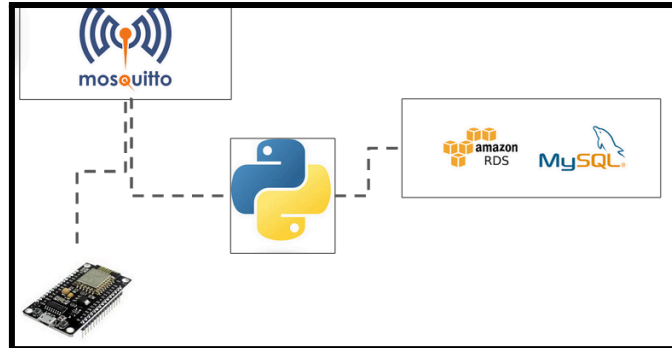


## Integración de Dispositivos IoT con AWS

En esta actividad, aprenderemos a integrar dispositivos IoT (Internet de las Cosas) con servicios en la nube de Amazon Web Services (AWS). Utilizaremos un ESP8266 o una placa Arduino para capturar datos de sensores y los enviaremos a un servidor MQTT Mosquitto. Luego, desarrollaremos un script en Python para procesar los datos y almacenarlos en una base de datos en una instancia de AWS.



### Configuración del Dispositivo IoT:

- Conecta los sensores al ESP8266 o a la placa Arduino y programa el dispositivo para enviar los datos al servidor MQTT Mosquitto.
- Configura el dispositivo para conectarse a la red Wi-Fi y al servidor MQTT.

### Configuración del Servidor MQTT Mosquitto:

- Instala y configura un servidor MQTT Mosquitto en una máquina accesible desde la red donde esté conectado el dispositivo IoT.

### Desarrollo del Script en Python:

- Crea un script en Python que se ejecute en una computadora (puede ser un servidor o un PC de usuario).
- Utiliza la biblioteca Paho MQTT para suscribirte al servidor MQTT Mosquitto y recibir los datos enviados por el dispositivo IoT.
- Procesa los datos recibidos según los requisitos del proyecto (análisis, filtrado, etc.).

### Conexión con AWS:

- Utiliza las bibliotecas de AWS SDK para Python para establecer una conexión con la instancia de AWS.
- Configura los permisos necesarios para acceder a los servicios de AWS desde el script.

### Almacenamiento de Datos en AWS:

- Implementa la lógica en el script para almacenar los datos procesados en la base de datos de la instancia de AWS (Amazon RDS o Amazon DynamoDB).

### Ejecución y Pruebas:

- Ejecuta el script en la computadora y monitorea la recepción, procesamiento y almacenamiento de los datos en AWS.
- Realiza pruebas para verificar la integridad y fiabilidad del sistema en diferentes condiciones.

## Instalación y Configuración de MySQL y PHPMyadmin

Para instalar MySQL en una instancia de EC2 en AWS y establecer una conexión remota, sigue estos pasos:

Actualiza la máquina:

- `apt-get update`

luego instala MySQL:

- `apt-get install mysql-server`

Verifica el estado del servicio MySQL

- `systemctl status mysql.`

Inicia sesión en MySQL con el usuario root y la contraseña predeterminada: root.

- `mysql -u root -p`

Crear una base de datos: Puedes crear una base de datos específica para tus datos. Por ejemplo, vamos a llamarla `sensores_db`.

- `CREATE DATABASE sensores_db;`

Crea el usuario:

- `CREATE USER 'adminsql'@'%' IDENTIFIED BY 'Admin123';`

Otorga los privilegios al usuario:

- `GRANT ALL PRIVILEGES ON sensores_db.* TO 'adminsql'@'%';`

Seleccionar la base de datos: Una vez creada la base de datos, selecciona esa base de datos para realizar las operaciones posteriores.

- `USE sensores_db;`

Crear una tabla para almacenar las lecturas del sensor: Define una tabla que tenga columnas para almacenar la fecha y hora de la lectura, la humedad y la temperatura. Por ejemplo:

```
CREATE TABLE lecturas (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  fecha_hora DATETIME,  
  humedad FLOAT,  
  temperatura FLOAT  
);
```

(NOTA AL CREAR LA TABLA, LE PUEDES PEDIR A CHATGPT QUE TE DE LAS INDICACIONES PARA CREAR UNA TABLA SEGÚN TUS NECESIDADES)

## Configuración de archivo `mysql.conf.d`

navega a la carpeta de configuración de MySQL utilizando

- `cd /etc/mysql/mysql.conf.d`

Edita el archivo de configuración `mysqld.cnf` con el comando

- `nano mysqld.cnf`

Busca y modifica la línea

**`bind-address`**, reemplazando la IP actual (127.0.0.1) por 0.0.0.0.

```
# socket            = /var/run/mysqld/mysqld.sock
# port              = 3306
# datadir           = /var/lib/mysql

# If MySQL is running as a replication slave, this should be
# changed. Ref https://dev.mysql.com/doc/refman/8.0/en/server-system-variables.html#sysvar_tmpdir
# tmpdir            = /tmp
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 0.0.0.0
mysqlx-bind-address = 127.0.0.1
#
# * Fine Tuning
```

Guarda los cambios con CTRL+X y confirma con Y.

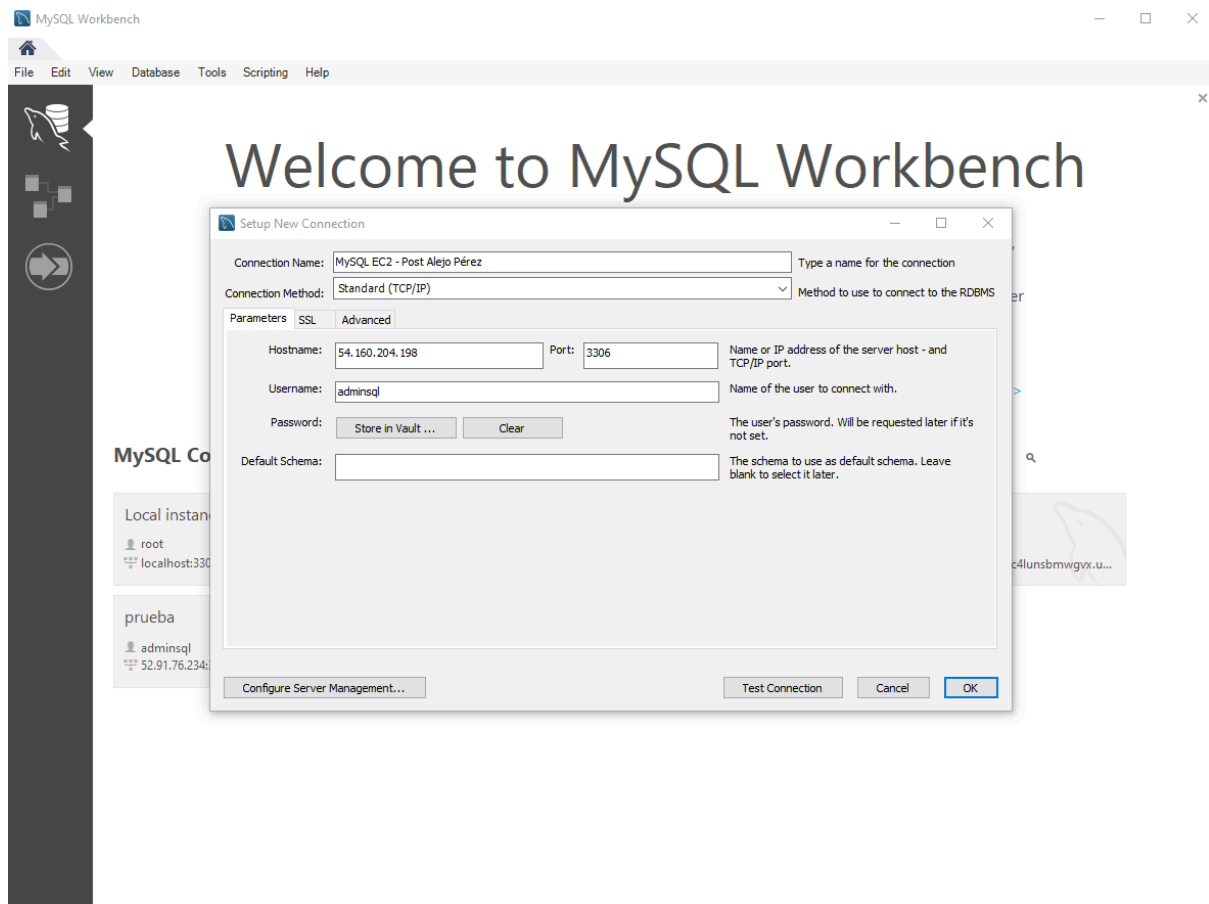
Reinicia el servidor MySQL para que los cambios surtan efecto:

- `systemctl restart mysql`

**Para probar la conexión remota, utiliza una herramienta de administración de bases de datos como MySQL Workbench.**

Crea una nueva conexión y proporciona la información necesaria:

- Nombre de conexión.
- Dirección IP pública de la instancia EC2.
- Puerto (por defecto, 3306).
- Nombre de usuario creado en MySQL (adminsqli en el ejemplo).
- Contraseña del usuario creado en MySQL.



Realiza una prueba de conexión y, si todo está correcto, podrás conectarte de forma remota al servidor MySQL instalado en el servidor

Siguiendo estos pasos, lograrás instalar MySQL en AWS y establecer una conexión remota para administrar la base de datos desde tu máquina local o cualquier otro lugar.

## instalación de phpmyadmin

**Actualiza el sistema:** Es una buena práctica asegurarse de que tu sistema esté actualizado antes de instalar nuevos paquetes. Ejecuta los siguientes comandos:

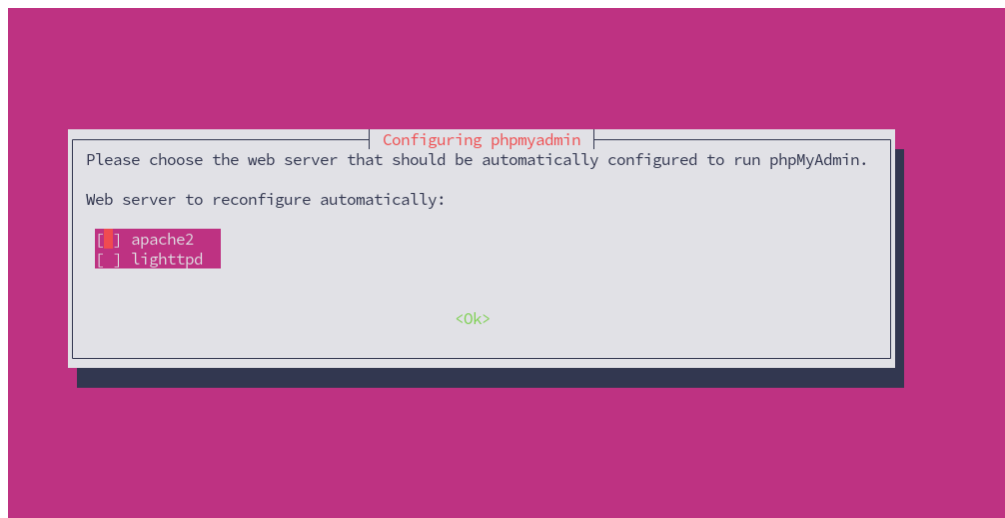
- `sudo apt update`
- `sudo apt upgrade`

**Instala PHPMyAdmin:** Puedes instalar PHPMyAdmin desde los repositorios de Ubuntu utilizando el administrador de paquetes apt. Ejecuta el siguiente comando:

- `sudo apt install phpmyadmin`

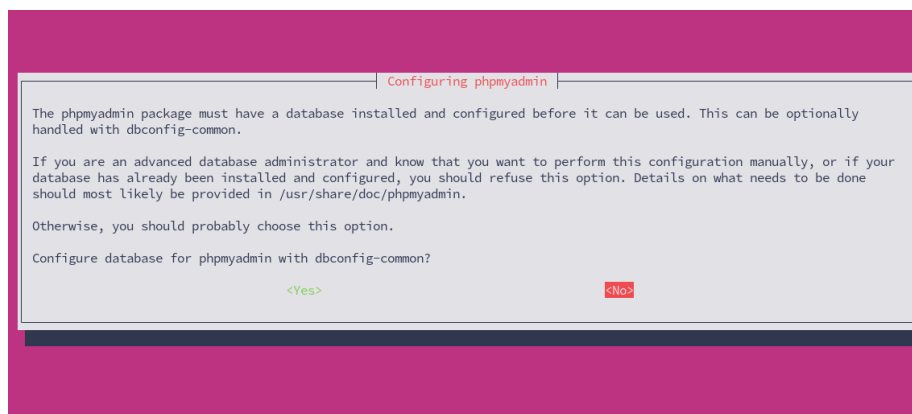
Durante la instalación de phpMyAdmin, se te pide que elijas el servidor web que se configurará automáticamente para ejecutar la aplicación, como Apache, Nginx o Lighttpd. Selecciona APACHE2.

(con la tecla “espacio” selecciona APACHE2)



Durante la instalación, se te pedirá que elijas el servidor web que estás utilizando selecciona PACHE. Luego selecciona NO para no configurar PHPMyAdmin con dbconfig-common.

Seleccionamos NO



**Accede a PHPMyAdmin:** Una vez instalado y configurado, puedes acceder a PHPMyAdmin abriendo un navegador web y navegando a `http://tu-direccion-ip/phpmyadmin`. Se te pedirá que inicies sesión con las credenciales de tu base de datos MySQL.

## **OPCIONAL:**

**Instala PHP:** Ejecuta el siguiente comando para instalar PHP y sus módulos necesarios:

- `sudo apt update`
- `sudo apt install php`

**Instala el módulo de Apache para PHP:** Instala el módulo de Apache que permite la integración con PHP:

- `sudo apt install libapache2-mod-php`

**Reinicia Apache:** Después de instalar el módulo de PHP, reinicia Apache para que los cambios surtan efecto:

- `sudo systemctl restart apache2`