior, comenzará a desplegarse automáticamente la infraestructura de red y, en cuestión de minutos, comprobaremos que la pila de recursos creados ha pasado a estado `CREATE_COMPLETE`:



LANZAMIENTO Y CONFIGURACIÓN DE UNA INSTANCIA DE AMAZON RDS

Para crear una instancia de Amazon RDS, es necesario crear previamente:

- Un **grupo de subredes**, que definirá en qué subredes podrá desplegarse la instancia de RDS. En esta práctica, crearemos un grupo de subredes que abarque las dos subredes privadas creadas en el apartado anterior.
 - Dos **grupos de seguridad**, uno de ellos determinará el tráfico de entrada y salida permitido a nuestra instancia RDS. En el caso de esta práctica, se permitirá únicamente el tráfico por el puerto 3306 TCP (MySQL) siempre y cuando provenga de la instancia EC2 que se desplegará en la subred pública. Por esta razón será necesario crear un segundo grupo de seguridad para la instancia EC2, pero en este caso, escuchará por el puerto 22 TCP (SSH)
- 8) Para crear el grupo de subredes, accedemos a la consola de Amazon RDS, accedemos a la opción **Subnet groups** del menú lateral y a continuación presionamos el botón **Create subnet group**:



- 9) Dentro del formulario siguiente, introducimos los siguientes valores:

- **Subnet group details**
 - **Name:** `rds-subnet-group`
 - **Description:** *Grupo de subredes de RDS*
 - **VPC:** Elegir la `vpc-rds`
- **Add subnets**

- **Availability Zones:** Elegir *us-east-1a* y *us-east-1b*
- **Subnets:** Elegir las subredes con bloques CIDR *10.2.1.0/24* y *10.2.3.0/24*

RDS > Subnet groups > Create DB subnet group

Create DB subnet group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

Subnet group details

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

Description
Grupo de subredes de RDS

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.
vpc-rds (vpc-07d97e5f84402e328)

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone
us-east-1a X us-east-1b X

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets
subnet-08e7cad553702e60e (10.2.1.0/24) X
subnet-0df90b030c22ca7c1 (10.2.3.0/24) X

For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones.

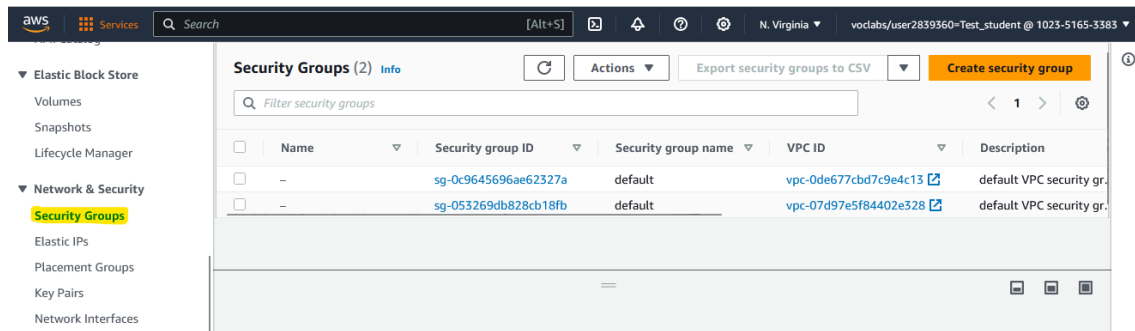
Availability zone	Subnet ID	CIDR block
us-east-1a	subnet-08e7cad553702e60e	10.2.1.0/24
us-east-1b	subnet-0df90b030c22ca7c1	10.2.3.0/24

Subnets selected (2)

Cancel Create

Por último, presionamos el botón **Create**. Tras ello, comprobaremos que el grupo de subredes se haya creado correctamente

- 10) Para crear los dos grupos de seguridad, accedemos a la consola del servicio de Amazon EC2 y, desde el menú lateral accedemos a **Network & Security / Security Groups** y a continuación presionamos el botón **Create security group**:



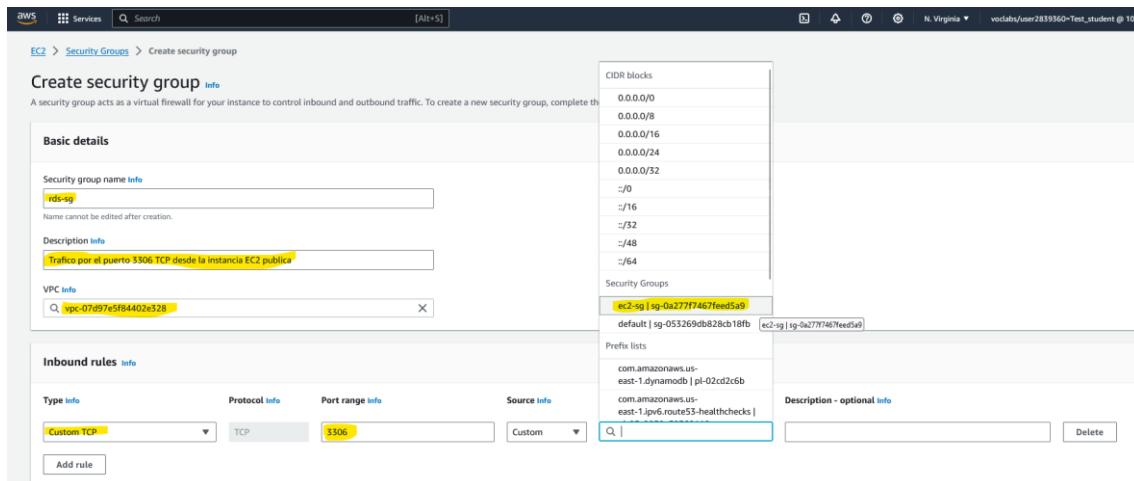
11) En primer lugar, crearemos el grupo de seguridad para nuestra instancia EC2 (donde instalaremos el cliente SQL). Completamos el formulario con los siguientes datos:

- **Basic details:**
 - **Security group name:** *ec2-sg*
 - **Description:** *Trafico por el puerto 22 TCP*
 - **VPC:** Seleccionamos el ID de la *vpc-rds*
- **Inbound rules:**
 - Añadimos una regla que permita el tráfico SSH desde nuestra IP
- **Outbound rules:** Dejamos las opciones por defecto

Por último, presionamos el botón **Create security group**.

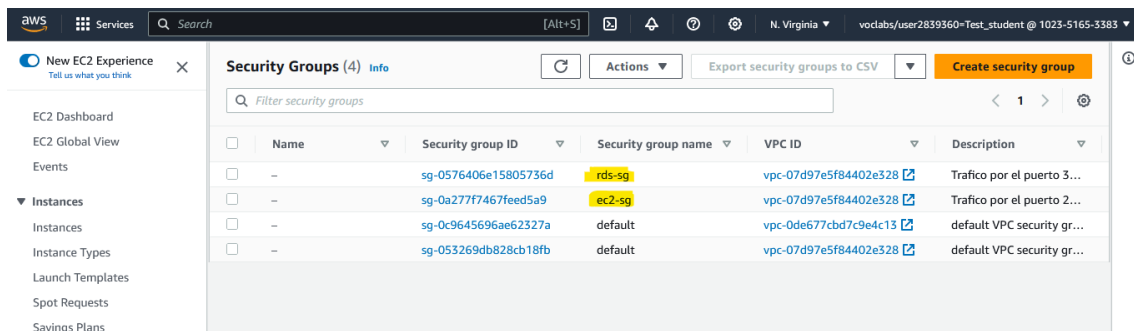
12) A continuación, repetimos el proceso para crear el grupo de seguridad para nuestra instancia de Amazon RDS, con los siguientes parámetros:

- **Basic details:**
 - **Security group name:** *rds-sg*
 - **Description:** *Trafico por el puerto 3306 TCP desde la instancia EC2 publica*
 - **VPC:** Seleccionamos el ID de la *vpc-rds*
- **Inbound rules:**
 - Añadimos una regla que permita el tráfico por el puerto 3306 TCP desde el grupo de seguridad creado en el paso anterior. Para ello, dentro de la opción **Source** hay que elegir el grupo de seguridad etiquetado como *ec2-sg*
- **Outbound rules:** Dejamos las opciones por defecto

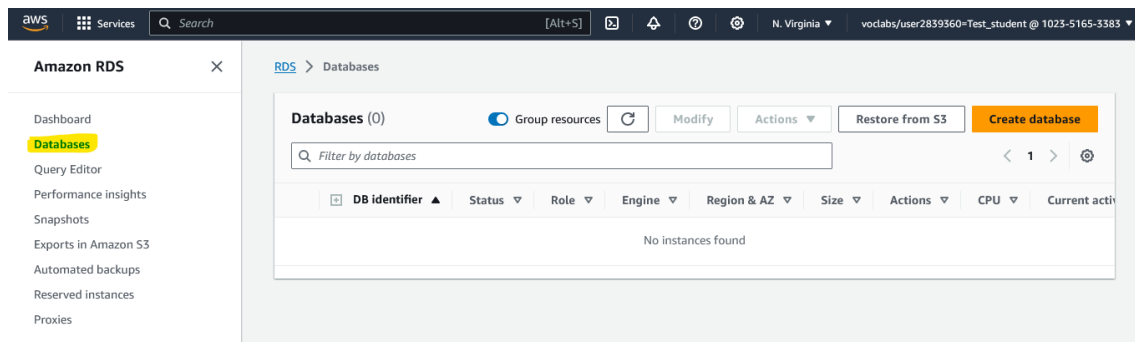


Por último, presionamos el botón **Create security group**.

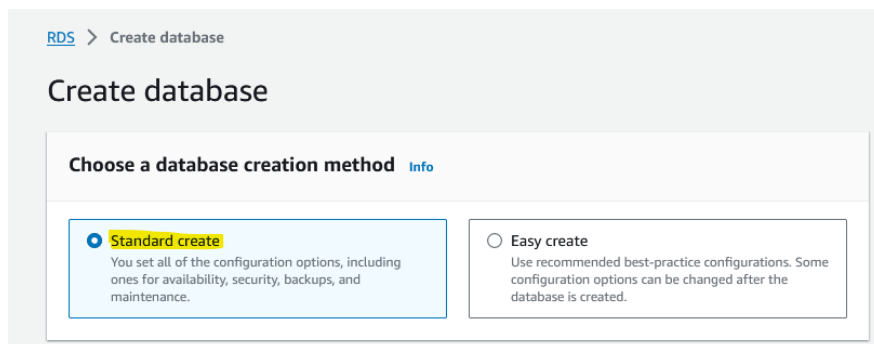
- 13) Tras crear los dos grupos de seguridad, podremos comprobar que aparecen en la consola de Amazon EC2:



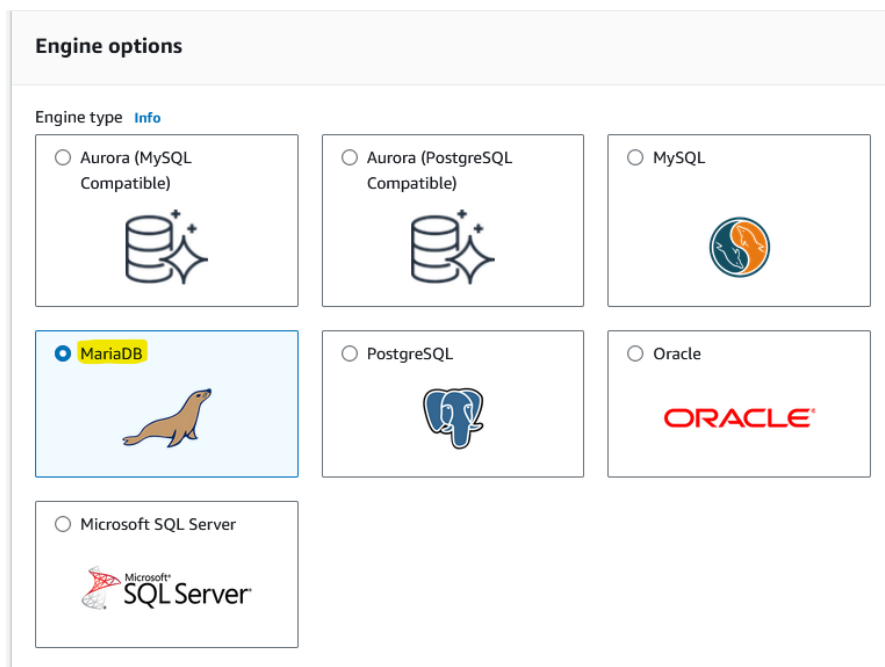
- 14) Ahora procederemos a configurar y lanzar la instancia de base de datos. Para ello, volvemos a la consola de administración de Amazon RDS y, desde el menú lateral, seleccionamos la opción **Databases** y a continuación presionamos el botón **Create database**:



- 15) En el siguiente asistente de configuración, en la opción **Choose a database creation method**, elegiremos la opción *Standard create*:



- 16) Dentro del apartado **Engine options**, elegimos la opción *MariaDB* y dejamos el resto de las configuraciones por defecto:



- 17) En el apartado **Templates** elegimos la opción *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](#)

18) En el apartado **Settings** indicamos la siguiente configuración:

- **DB instance identifier:** *db-practica*
- **Master username:** *admin*
- **Master password:** *adminadmin*
- **Confirm master password:** *adminadmin*

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.

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 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.

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

☐ **Manage master credentials in AWS Secrets Manager**
Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

[If you manage the master user credentials in Secrets Manager, some RDS features aren't supported.](#)
[Learn more](#)

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

Master password [Info](#)


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

Confirm master password [Info](#)

19) En el apartado **Instance configuration** elegimos dentro del campo **DB instance class** la opción *Burstable classes (including t classes)* y en el menú desplegable elegimos la opción *db.t4g.small*:

Instance configuration

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

 **Amazon RDS Optimized Writes - new** [Info](#)

☐ Show instance classes that support Amazon RDS Optimized Writes

DB instance class [Info](#)

☐ Standard classes (includes m classes)

☐ Memory optimized classes (includes r and x classes)

☒ **Burstable classes (includes t classes)**

db.t4g.small

2 vCPUs 2 GiB RAM Network: 2085 Mbps

☐ Include previous generation classes

- 20) En el apartado de **Storage**, configuraremos el volumen EBS asociado a nuestra instancia RDS. Elegiremos dentro del menú desplegable **Storage type** el valor *General Purpose SSD (gp3)*, y en la opción **Allocated storage** indicaremos 20 GiB. El resto de las opciones las dejaremos a sus valores por defecto:

Storage

Storage type [Info](#)

General Purpose SSD (gp3)


Performance scales independently from storage

Allocated storage [Info](#)

20

GiB

Minimum: 20 GiB. Maximum: 16,384 GiB

 After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes. [Learn more](#)

► **Advanced settings**

Baseline IOPS of 3,000 IOPS and storage throughput of 125 MiBps are included for allocated storage less than 400 GiB.

► **Storage autoscaling**

- 21) En el apartado **Connectivity**, seleccionaremos en el menú desplegable **Virtual private cloud (VPC)** la opción *vpc-rds*, para indicar la VPC creada anteriormente. Dentro del menú desplegable **DB subnet group** elegiremos la opción correspondiente al grupo de subredes *rds-subnet-group*, creado anteriormente:

Connectivity
Info

☒
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.

☒
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.

vpc-rds (vpc-07d97e5f84402e328)
4 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.

rds-subnet-group
2 Subnets, 2 Availability Zones

Además, dentro de la opción **Public Access** indicaremos el valor *No*. Igualmente, en la opción **VPC security group (firewall)** elegiremos **Choose existing**, y en el menú desplegable **Existing VPC security groups** desmarcaremos la opción *default* y marcaremos la opción *rds-sg*, para seleccionar el grupo de seguridad creado anteriormente.

Por otra parte, y para ser fieles al diseño arquitectónico propuesto, en la opción **Availability Zone** seleccionaremos el valor *us-east-1a*:

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

Existing VPC security groups

Choose one or more options

rds-sg

Availability Zone
Info

us-east-1a

- 22) En el apartado **Monitoring**, deshabilitamos la opción **Enable Enhanced monitoring**:

Monitoring

Monitoring

☐ **Enable Enhanced monitoring**

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

- 23) En el apartado **Additional configuration**, buscamos la sección **Backup** e indicamos en el menú desplegable **Backup retention period** el valor **1** (con el objeto de no incurrir en costes adicionales; la primera copia de respaldo de Amazon RDS está incluida en el precio de la instancia):

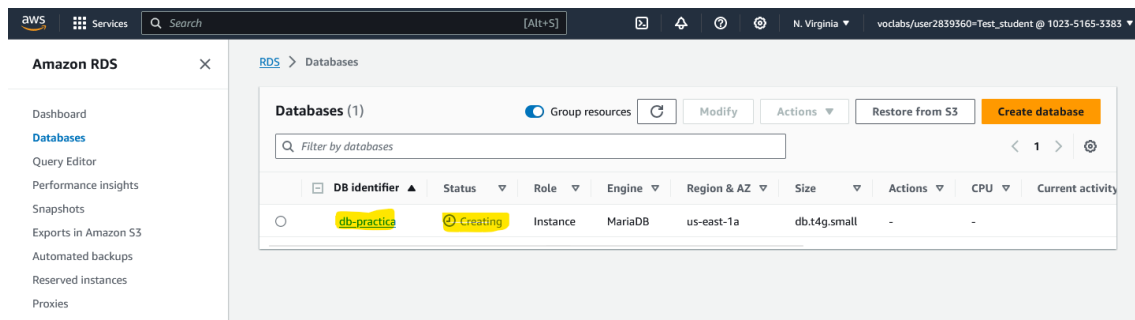
Backup

☒ **Enable automated backups**
Creates a point-in-time snapshot of your database

Backup retention period [Info](#)
The number of days (1-35) for which automatic backups are kept.

1 day

- 24) Una vez terminada la configuración, presionamos el botón **Create database** para comenzar con el despliegue de la instancia de Amazon RDS. El lanzamiento de la instancia puede demorar unos 5-10 minutos. Podremos comprobar el progreso desde la consola de Amazon RDS:



LANZAMIENTO DE UNA INSTANCIA DE AMAZON EC2 Y CONEXIÓN A LA BASE DE DATOS EN RDS

A continuación, se procederá al lanzamiento de una instancia de Amazon EC2 con sistema operativo Linux accesible públicamente, en la cual se instalará un cliente de MySQL para acceder a nuestra instancia RDS.

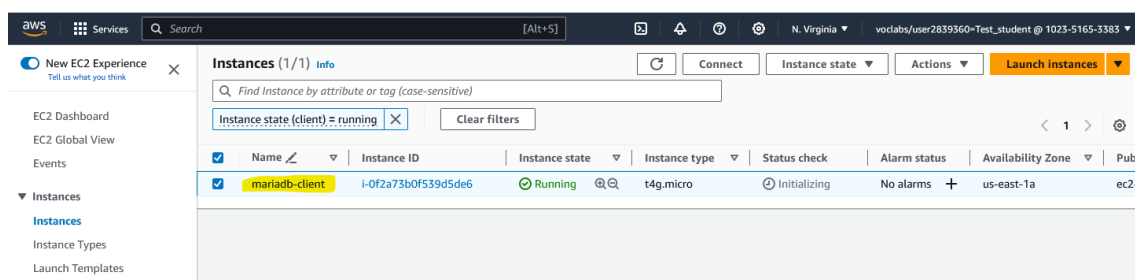
- 25) Para lanzar la instancia EC2, accederemos a la consola de Amazon EC2 y crearemos la instancia con los siguientes parámetros (véase la práctica de lanzamiento de una instancia EC2 Linux https://github.com/jose-emilio/aws-academy-fp-ec2/blob/main/Amazon_EC2_Linux.pdf):

- **Name:** *mariadb-client*
- **Amazon Machine Image (AMI):** *Amazon Linux 2023 AMI*
- **Architecture:** *64-bit (ARM)*
- **Instance type:** *t4g.micro*
- **Key pair (login):** *vockey*

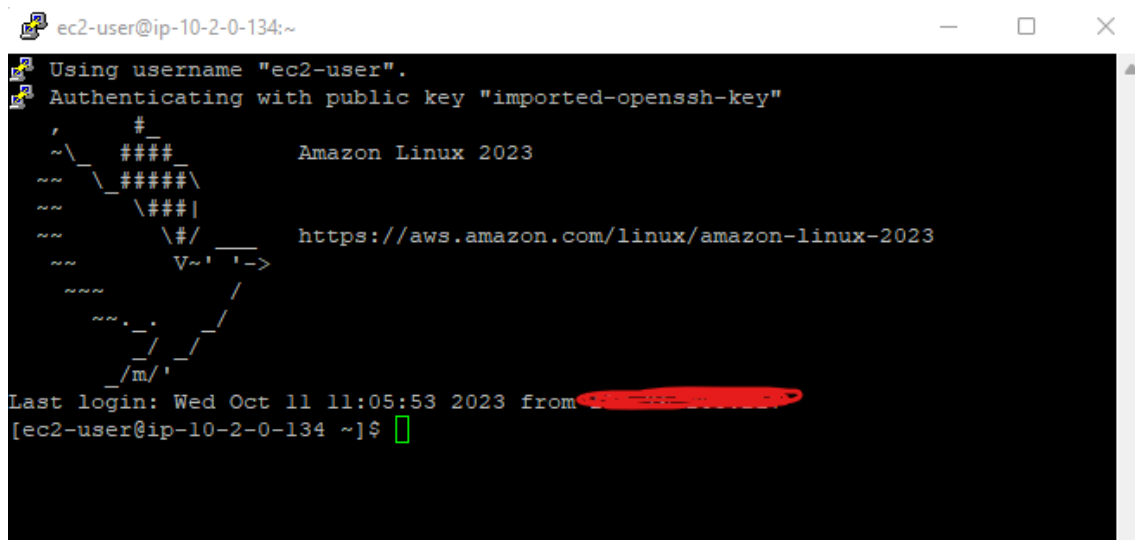
- **Network settings:**
 - **VPC:** *vpc-rds*
 - **Subnet:** *rds-publica-1*
 - **Firewall (security groups):** *Select existing security group y seleccionar ec2-sg*
- **Advanced details:**
 - **User data:** Copiar y pegar el siguiente *script* para instalar automáticamente el cliente de MySQL durante el *bootstrapping* de la instancia EC2:

```
#!/bin/bash
wget https://dev.mysql.com/get/mysql80-community-release-el9-3.noarch.rpm
dnf install mysql80-community-release-el9-3.noarch.rpm -y
rm mysql80-community-release-el9-3.noarch.rpm
dnf update -y
dnf install mysql -y
```

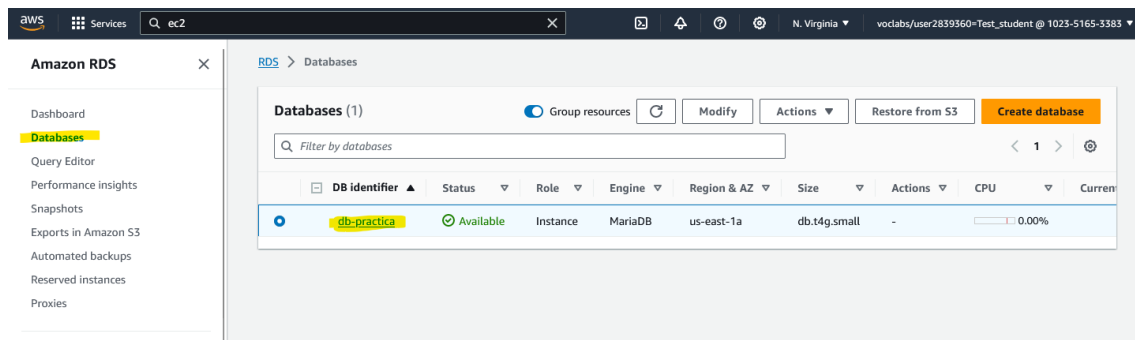
26) Una vez lanzada, podremos visualizarla en la consola de Amazon EC2:



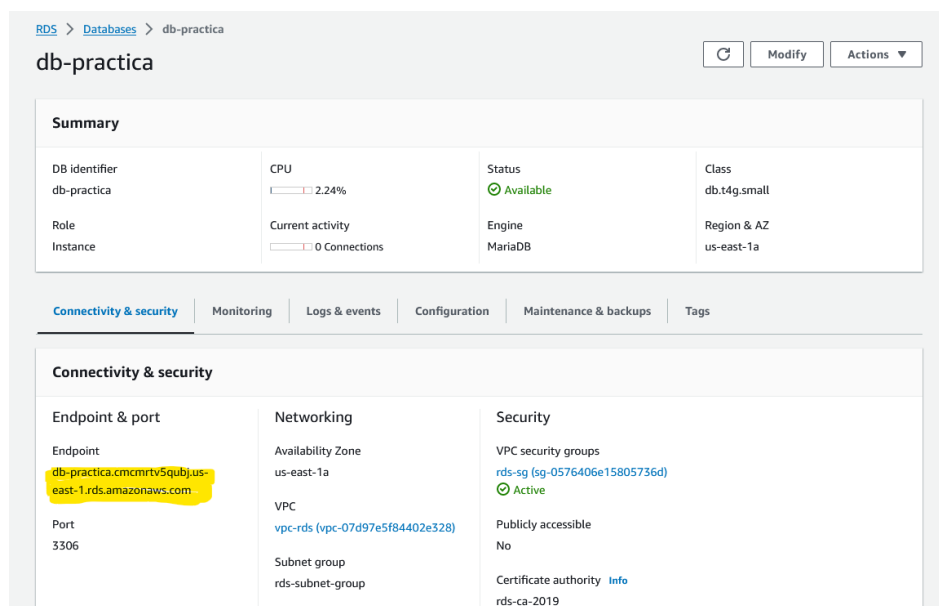
A continuación, lanzaremos una sesión SSH contra la IP pública o nombre DNS público de nuestra instancia EC2 (véase la práctica de lanzamiento de una instancia EC2 Linux):



- 27) Ahora sólo nos resta conectarnos al punto de enlace privado de nuestra instancia de Amazon RDS. Para conocer dicho punto de enlace, volvemos a la consola del servicio de Amazon RDS y, desde el menú lateral, seleccionamos la opción **Databases** y pinchamos en el enlace de nuestra instancia RDS:



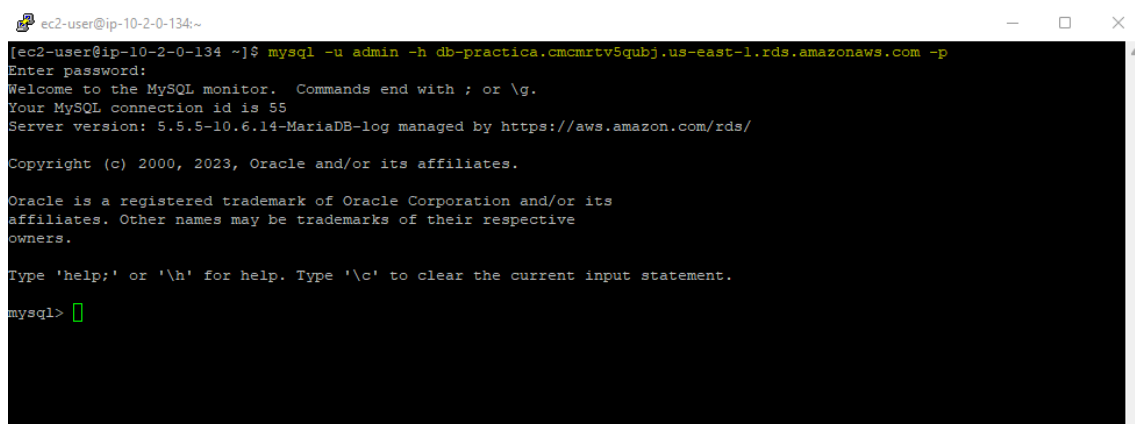
28) A continuación, desde la pestaña **Connectivity & security** copiamos el valor indicado en **Endpoint**:



29) Por último, volvemos a la sesión SSH en nuestra instancia EC2 y procedemos a la conexión con la siguiente orden, sustituyendo el *placeholder* por el valor del punto de enlace copiado:

```
$ mysql -u admin -h <punto-enlace-rds> -p
```

Tras ello, solicitará la contraseña del usuario *admin* e introduciremos *adminadmin*:



- 30) Tras los pasos anteriores, ya habríamos realizado la conexión a nuestra base de datos en Amazon RDS y podríamos introducir las instrucciones DDL y DML para la creación de esquemas o manipulación de datos.

```
ec2-user@ip-10-2-0-134:~$ mysql -u admin -h db-practica.cmcrtv5qubj.us-east-1.rds.amazonaws.com -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 55
Server version: 5.5.5-10.6.14-MariaDB-log managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database prueba;
Query OK, 1 row affected (0.01 sec)

mysql> create table prueba.a (a varchar(10) primary key);
Query OK, 0 rows affected (0.01 sec)

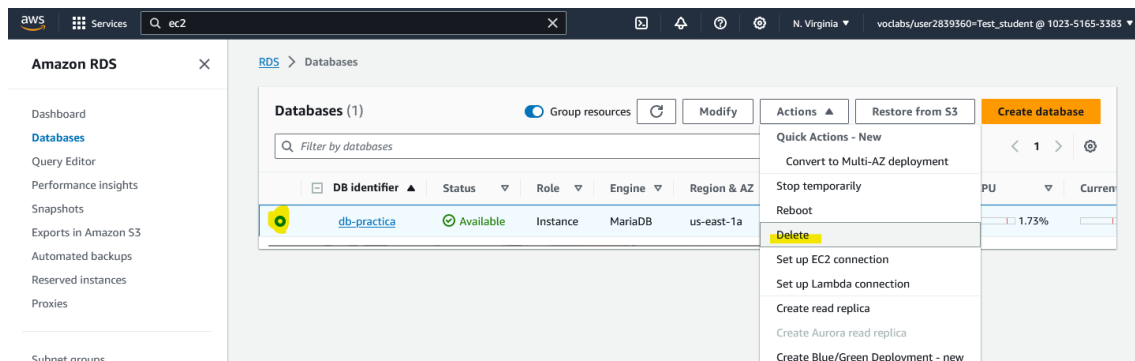
mysql> insert into prueba.a values ("AWS");
Query OK, 1 row affected (0.00 sec)

mysql>
```

Limpieza de la Práctica:

Para terminar esta práctica y liberar los recursos creados, evitando así el consumo de créditos de AWS Academy Learner Labs, simplemente debemos dar los siguientes pasos:

- Eliminar la instancia de Amazon RDS. Para ello, desde la consola de Amazon RDS seleccionamos nuestra instancia y, desde el menú **Actions** elegimos la opción **Delete**.



En la siguiente ventana, desmarcamos las opciones **Create final snapshot** y **Retain automated backups**. Marcamos la opción **I acknowledge that upon instance deletion, ...** y confirmamos la eliminación. La eliminación podrá tardar varios minutos:

Delete db-practica instance?

Permanently delete **db-practica** DB instance? You can't undo this action.

Proceeding with this action will delete the instance with all its content and can affect related resources. [Learn more](#)

☐ **Create final snapshot**
Determines whether a final DB Snapshot is created before the DB instance is deleted.

☐ **Retain automated backups**
Determines whether retaining automated backups for 1 day after deletion

☒ **I acknowledge that upon instance deletion, automated backups, including system snapshots and point-in-time recovery, will no longer be available.**

To avoid accidental deletion provide additional written consent.

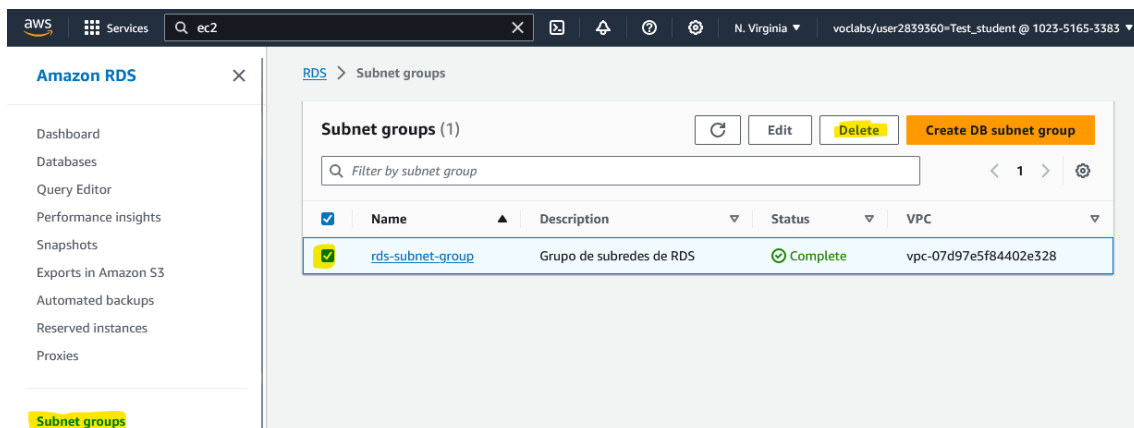
To confirm deletion, type *delete me* into the field.

delete me

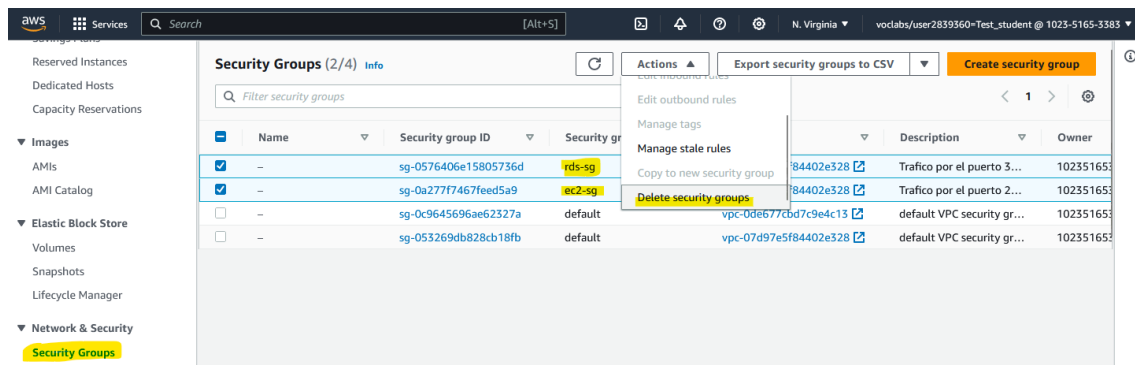
We strongly recommend taking a final snapshot before instance deletion since after your instance is deleted, automated backups will no longer be available.

Cancel Delete

- Eliminar la instancia de Amazon EC2 (véase la práctica de lanzamiento de una instancia EC2 Linux)
- Eliminar el grupo de subredes de Amazon RDS. Desde la consola de Amazon RDS, accediendo al menú lateral **Subnet groups**, seleccionamos el grupo de subredes *rds-subnet-group* y presionamos el botón **Delete**:



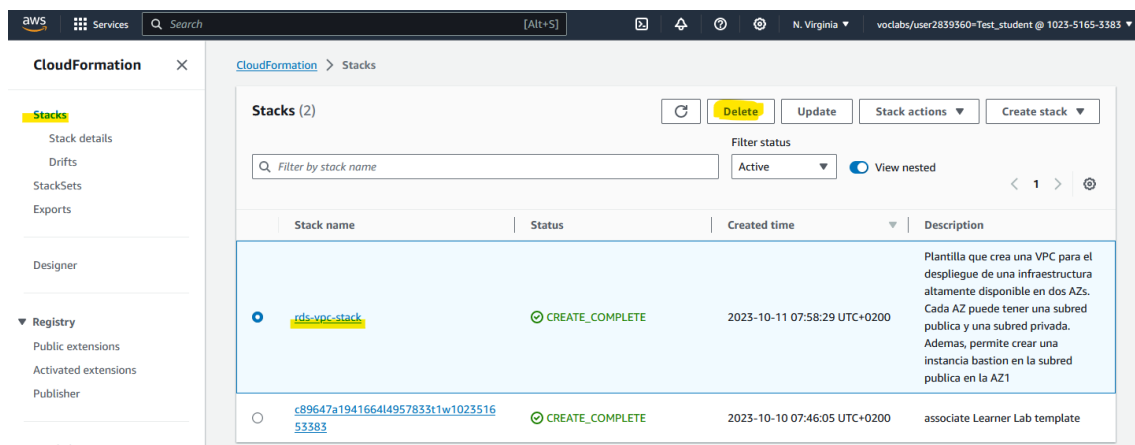
- Eliminar los grupos de seguridad *ec2-sg* y *rds-sg* desde la consola de administración de Amazon EC2, accediendo al menú lateral **Network & Security / Security Groups**



- Eliminar la infraestructura de red creada mediante AWS CloudFormation.

Es necesario que el proceso de eliminación de la instancia de Amazon RDS como el de la instancia EC2 se hayan completado correctamente y las instancias estén eliminadas, antes de continuar con este paso.

Para ello, accedemos a la consola de AWS CloudFormation, seleccionamos en el menú lateral la opción **Stacks**, activamos nuestra pila *rds-vpc-stack* y presionamos el botón **Delete**:



En pocos minutos, nuestra infraestructura de red se habrá decomisionado correctamente.