

## DAW Práctica 3.2: Creación de BD y despliegue de aplicación en AWS

**Elabora un documento donde figuren todos los pasos realizados con las pantallas significativas, explicando cada uno de los pasos.**

En este ejercicio vamos a crear una instancia de base de datos MySQL administrada por AWS, conectar con la instancia EC2 creada y desplegar una aplicación web que utiliza la Base de datos.

Amazon Relational Database Service (Amazon **RDS**) facilita las tareas de configuración, operación y escalado de una base de datos relacional en la nube. Proporciona una capacidad rentable y de tamaño modificable, al mismo tiempo que permite gestionar las tareas de administración de base de datos que requieren mucho tiempo, lo que permite centrarse en las aplicaciones y el negocio. Amazon RDS ofrece seis motores de base de datos familiares entre los que elegir: Amazon Aurora, Oracle, Microsoft SQL Server, PostgreSQL, MySQL y MariaDB.

### **Procedimiento:**

1. Crearemos un grupo de seguridad para permitir que el servidor web acceda a la instancia de base de datos de RDS. El grupo de seguridad se utilizará al lanzar la instancia de base de datos. El grupo de seguridad tendrá la siguiente configuración:
  - **Nombre del grupo de seguridad:** grupoSeguridadDB
  - **Descripción:** Permitir el acceso a la BD
  - **VPC:** VPC de laboratorio creada

También se agregará una regla al grupo de seguridad para permitir las solicitudes de base de datos entrantes. En este caso queremos permitir el acceso desde el grupo de seguridad creado en la práctica 1.4. La regla de entrada contará con los siguientes ajustes:

2. **Tipo:** *MySQL/Aurora (3306)*
3. **Bloques CIDR, IP, grupos de seguridad or lista de prefijos:** escriba sg y luego seleccione *Grupo de seguridad web*.

Así se configura el grupo de seguridad de base de datos para permitir el tráfico entrante en el puerto 3306 desde cualquier instancia EC2 asociada al *Grupo de seguridad web*.

4. Ahora debemos crear otra zona de disponibilidad por ejemplo en **us-east-1b**. Para ello, crear una subred pública con un **10.0.2.0/24** y una subred privada **10.0.3.0/24** en dicha zona de disponibilidad.

Creamos una subred con la siguiente configuración:

- **VPC:** *VPC de laboratorio*
- **Nombre:** lab-subnet-public2-us-east-1b
- **Zona de disponibilidad:** us-east-1b
- **Bloque CIDR:** 10.0.0.0/16
- **Bloque de CIDR de la subred:** 10.0.2.0/24 y agregamos otra subred con las mismas características excepto, nombre lab-subnet-private2-us-east-1b y 10.0.3.0/24.

Crearemos un *grupo de subredes de base de datos* que se emplea a fin de informar a RDS acerca de qué subredes se pueden utilizar para la base de datos. Cada grupo de subredes de base de datos requiere subredes en al menos dos zonas de disponibilidad.

5. En el menú Servicios, haga clic en **RDS** y posteriormente, en el panel de navegación izquierdo, haga clic en **Grupos de subredes** para crear un grupo con la siguiente configuración:

- **Nombre:** grupo de subredes de base de datos
- **Descripción:** grupo de subredes de base de datos
- **VPC:** *VPC de laboratorio*
- **Zonas de disponibilidad:** las dos primeras (**us-east-1a** y **us-east-1b**)
- **Subredes** y seleccionamos las que están asociadas con los rangos de CIDR **10.0.1.0/24** y **10.0.3.0/24** (las privadas).

Utilizaremos este grupo de subredes de base de datos en la creación de la base de datos que crearemos en el siguiente punto.

Continuamos configurando y lanzando una instancia de base de datos Multi-AZ de Amazon RDS para MySQL.

Las implementaciones **Multi-AZ** de Amazon RDS proporcionan mejoras en la disponibilidad y la durabilidad de las instancias de base de datos, lo que las hace adecuadas para las cargas de trabajo de bases de datos de producción. Cuando aprovisiona una instancia Multi-AZ de base de datos, Amazon RDS crea automáticamente una instancia de base de datos principal y, de forma sincronizada, replica los datos a una instancia en espera en una zona de disponibilidad diferente.

6. Volvemos a AWS y creamos una BD configurando las siguientes características:


- Método de creación de base de datos: **creación estándar**.
- Opciones del motor: **MySQL**.

- Plantillas: **Producción**.
- Disponibilidad y durabilidad: **instancia de base de datos Multi-AZ**.
- **Configuración** configuramos lo siguiente:
  - Identificador de instancias de bases de datos: lab-db
  - Nombre de usuario maestro: admin
  - Administración de credenciales: Autoadministrado
  - Contraseña maestra: lab-password
  - Confirmar contraseña: lab-password
- **Configuración de la instancia** configuramos lo siguiente:
  - **Clases con ráfagas (incluye clases t)** .
  - Selecciona *db.t3.micro*
- **Almacenamiento**, configuramos lo siguiente:
  - **Tipo de almacenamiento**: *SSD de uso general (gp3)*
  - **Almacenamiento asignado**: 20
- En **Conectividad**, configuramos lo siguiente:
  - **Nube privada virtual (VPC)**: La VCP creada.
  - **Grupos de subredes de la base de datos**: el recientemente creado.
  - **Acceso público**: si (normalmente esta propiedad estará a no pero para permitir que nos conectemos desde fuera a la BD con el MySQLWorkbench lo vamos a poner público)
  - En **grupo de seguridad VPC firewall** vamos a crear uno nuevo para permitir la conexión exterior nombre grupo acceso exterior bd
- En **Configuración adicional** configuramos lo siguiente:
  - **Nombre de base de datos inicial**: lab
  - Desactivar **Habilitar copias de seguridad automáticas**
  - Desactivar **Habilitar el cifrado**.
  - Desactivar **Habilitar actualización automática de versiones secundarias**

Esto desactivará las copias de seguridad, lo que no suele recomendarse, pero permitirá una implementación más rápida de la base de datos para este ejercicio.

7. Una vez creada, hacer clic en lab-db (haga clic en el propio enlace). Esperar unos minutos para que se habilite la disponibilidad de la base de datos. El proceso de implementación implica la implementación de una base de datos en dos zonas de disponibilidad diferentes. (Esperar hasta que **Estado** cambie a **Disponible**.)

**lab-db**

 **Modificar** **Acciones** ▼

**Resumen**

|   |                                |                         |                                  |                        |
|---|--------------------------------|-------------------------|----------------------------------|------------------------|
| <b>Identificador de base de datos</b><br>lab-db | <b>Estado</b><br>⌚ Modificando | <b>Rol</b><br>Instancia | <b>Motor</b><br>MySQL Community  | <b>Recomendaciones</b> |
| <b>CPU</b><br>-                                 | <b>Clase</b><br>db.t3.micro    | <b>Actividad actual</b> | <b>Región y AZ</b><br>us-east-1b |                        |

**Estado**  
✔ **Disponible**

**Clase**  
db.t3.micro

8. Ahora deberemos hacer que La instancia de RDS creada tenga acceso desde el exterior a través de MySQLWorkbench.

Esto no es habitual porque como hemos visto, la base de datos ha sido creada en las redes privadas, es decir, sin acceso desde el exterior. Un paso que hemos realizado anteriormente es crear la base de datos con **acceso público** y ahora tendremos que abrir el paso de acceso a través de internet (se añadió un nuevo grupo de seguridad *grupo acceso exterior bd*)

Al grupo de seguridad creado le añadimos una nueva regla de entrada para que permita la conexión desde cualquier ip por el puerto 3306. Otra opción más segura es crear un *NAT Gateway*.

- a. Volvemos al panel de VPC y accedemos al *grupo acceso exterior bd* para configurarlo.
- b. Editamos sus reglas de entrada para permitir su entrada desde cualquier IPv4 (Anywhere-IPv4).

9. Ahora asignaremos a una de las subredes privadas, del grupo de subredes creado, para acceso a internet.
- Dentro del panel de RDS nos dirigimos a la sección grupo de subredes. Y entramos en el grupo creado previamente.
  - En el podemos ver cada una de las subredes que pertenecen al grupo. Deberemos entrar en cada una de ellas (2) y modificar lo siguiente.

| Subredes (2)           |  |  |                |
|------------------------|--|--|----------------|
| Zona de disponibilidad | Subnet name                              | ID de subred                             | Bloque de CIDR |
| us-east-1b             | PRUEBA-lab-subnet-private2-us-east-1b    | <a href="#">subnet-045c84578b8d1ba10</a> | 10.0.3.0/24    |
| us-east-1a             | LabPruebaDAW2-subnet-private1-us-east-1a | <a href="#">subnet-0abb1d84fee606ad0</a> | 10.0.1.0/24    |

- Dentro de cada subred se puede acceder a su tabla de enrutamiento.

### subnet-045c84578b8d1ba10 / PRUEBA-lab-subnet-private2-us-east-1b

| Detalles   | Registros de flujo | Tabla de enrutamiento | ACL de red | Reservas de CIDR |
|--|--------------------|-----------------------|------------|------------------|
| Tabla de enrutamiento: <a href="#">rtb-08d8896875604aca0</a> |                    |                       |            |                  |

- Dentro de estas tablas, en la sección "Rutas" se pueden editar.

| Detalles                                   | Rutas                                 | Asociaciones de subredes | Asociaciones de borde | Propagación de rutas | Etiquetas |
|--|---------------------------------------|--------------------------|-----------------------|----------------------|-----------|
| Rutas (2)                                  |                                       |                          |                       |                      |           |
| <input type="text" value="Filtrar rutas"/> |                                       |                          |                       |                      |           |
| Destino                                    | Destino                               | Estado                   | Propagada             |                      |           |
| 0.0.0.0/0                                  | <a href="#">igw-0d20118b29237fe49</a> | ✓ Activo                 | No                    |                      |           |
| 10.0.0.0/16                                | local                                 | ✓ Activo                 | No                    |                      |           |

- Añadimos una nueva ruta para que tenga acceso a internet.

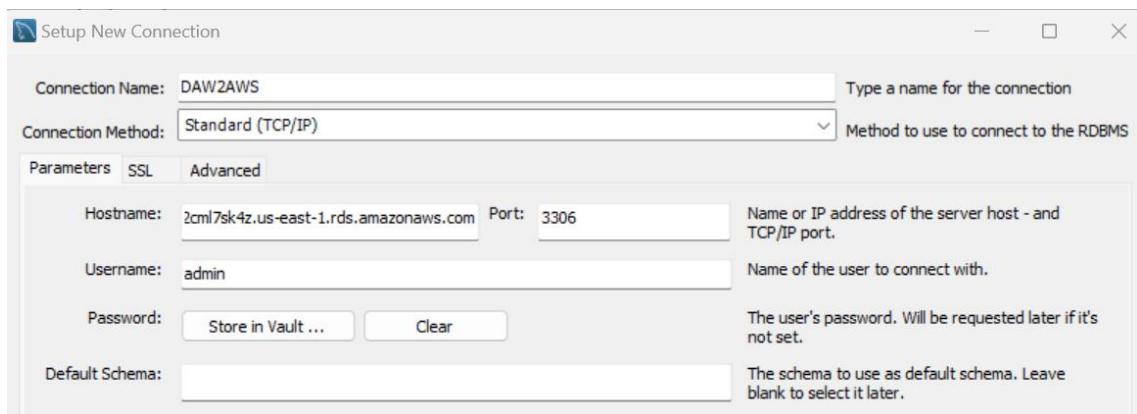
| Destino                                     | Destino   | Estado   |
|---|---|----------|
| 10.0.0.0/16                                 | local   | ✓ Activo |
| <input type="text" value="0.0.0.0/0"/>      | <input type="text" value="local"/>                          | -        |
|   | <input type="text" value="Puerta de enlace de Internet"/>   |          |
|   | <input type="text" value="igw-07c1e5d8c7c159852"/>          |          |
|   | Utilizar: "igw-07c1e5d8c7c159852"                           |          |
|   | <a href="#">igw-07c1e5d8c7c159852</a> (LaboratorioDAW2-igw) |          |
| <input type="button" value="Agregar ruta"/> |   |          |

10. Una vez tengamos configurado el enrutador con la puerta de enlace de internet podemos configurar el acceso desde MySQLWorkbench. Los datos que necesitamos de la RDS creada son:

- **Punto de enlace & puerto**, lo encontramos una vez seleccionada la base de datos en el apartado conectividad & seguridad.
- **Nombre del usuario maestro**, lo encontramos una vez seleccionada la base de datos en el apartado Configuración y la password dada a la base de datos.

11. Configuramos la conexión en MySQLWorkbench:

- **Hostname**, ponemos el valor de punto de enlace de la base de datos
- **port**, ponemos el valor de puerto de la base de datos.
- **Username**, ponemos el valor de nombre del usuario maestro de la base de datos.
- **Password**, ponemos el valor de la password dada en la creación de la base de datos.



Setup New Connection

Connection Name: DAW2AWS Type a name for the connection

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname: 2cm17sk4z.us-east-1.rds.amazonaws.com Name or IP address of the server host - and TCP/IP port.

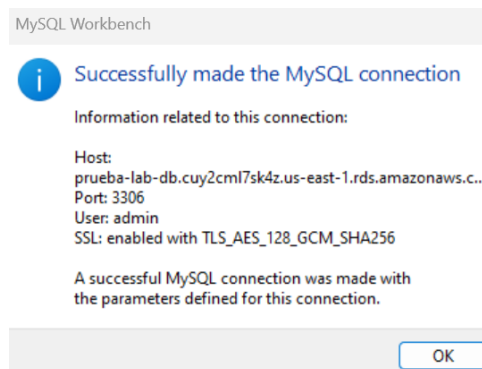
Port: 3306

Username: admin Name of the user to connect with.

Password:   The user's password. Will be requested later if it's not set.

Default Schema:  The schema to use as default schema. Leave blank to select it later.

12. Probamos la conexión.



13. Elegimos una aplicación simple que tengamos del curso y la vamos a desplegar en AWS.

14. Para ello, la comprimimos en un zip y utilizamos la siguiente instrucción desde la consola de Windows.

```
scp -i "labsuser.pem" nombreAplicacionPHP.zip ubuntu@ec2-34-207-226-176.compute-1.amazonaws.com:/var/www/miWeb
```

Recuerda modificar la **IP de tu máquina**, el **nombre del zip** donde tienes la aplicación y la **ruta de la raíz de tu host virtual**.

15. Te deja subir el archivo a esa ruta ¿Por qué? ¿Qué deberías hacer para solucionarlo?

16. Descomprime el zip y prueba el funcionamiento de tu aplicación desde AWS.