

AMAZON RDS: RESPALDO DE UNA BASE DE DATOS

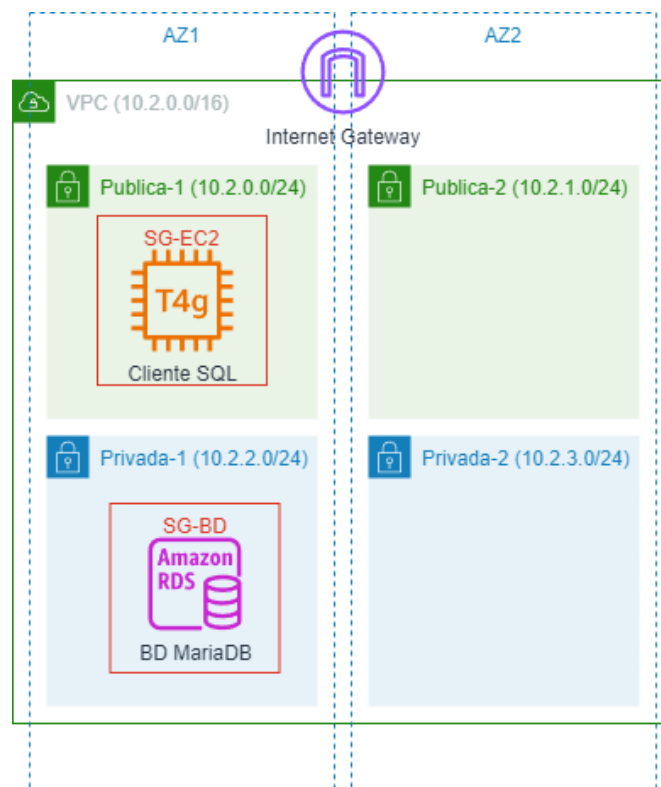
En el mundo actual de la informática en la nube, la gestión eficiente de bases de datos es esencial para el éxito de muchas aplicaciones y servicios en línea. Amazon Web Services (AWS) ofrece una amplia gama de servicios para satisfacer estas necesidades, y uno de los componentes fundamentales es Amazon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

En esta práctica, se configurará una base de datos de Amazon RDS para realizar los respaldos automáticos y se explorarán opciones para realizar respaldos personalizados. Asimismo, se comprobará la forma de restaurar una nueva base de datos a partir de un respaldo.

Requerimientos:

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

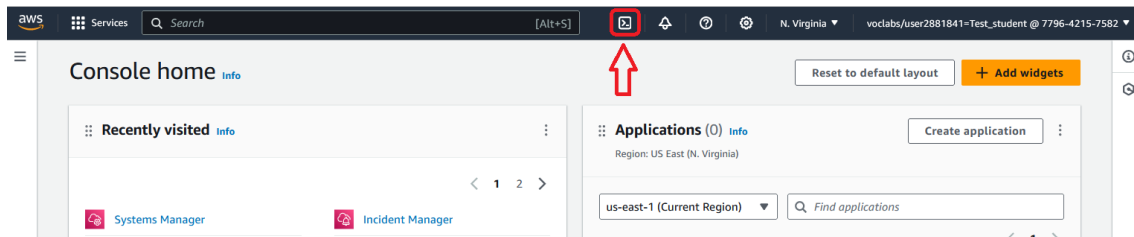
Arquitectura propuesta:



Realización:

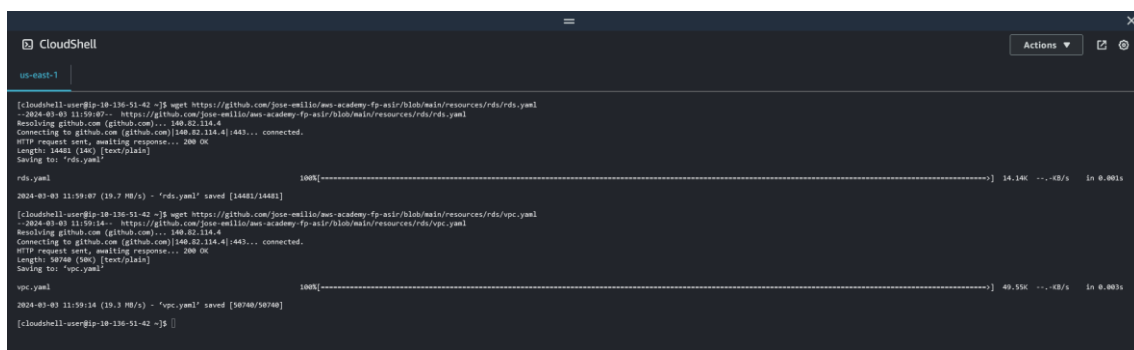
CREACIÓN DE LA INFRAESTRUCTURA DE LA PRÁCTICA

- 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. Para ello abrimos una sesión en AWS CloudShell, tal y como se muestra en la siguiente figura:



- 2) Una vez inicializada la sesión de AWS CloudShell, ejecutamos los siguientes comandos para descargar las plantillas de AWS CloudFormation necesarias para desplegar la infraestructura de la práctica:

```
wget https://raw.githubusercontent.com/jose-emilio/aws-academy-fp-asir/main/resources/rds/rds.yaml
wget https://raw.githubusercontent.com/jose-emilio/aws-academy-fp-asir/main/resources/rds/vpc.yaml
```



- 3) Una vez descargadas las plantillas anteriores, ejecutamos las siguientes instrucciones (sustituyendo los *placeholders* marcados en rojo por los valores apropiados) para el despliegue. Previamente hay que crear un bucket de S3 que se utilizará para almacenar los artefactos del despliegue y establecer el usuario y la contraseña de la instancia RDS:

```
aws s3 mb s3://<nombre-bucket>
BUCKET=<nombre-bucket>
USUARIO=<nombre-usuario-RDS>
PASSWORD=<contraseña-RDS>

aws cloudformation package --s3-bucket $BUCKET --template-file rds.yaml --output-template-file rds-transformed.yaml

aws cloudformation deploy --template-file rds-transformed.yaml --stack-name practica-rds --parameter-overrides Usuario=$USUARIO Password=$PASSWORD
```

Tras la ejecución de las instrucciones anteriores será necesario esperar unos 5 minutos hasta que la infraestructura de la práctica esté completamente preparada (cuando el servicio AWS CloudFormation devuelva el control del *prompt*).

```
CloudShell
us-east-1

[cloudshell-user@ip-10-132-53-108 ~]$ aws s3 mb s3://practica-rds
make_bucket: practica-rds
[cloudshell-user@ip-10-132-53-108 ~]$ BUCKET=practica-rds
[cloudshell-user@ip-10-132-53-108 ~]$ USUARIO=admin
[cloudshell-user@ip-10-132-53-108 ~]$ PASSWORD=adminadmin
[cloudshell-user@ip-10-132-53-108 ~]$
[cloudshell-user@ip-10-132-53-108 ~]$ aws cloudformation package --s3-bucket $BUCKET --template-file rds.yaml --output-template-file rds-transformed.yaml
Uploading to isaabe9567b31a32bf6bd4fa2be97afc.template 12535 / 12535.0 (100.00%)
Successfully packaged artifacts and wrote output template to file rds-transformed.yaml.
Execute the following command to deploy the packaged template
aws cloudformation deploy --template-file /home/cloudshell-user/rds-transformed.yaml --stack-name <YOUR STACK NAME>
[cloudshell-user@ip-10-132-53-108 ~]$
[cloudshell-user@ip-10-132-53-108 ~]$ aws cloudformation deploy --template-file rds-transformed.yaml --stack-name practica-rds --capabilities CAPABILITY_IAM --parameter-overrides Usuario=$USUARIO Password=$PASSWORD

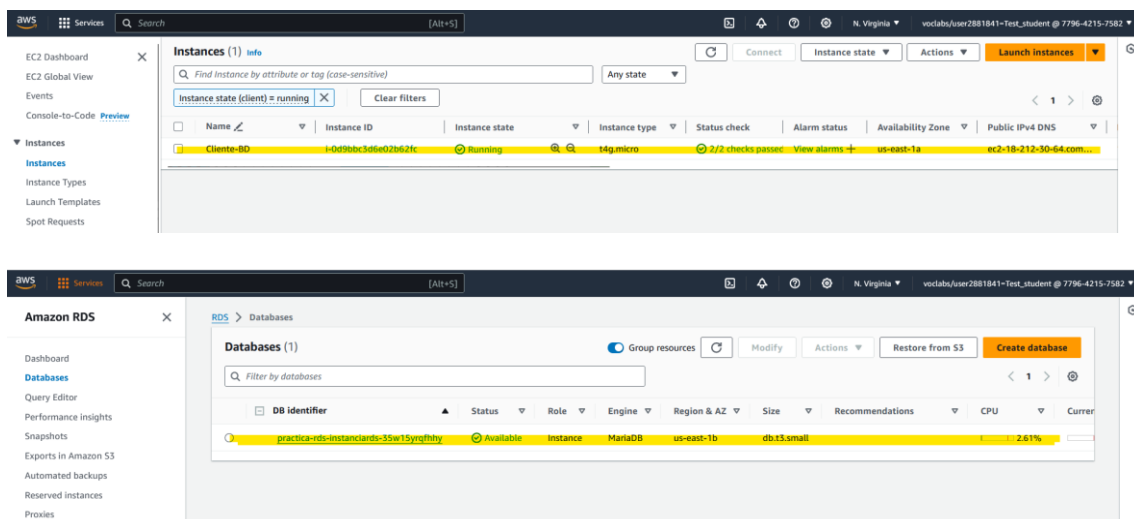
Waiting for changeset to be created..
Waiting for stack create/update to complete
Successfully created/updated stack - practica-rds
[cloudshell-user@ip-10-132-53-108 ~]$
```

EXPLORACIÓN DEL ENTORNO CREADO PARA LA PRÁCTICA

La infraestructura creada consta de un entorno de red (VPC) personalizado con:

- Una instancia EC2 con un cliente MySQL instalado
- Una instancia RDS con motor MariaDB accesible desde la instancia EC2

Podemos comprobarlo accediendo a la consola de los servicios de Amazon EC2 (sección **Instances**) y Amazon RDS (sección **Databases**):



- 4) Si accedemos a la instancia de Amazon RDS desde el enlace de su nombre, podremos visualizar las propiedades, entre las que se incluye el punto de enlace (nombre DNS del recurso):

The screenshot shows the Amazon RDS console. The instance 'practica-rds-instanciards-35w15yrfhhy' is in the 'Available' state. The summary table shows:

DB Identifier	Status	Role	Engine	Recommendations
practica-rds-instanciards-35w15yrfhhy	Available	Instance	MariaDB	
CPU	Class db.t3.small	Current activity	Region & AZ us-east-1b	
	4.20%	1 Connections		

The 'Connectivity & security' section shows:

Endpoint & port	Networking	Security
Endpoint: practica-rds-practica-rds-35w15yrfhhy.chazhczxhyf5aa-east-1-rds.amazonaws.com Port: 3306	Availability Zone: us-east-1b VPC: vpc-practica-rds (vpc-02e4a3d3d8776608) Subnet group: practica-rds-gruposubredesrds-9en0nrwaj7n	VPC security groups: rds-practica-rds-gruposseguridadrds-1dtxxbhi6necu-gbof (tg-05fc458b462de5de) Status: Active Publicly accessible: No

Seleccionamos y copiamos en el portapapeles el punto de enlace de la instancia RDS.

- 5) A continuación, volvemos a la consola del servicio de Amazon EC2 y seleccionando la instancia etiquetada como Cliente-DB, presionamos el botón **Connect**. Desde la pestaña **Session Manager** presionamos el botón **Connect**:

The screenshot shows the 'Connect to instance' dialog in the Amazon EC2 console. The instance 'i-0d9bbc3d6e02b62fc' is selected. The 'Session Manager' tab is active, showing instructions on how to use Session Manager to connect to the instance without SSH keys or bastion hosts.

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) page.

Buttons: Cancel, Connect

- 6) A continuación, visualizaremos el intérprete de órdenes de la instancia EC2. Para conectar desde la instancia EC2 a la instancia RDS, ejecutamos las siguientes órdenes (sustituyendo los *placeholder* por los valores apropiados):

```
RDS=<punto-enlace-BD>
USUARIO=<nombre-usuario-RDS>
PASSWORD=<contraseña-RDS>

mysql -u $USUARIO -p$PASSWORD -h $RDS
```

Session ID: user2881841=Test_student-0e8669868ea6c63e3

Instance ID: i-0d9bbc3d6e02b62fc

[Terminate](#)

```
sh-4.2$ RDS=practica-rds-instanciards-35w15yrgfhhy.cbczhwzwhyf5.us-east-1.rds.amazonaws.com
sh-4.2$ USUARIO=admin
sh-4.2$ PASSWORD=adminadmin
sh-4.2$ mysql -u $USUARIO -p$PASSWORD -h $RDS
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.11.6-MariaDB managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]>
```

- 7) Desde aquí, podemos acceder a la base de datos *sakila* y mostrar las tablas que se han creado, ejecutando las órdenes:

```
use sakila;
show tables;
```

Session ID: user2881841=Test_student-0e8669868ea6c63e3 Instance ID: i-0d9bbc3d6e02b62fc

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> use sakila;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

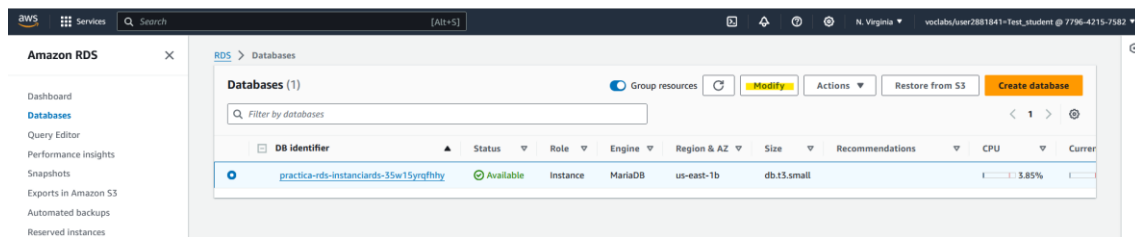
Database changed
MariaDB [sakila]> show tables;
+-----+
| Tables_in_sakila |
+-----+
| actor              |
| actor_info         |
| address            |
| category           |
| city              |
| country            |
| customer           |
| customer_list      |
| film               |
| film_actor         |
| film_category      |
| film_list          |
| film_text          |
| inventory          |
| language           |
| nicer_but_slower_film_list |
| payment            |
| rental             |
| sales_by_file_category |
| sales_by_store     |
| staff              |
| staff_list         |
| store              |
+-----+
23 rows in set (0.00 sec)

MariaDB [sakila]>
```

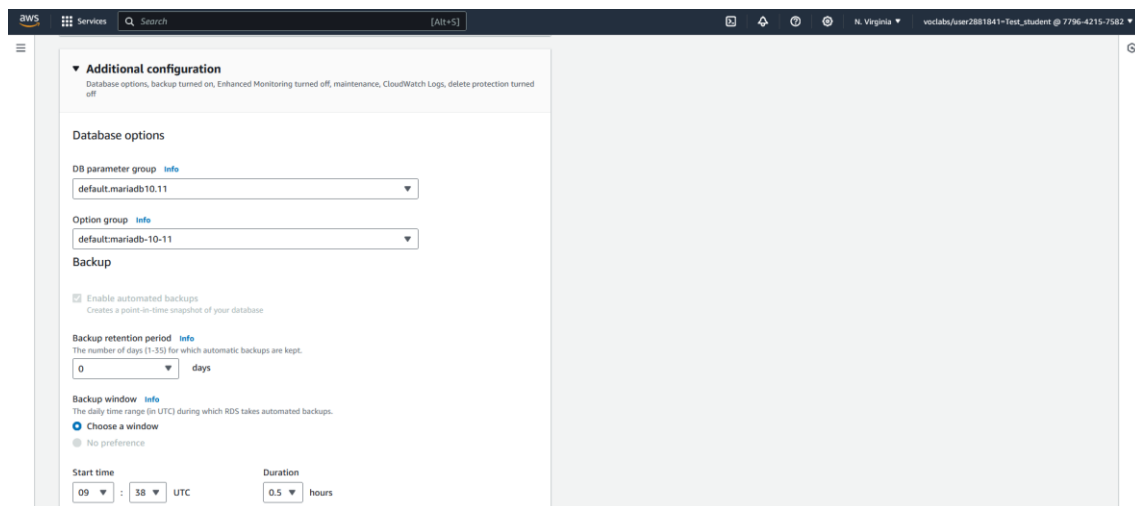
RESPALDOS AUTOMATIZADOS EN AMAZON RDS

Amazon RDS permite habilitar los respaldos automatizados de las instancias RDS. Estos respaldos se realizan a diario, dentro de la ventana de respaldo configurada para ello, y se almacenan en un bucket de S3 administrado por AWS. Adicionalmente, el servicio Amazon RDS almacena los logs transaccionales (*redo log*) en Amazon S3 cada cinco minutos, por lo que en caso de fallo y en el peor de los escenarios, sólo podrían perderse los datos correspondientes a las transacciones de los últimos cinco minutos.

- 8) Para habilitar el respaldo automatizado de nuestra instancia de Amazon RDS, desde la consola de Amazon RDS seleccionamos nuestra instancia y presionamos el botón **Modify**:



- 9) En la sección **Additional configuration** existe un apartado denominado **Backup** donde podemos visualizar que los respaldos automáticos no están habilitados:



- 10) Para habilitar y configurar los respaldos automáticos se deben tener en cuenta:
- El campo **Backup retention period**. Este campo indica cuántos respaldos diarios estamos dispuestos a almacenar. El valor de este campo actualmente es 0, lo que indica que no se realizan respaldos. El valor máximo que se puede indicar es 35, para mantener los respaldos de los últimos 35 días. Indicaremos el valor 3, para mantener los respaldos de los últimos 3 días. Estos respaldos son incrementales (a excepción del primero que es completo)
 - El campo **Backup window**, donde se configura la ventana temporal durante la que se realizan los respaldos. Para mantener una mayor eficiencia, esta ventana debe establecerse cuando menos carga de trabajo exista sobre la base de datos. En nuestro ejemplo, seleccionaremos una ventana de tiempo entre las 3:00 am con una duración de 1 hora.
 - El campo **Backup replication** permite crear, adicionalmente al respaldo en la región actual, una réplica del respaldo en otra región diferente. Esta acción se utiliza para escenarios de recuperación ante desastres; en el improbable caso de inactividad regional sería posible restaurar una nueva instancia RDS en otra región a partir de la réplica del respaldo.

Una vez configurados los campos anteriores, presionamos el botón **Continue**. Tras ello, aparecerá una nueva ventana de confirmación de los cambios en nuestra instancia RDS. Estos cambios pueden aplicarse o bien inmediatamente o en la próxima ventana de mantenimiento de la instancia. Seleccionamos la opción **Apply immediately** y presionamos el botón **Modify DB instance**:

Attribute	Current value	New value
Backup retention period	0	3
Backup window	09:38-10:08	03:00-04:00

A continuación, podremos visualizar que la instancia RDS se está modificando para aplicarse los cambios. Este proceso podría demorar varios minutos:

DB Identifier	Status	Role	Engine	Region & AZ	Size	Recommendations	CPU	Curr
practica-rds-instanciards-35w15yqfhhhy	Modifying	Instance	MariaDB	us-east-1b	db.t3.small		3.84%	

- 11) Una vez nuestra instancia RDS se encuentre en estado disponible (*Available*), si accedemos a la consola de Amazon RDS a nuestra instancia, desde la pestaña **Maintenance & Backups** podremos comprobar que el respaldo automático se ha habilitado según la configuración indicada:

The screenshot shows the 'Maintenance & backups' tab in the AWS Management Console. It includes sections for 'Maintenance' (Auto minor version upgrade: Enabled, Maintenance window: March 07, 2024 05:53 - 06:23 (UTC+01:00), Pending maintenance: none, Pending modifications: none) and 'Backup'. The 'Backup' section shows 'Automated backups' as 'Enabled (1 Day)', 'Copy tags to snapshots' as 'Disabled', 'Backup target' as 'AWS Cloud (US East (N. Virginia))', 'Latest restore time' as 'March 03, 2024, 18:10 (UTC+01:00)', 'Backup window' as '03:00-04:00 UTC (GMT)', and 'Replicate to Region' as 'Replicated automated backup'.

- 12) Desde la consola de Amazon RDS a la opción **Automated backups** del menú lateral. Desde la ventana, podrá visualizarse nuestra instancia RDS y el respaldo que se ha realizado. La columna **Latest restorable time** indica el primer punto del tiempo en el que puede restaurarse una nueva instancia RDS a partir del respaldo. La columna **Earliest restorable time** indica el instante del tiempo más actual al que puede restaurarse una nueva instancia RDS a partir del respaldo. Esto implica que, como máximo es posible restaurarse una nueva instancia RDS desde los últimos 5 minutos hasta los últimos 35 días; aunque en nuestro caso sería a los 3 últimos días, ya que sólo mantenemos los 3 últimos respaldos automáticos. Esta característica de Amazon RDS se denomina **PITR (Point in Time Recovery)**

The screenshot shows the 'Current Region backups' table in the AWS Management Console. The table has columns: DB instance or cluster, Earliest restorable time, Latest restorable time, Engine, and Encrypted. The data row shows a backup for 'practica-rds-instancias-35w15yqfthy' with an Earliest restorable time of 'March 03, 2024, 17:40 (UTC+01:00)' and a Latest restorable time of 'March 03, 2024, 17:40 (UTC+01:00)'. The engine is 'mysql' and it is not encrypted.

- 13) Para restaurar una instancia RDS a partir de un respaldo automatizado, bastaría con seleccionar el respaldo y, desde el menú **Actions** seleccionar la opción **Restore to Point in Time**:

The screenshot shows the 'Current Region backups' table in the AWS Management Console. The 'Actions' menu is open, and the 'Restore to point in time' option is highlighted. The table data is the same as in the previous screenshot.

- 14) Desde la ventana siguiente, podremos comprobar que podemos elegir diferentes opciones de configuración de la restauración. En concreto, en la sección **Restore time** podremos seleccionar la opción **Latest restorable time** para restaurar nuestra instancia RDS a los 5 minutos antes del momento actual, o bien la opción **Custom date and time** para elegir un punto en el tiempo desde el que deseamos restaurar la instancia RDS:

Restore time

Point in time to restore from

☐ Latest restorable time
March 03, 2024, 17:50:00 (UTC+1:00)

☒ Custom date and time
The date must be before the latest restorable time for the DB instance.

Date

YYYY/MM/DD

Time

00 : 00 : 00 UTC+1:00

La configuración final para la restauración de la instancia RDS será la siguiente:

- En el apartado **Restore time**, seleccionamos la opción *Latest restorable time*.
- En la sección **Settings**, introducimos en el campo **DB instance identifier** el valor *db-restore-1*
- En la sección **Additional configuration**, en el apartado **Backup** deshabilitaremos los respaldos automatizados para la nueva instancia RDS.

Restore time

Point in time to restore from

☒ Latest restorable time
March 03, 2024, 17:50:00 (UTC+1:00)

☐ Custom date and time
The date must be before the latest restorable time for the DB instance.

Settings

DB engine
Name of the database engine to be used for this instance.

MariaDB

License model
License type associated with the database engine

general-public-license

Source DB instance identifier [Info](#)
practica-rds-instanciards-35w15yrfhhy

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.

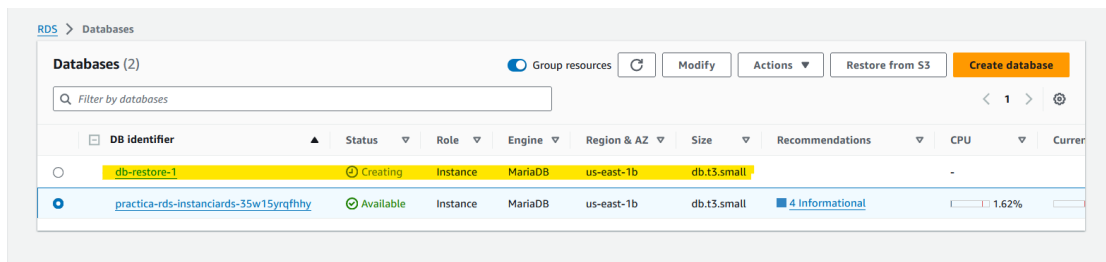
db-restore-1

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.

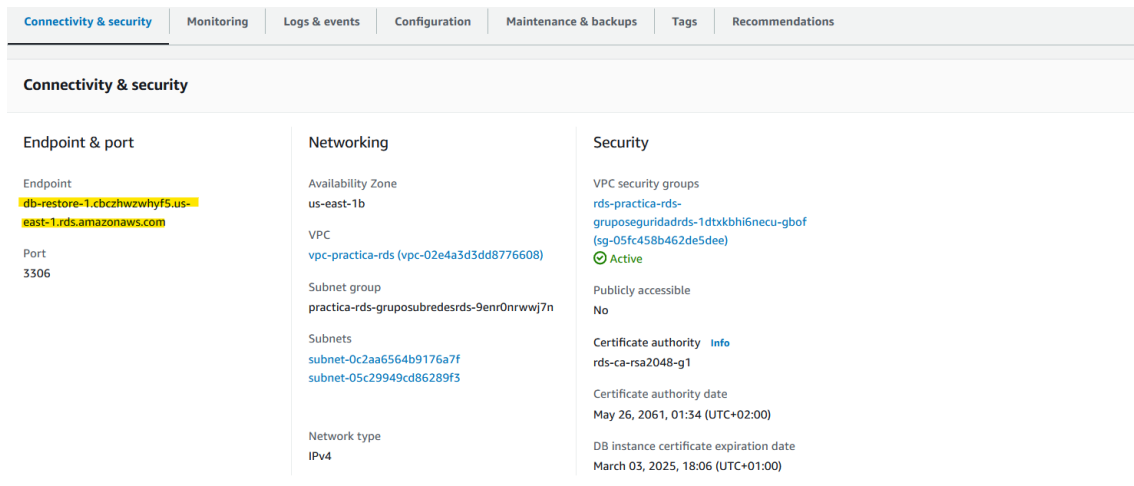
Backup

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

Por último, presionamos el botón **Restore Point in Time**. Tras ello volveremos a la ventana de instancias RDS y podremos visualizar que la nueva instancia se está creando. En unos minutos se encontrará operativa:



- 15) Accedemos la instancia etiquetada como *db-restore-1* y desde el apartado **Connectivity & security** copiamos el punto de enlace:



- 16) Volvemos a nuestra sesión en el intérprete de órdenes de nuestra instancia EC2 e intentamos la conexión a la instancia RDS restaurada ejecutando las siguientes órdenes:

```
RDS=<punto-enlace-BD-restaurada>
USUARIO=<nombre-usuario-RDS>
PASSWORD=<contraseña-RDS>

mysql -u $USUARIO -p$PASSWORD -h $RDS
```

Posteriormente, una vez establecida la conexión con la BD, podemos visualizar los contenidos de la BD *sakila*:

Session ID: user2881841=Test_student-0ef620bd1678a1795 Instance ID: i-049bbc3d6e02b62fc

```
sh-4.2$ mysql -u $USUARIO -p$PASSWORD -h $RDS
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 17
Server version: 10.11.6-MariaDB managed by https://aws.amazon.com/rds/
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> use sakila;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

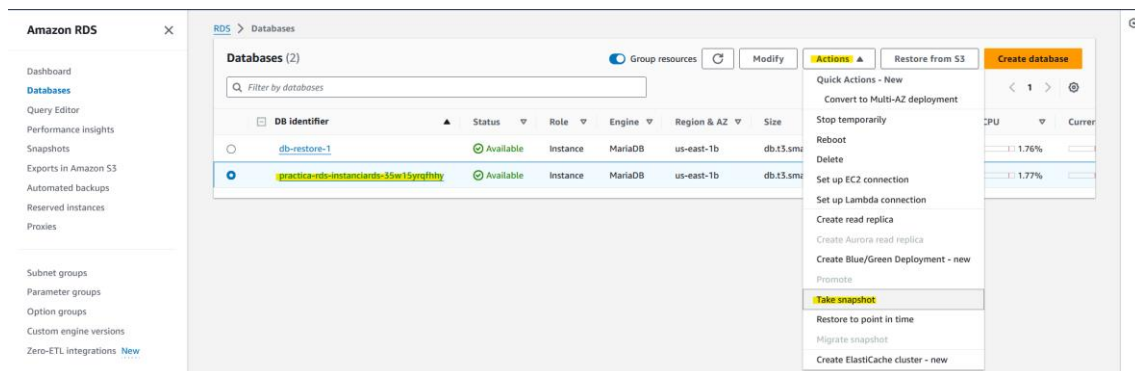
Database changed
MariaDB [sakila]> show tables;
+-----+
| Tables_in_sakila |
+-----+
| actor              |
| actor_info         |
| address            |
| category           |
| city               |
| country            |
| customer           |
| customer_list      |
| film               |
| film_actor         |
| film_category      |
| film_list          |
| film_text          |
| inventory          |
| language           |
| nicer_but_slower_film_list |
| payment            |
| rental             |
| sales_by_film_category |
+-----+
```

INSTANTÁNEAS MANUALES DE AMAZON RDS

Otra opción para respaldar una instancia de Amazon RDS son las instantáneas (*snapshots*). Una instantánea de una instancia RDS captura el estado de dicha instancia en un momento puntual en el tiempo. Las instantáneas se inicial manualmente por el usuario y se realizan incrementalmente, almacenándose en un bucket de Amazon S3 administrado por AWS.

Las instantáneas, al contrario que los respaldos automatizados no expiran con el tiempo automáticamente. Es responsabilidad de los usuarios eliminar las instantáneas, por lo que no están vinculadas al ciclo de vida de la instancia RDS. Por esta misma razón, sólo sirven para restaurar la BD al instante en el que se tomó la instantánea, no disponiendo de la capacidad de **PITR** (*Point in Time Recovery*) de los respaldos automáticos.

- 17) Para crear una instantánea manual de nuestra instancia RDS original, accedemos a la consola de Amazon RDS, seleccionamos nuestra instancia y, accedemos al menú **Actions / Take snapshot**:



- 18) Desde la ventana siguiente, en el campo **Snapshot type** seleccionamos el valor **DB instance** y seleccionamos nuestra instancia RDS. Indicamos en el campo **Snapshot name** el valor *instantanea-1*:

Take DB Snapshot

Preferences
To take a DB Snapshot, choose a DB Instance and name your DB Snapshot.

Snapshot type
☒ **DB instance**
☐ DB cluster

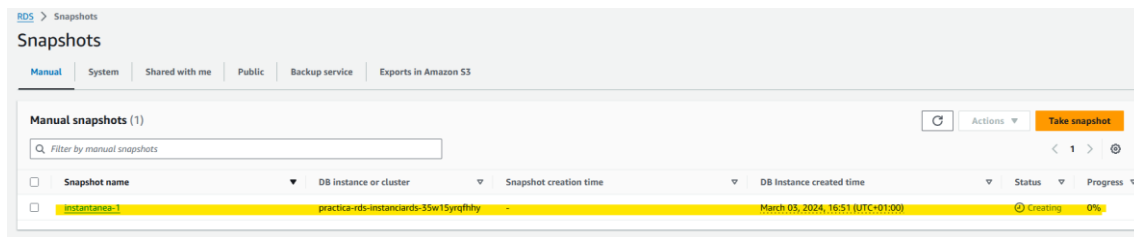
DB instance
DB Instance identifier. This is the unique key that identifies a DB Instance.
 practica-rds-instanciads-35w15yqfthly

Snapshot name
Identifier for the DB Snapshot.
 instantanea-1

Snapshot identifier is case insensitive, but stored as all lower-case, as in "mysnapshot". Cannot be null, empty, or blank. Must contain from 1 to 255 alphanumeric characters or hyphens. First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

Cancel Take snapshot

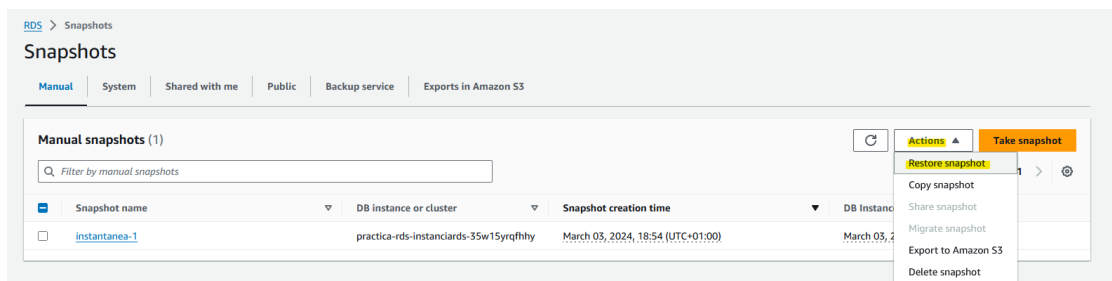
Por último, presionamos el botón **Take snapshot**. Tras esto, aparecerá una ventana con la instantánea que se está creando. Esperamos unos minutos a que se complete:



19) Las instantáneas de RDS son un recurso muy versátil, ya que permiten:

- Compartirlas con otras cuentas de AWS o hacerlas públicas
- Exportarlas a un bucket de Amazon S3 administrado por el cliente en formato Apache Parquet para posteriormente, poder consultar las BDs almacenadas mediante Amazon Athena.
- Copiarlas a otras regiones de AWS
- Restaurar una nueva instancia RDS a partir de la instantánea

En nuestra práctica, vamos a crear una nueva instancia RDS a partir de la instantánea creada. Para ello seleccionamos la opción **Actions / Restore snapshot**:



20) Desde la siguiente ventana, configuramos la creación de la nueva instancia RDS indicando los siguientes parámetros:

- En el apartado **Settings**, indicamos en el campo **DB instance identifier** el valor `db-restore-2`

Settings

DB snapshot ID

The identifier for the DB snapshot.

`rds:practica-rds-instances-35w15yrqfhhy-2024-03-03-16-40`

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.

`db-restore-2`

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.

- En el apartado **Instance configuration**, seleccionamos la opción **Burstable classes (including t classes)** y desde el menú desplegable elegimos la opción `db.t3.small`

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

- ☒ Include previous generation classes
- ☐ Standard classes (includes m classes)
- ☐ Memory optimized classes (includes r and x classes)
- ☒ Burstable classes (includes t classes)

db.t3.small
2 vCPUs 2 GiB RAM Network: 2085 Mbps

- c) En el apartado **Connectivity**, en la sección **VPC security group (firewall)** seleccionamos la opción *Existing VPC security groups* y elegimos desde el menú desplegable el grupo de seguridad que comienza por *rds-practica* y eliminamos el grupo de seguridad *default*

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

- ☐ practica-rds-GrupoSeguridadEC2-GPTD018EOFMF
- ☒ **rds-practica-rds-gruposeguridadrds-1dtxbhi6necu-qbof**
- ☐ default

Por último, presionamos el botón **Restore DB instance**. Inmediatamente volveremos a las instancias RDS, donde visualizaremos el estado de creación de la nueva instancia RDS etiquetada como *db-restore-2*. Tras unos minutos quedará operativa:

RDS > Databases										
Databases (3)										
Filter by databases										
<input type="checkbox"/>	DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendations	CPU	Current activity	Ma
<input type="radio"/>	db-restore-1	Available	Instance	MariaDB	us-east-1b	db.t3.small		2.18%	0 Connections	nor
<input checked="" type="radio"/>	db-restore-2	Creating	Instance	MariaDB	us-east-1b	db.t3.small		-		nor
<input type="radio"/>	practica-rds-instanciardb-35w15yqfhy	Available	Instance	MariaDB	us-east-1b	db.t3.small	4 Informational	2.13%	0 Connections	nor

- 21) Repetimos los pasos anteriores 15) y 16) pero en esta ocasión con la nueva instancia RDS restaurada:

```

db-4.25 USER@db-4.25:~$ mysql -u root -p
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 15
Server version: 10.11.6-MariaDB-log managed by https://www.amazon.com/rds/
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> use sakila;
Reading table information for completion of table and column names
You can turn off this feature with 'set -A'

Database changed
MariaDB [(none)]> show tables;
+-----+
| Tables_in_sakila |
+-----+
| actor |
| actor_info |
| address |
| category |
| city |
| country |
| customer |
| customer_list |
| film |
| film_actor |
| film_category |
| film_list |
| film_text |
| inventory |
| language |
| newer_but_slower_film_list |
| payment |
| rental |
| sales_by_film_category |
| sales_by_store |
| staff |
| staff_list |
| store |
+-----+
23 rows in set (0.01 sec)

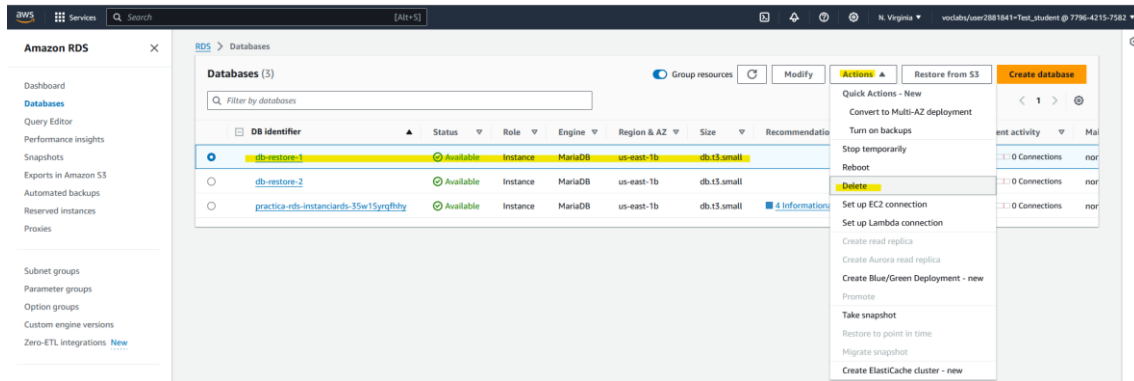
MariaDB [(none)]>

```

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 las instancias RDS con identificadores *db-restore-1* y *db-restore-2*. Para ello, desde la consola de Amazon RDS, seleccionamos la opción **Databases** en el menú lateral y, seleccionando la instancia RDS, presionamos el botón **Actions / Delete**:



A continuación, en la siguiente ventana desactivamos la opción **Create final snapshot** y marcamos la opción para confirmar la eliminación de la instancia RDS:

Delete db-restore-1 instance

Permanently delete **db-restore-1** 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.

☒ 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.

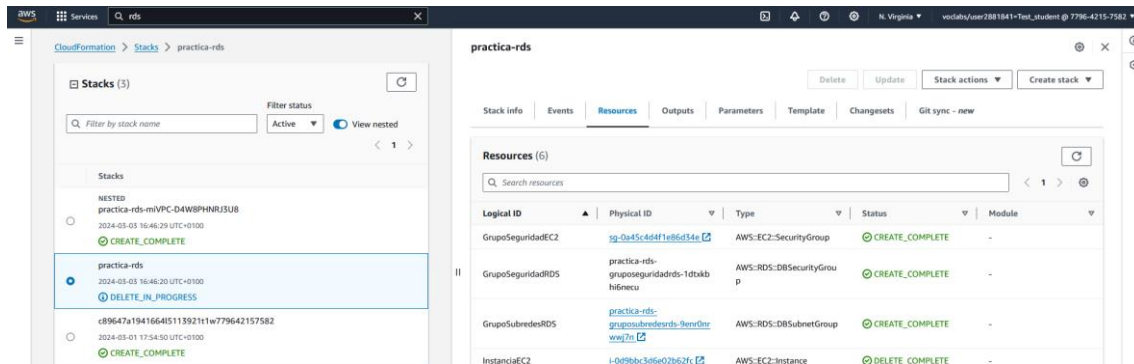
Cancel

Delete

- Desaprovisionar la infraestructura creada en la práctica. Para ello, desde la AWS CloudShell, ejecutamos la siguiente orden:

```
aws cloudformation delete-stack --stack-name practica-rds
```

El desaprovisionamiento de la infraestructura puede durar varios minutos, podemos seguirlo desde la consola del servicio de AWS CloudFormation:



- Eliminar la instantánea creada, para ello desde la consola de Amazon RDS elegimos la opción **Snapshots** del menú lateral, seleccionamos la instantánea y desde presionamos el botón **Actions / Delete snapshot**:

