

Santiago Hernández Ramos

Gestión de Base de Datos

Despliegue de una aplicación web completa en la nube

Introducción

En este ejercicio práctico se propone el despliegue de una aplicación web en AWS que debe cumplir con el siguiente diagrama de arquitectura:



La aplicación web que debe desplegarse en la nube se corresponde con la siguiente:

<https://github.com/aws-samples/simple-phonebook-web-application>

Para obtener la máxima calificación el alumno deberá completar satisfactoriamente todos los apartados que se indican en el enunciado.

IMPORTANTE: El alumno deberá documentar todos los pasos que ha realizado en un informe técnico. El informe debe tener evidencias de que ha sido el alumno el autor de las actividades (capturas de pantalla). Cada una de las secciones debe estar correctamente documentada con una pequeña descripción de las tareas realizadas por el alumno con las evidencias pertinentes, en caso contrario, la puntuación para esa sección será la mínima.

IMPORTANTE: El alumno es responsable de la eliminación de los recursos creados en su cuenta de AWS siguiendo las instrucciones que se indican en la última sección de la práctica. En caso de no eliminar los recursos creados o crear recursos adicionales a los que se proponen en el enunciado de esta práctica, el alumno debe ser consciente de que pueden repercutir en un coste económico del que la escuela no se hace responsable.

Enunciado

Configuración del acceso y seguridad

Cuando realizamos el despliegue de una aplicación web que incorpora diferentes componentes como una interfaz web, un servidor de aplicación, una base de datos... Debemos asegurarnos de implementar seguridad en cada una de las capas: en la capa de presentación, en la capa de aplicación, en la capa de datos y en la red.

Para este propósito vamos a utilizar un elemento de AWS conocido como *Security Group* o Grupo de seguridad. Un grupo de seguridad nos permite controlar el tráfico de red entrante y saliente de una instancia de AWS, actuando como si fuese un firewall virtual. Para gestionar este acceso, nos permite añadir diferentes reglas.

Lo primero que vamos a hacer es crear un grupo de seguridad y configurarlo para que permita conexiones HTTP desde Internet a nuestra instancia en la nube.

1. En la consola de administración de AWS pulsamos sobre el servicio **"EC2"**
2. Dentro del panel de administración de EC2, en el menú de la izquierda, pulsamos sobre **"Security Groups"**
3. Podemos observar que ya existe algún grupo de seguridad, en este caso se corresponde con un grupo que viene por defecto. Creamos un nuevo grupo pulsando sobre **"Crear grupo de seguridad"**
4. Completamos el formulario con la siguiente información:
 - a. Nombre del grupo de seguridad: **"WebApp-SG"**
 - b. Descripción: **"Permitir acceso HTTP"**
 - c. VPC: **Dejamos el valor por defecto.** (Lo ideal es crear una VPC para cada una de las aplicaciones web que queramos alojar en la nube. Veremos más en detalle lo que es una VPC en el siguiente módulo)
5. En la sección **"Reglas de entrada"** pulsamos sobre **"Agregar regla"** y la configuramos de la siguiente forma:
 - a. Tipo: **"HTTP"**
 - b. Origen: **"Anywhere-IPv4"**
6. En la parte inferior de la página pulsamos sobre **"Crear grupo de seguridad"**.

En este punto del ejercicio hemos creado el grupo de seguridad que nos permitirá acceder desde internet a nuestra aplicación web. Sin embargo, todavía nos queda crear el grupo de seguridad que permitirá el acceso desde nuestra aplicación web a la base de datos.

1. Creamos un nuevo grupo de seguridad con los siguientes detalles:
 - a. Nombre del grupo de seguridad: **"DB-SG"**
 - b. Descripción: **"Permitir acceso DB"**
 - c. VPC: **Dejamos el valor por defecto.**
2. En la sección **"Reglas de entrada"** pulsamos sobre **"Agregar regla"** y la configuramos de la siguiente forma:
 - a. Tipo: **"MYSQL/Aurora"**
 - b. Origen: **"Personalizada"** y seleccionamos el Grupo de seguridad **"WebApp-SG"**

3. En la parte inferior de la página pulsamos sobre **“Crear grupo de seguridad”**.

Como puede observarse hemos establecido las reglas de acceso relacionándolas con los grupos de seguridad que hemos creado. Esta es una de las características que nos proporciona Amazon para evitar que tengamos que lidiar a bajo nivel con la infraestructura de red.

Creación de la base de datos

En ejercicios anteriores ya hemos visto la complejidad que tiene el despliegue y administración de una base de datos en local, en este apartado vamos a configurar una base de datos MySQL utilizando el servicio de Amazon RDS.

1. Accedemos al panel de control del servicio de AWS **"RDS"**
2. Pulsamos en **"Crear base de datos"**
3. En la sección **"Elegir un método de creación de base de datos"** elegimos **"Creación estándar"**
4. En la sección **"Opciones del motor"** seleccionamos **"MySQL"**
5. En el campo **"versión"** seleccionamos **"MySQL 5.7.34"**
6. En la sección **"Plantillas"** seleccionamos **"Desarrollo y pruebas"**
7. En la sección de **"Configuración"** indicamos los siguientes valores:
 - a. Identificador de instancias de base de datos: **"phonebook-db"**
 - b. Nombre de usuario maestro: **"dbadmin"**
 - c. Contraseña maestra: **<Introduce una contraseña robusta>**
 - d. Confirmar contraseña: **<Introduce una contraseña robusta>**
8. En la sección **"Clase de instancia de base de datos"** seleccionamos **"Clases con ráfagas"**, marcamos la opción **"Incluir clases de generación anterior"** y seleccionamos **"db.t2.micro"**.
9. En la sección de **"Disponibilidad y durabilidad"** podemos crear una instancia en espera tal y como hemos visto en la teoría. En este caso práctico en particular, no vamos a utilizar esta funcionalidad ya que se corresponde con un entorno de pruebas.
10. En la sección **"Conectividad"**, en el apartado **"Grupos de seguridad de VPC existentes"**, eliminamos **"default"** y añadimos **"DB-SG"**
11. En la sección **"Configuración adicional"**, en el apartado **"Nombre de base de datos inicial"** introducimos **"phonebook"**. Este nombre se corresponde con un nombre lógico que utilizará la aplicación para referenciar la base de datos.
12. Desactivamos la opción **"Habilitar la monitorización mejorada"**.
13. Pulsamos en **"Crear base de datos"**.

Despliegue de un servidor de aplicación

En este punto ya tenemos desplegada nuestra base de datos y asociado el grupo de seguridad que permite el acceso desde un servidor de aplicación. Lo que vamos a hacer en esta sección del ejercicio es crear un servidor Linux en nuestra cuenta de Amazon en la nube e instalar un servidor de aplicación que aloje una aplicación web que se conectará a la base de datos que hemos creado previamente.

1. Accedemos al servicio **"EC2"** desde nuestra consola de administración de AWS y pulsamos sobre **"Panel de EC2"** en el menú de la izquierda.
2. Pulsamos sobre **"Lanzar la instancia"**
3. Para el propósito de este ejercicio vamos a desplegar un servidor Linux por lo tanto seleccionamos la opción **"Amazon Linux 2 AMI"**
4. En el Paso 2 seleccionamos la opción **"t2.micro"**, asegurándonos que es **"Apto para la capa gratuita"** y pulsamos sobre **"Siguiente: Página Configuración de los detalles de la instancia"**
5. En el Paso 3, pulsamos sobre la sección **"Detalles avanzados"** y en el campo **"Datos de usuarios"** introducimos el siguiente script:

```
#!/bin/bash
yum -y install httpd php mysql php-mysql

case $(ps -p 1 -o comm | tail -1) in
systemd) systemctl enable --now httpd ;;
init) chkconfig httpd on; service httpd start ;;
*) echo "Error starting httpd (OS not using init or systemd)." 2>&1
esac

if [ ! -f /var/www/html/bootcamp-app.tar.gz ]; then
cd /var/www/html
wget https://s3.amazonaws.com/immersionday-labs/bootcamp-app.tar
tar xvf bootcamp-app.tar
chown apache:root /var/www/html/rds.conf.php
fi
yum -y update
```

6. Pulsamos sobre **"Siguiente: Adición de almacenamiento"**. En este punto podríamos añadir un volumen de Amazon EBS si quisiésemos aumentar la capacidad de almacenamiento y el rendimiento. Para este ejercicio en particular vamos a mantenerlo por defecto y pulsamos sobre **"Siguiente: Agregar etiquetas"**
7. Pulsamos sobre **"Añadir etiqueta"** e introducimos los siguientes valores:
 - a. Clave: **"Name"**
 - b. Valor: **"Webserver"**
8. Pulsamos sobre **"Siguiente: Página Configure Security Group"**
9. Seleccionamos **"Seleccionar un grupo de seguridad existente"** y marcamos **"WebApp-SG"**
10. Pulsamos sobre **"Revisar y lanzar"**
11. Nos aparecerá un mensaje indicando que no podremos conectarnos por ssh a la instancia que vamos a crear pulsamos en **"Continuar"**.
12. Revisamos todos los datos y pulsamos en **"Lanzar"**

13. Nos aparecerá de nuevo una ventana emergente indicando que seleccionemos un par de claves para conectarnos por ssh, en este caso no vamos a necesitar conectarnos por ssh por lo tanto seleccionamos la opción **“continuar sino un par de claves”** y le pulsamos sobre lanzar instancias.

En este punto del ejercicio nuestro servidor web Linux debería de estar iniciándose y deberían instalarse todos los componentes que le hemos indicado en el script inicial. Para comprobar el estado en el que se encuentra el servidor podemos irnos al panel de EC2 y pulsar sobre **“instancias”**.

Verificando el funcionamiento de la aplicación

1. En el panel de administración de EC2 seleccionamos el servidor web que hemos creado previamente
2. En la pestaña **“Detalles”** copiamos el campo **“DNS de IPv4 pública”** y lo copiamos en una nueva pestaña de nuestro navegador. El resultado debería ser una página web similar a la que se muestra a continuación.

LOAD TEST	RDS
Meta-Data	Value
Instanceld	<i>i-0056a81157286c47c</i>
Availability Zone	<i>us-east-2c</i>

Current CPU Load: **100%**

3. Pulsamos sobre **“RDS”** en la página web y nos pedirá los detalles de nuestra base de datos en la nube para establecer la conexión.
4. Accedemos al panel de administración del servicio **“RDS”** y pulsamos sobre la instancia de base de datos que hemos creado previamente.
5. En la pestaña **“conectividad y seguridad”** copiamos el valor del campo **“Punto de enlace”**. (Debería ser algo similar a lo siguiente: *phonebook-db.caayrsrdlchq.us-east-2.rds.amazonaws.com*)
6. Introducimos los datos en la aplicación web de la siguiente forma.

LOAD TEST	RDS
Endpoint:	<input type="text" value="phonebook-db.caayrsrdlchq.us-east-2.rds.amazonaws.com"/>
Database:	<input type="text" value="phonebook"/>
Username:	<input type="text" value="dbadmin"/>
Password:	<input type="password" value="....."/>
<input type="button" value="Submit"/>	

El resultado debería ser que la aplicación web se conecta a la base de datos que está en la nube y se muestra la siguiente pantalla.

LOAD TEST RDS

Address Book

Name	Phone	Email	Admin	
			Add Contact	
Alice	571-555-4875	alice@address2.us	Edit	Remove
Bob	630-555-1254	bob@fakeaddress.com	Edit	Remove

También podemos probar a añadir nuevos contactos a nuestra agenda y por lo tanto introducirnos en la base de datos pulsando sobre **“Add Contact”**

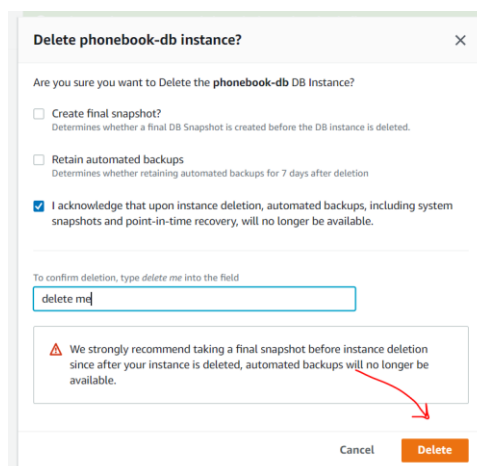
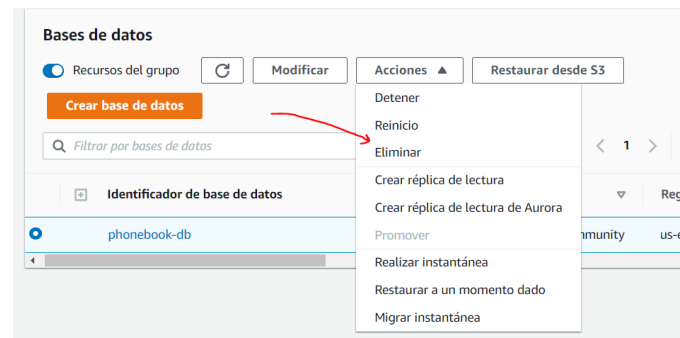
LOAD TEST RDS

Address Book

Name	Phone	Email	Admin	
			Add Contact	
Alice	571-555-4875	alice@address2.us	Edit	Remove
Bob	630-555-1254	bob@fakeaddress.com	Edit	Remove
Santiago	888888888	santiago@test.com	Edit	Remove

Limpieza del laboratorio

Para limpiar la infraestructura que hemos generado y asegurarnos de que no se cargue ningún importe en nuestra cuenta accedemos al panel de administración de Amazon RDS seleccionamos la base de datos que hemos creado y pulsamos sobre **“Acciones”** y **“eliminar”**



De igual manera, para eliminar el servidor web que hemos creado nos vamos al panel de administración de EC2, seleccionamos la instancia que hemos creado, pulsamos sobre **“Estado de la instancia”** y seleccionamos **“Terminar instancia”**.

