



DESPLIEGUE DE APLICACIONES WEB

2º DAW -IES ALIXAR

Despliegue en Servidor Web y EC2 de AWS

Carmen ciscar arroyo
mcisarr228@g.educaand.es

Contenido

| | |
|--|----|
| Despliegue en Servidor Web y EC2 de AWS | 0 |
| Manual de Despliegue en Servidor Web con EC2 en AWS..... | 2 |
| Objetivo | 2 |
| Requisitos Previos | 2 |
| Conectarse a la Instancia EC2..... | 3 |
| Paso 1: Crear una Instancia EC2 | 3 |
| Paso 2: Conectarse a la Instancia EC2 | 8 |
| Paso 3: Instalar y Configurar la Pila LAMP..... | 10 |
| Paso 4: Habilitar Certificado SSL..... | 14 |
| Paso 5: Instalar WordPress | 16 |

Manual de Despliegue en Servidor Web con EC2 en AWS

Objetivo

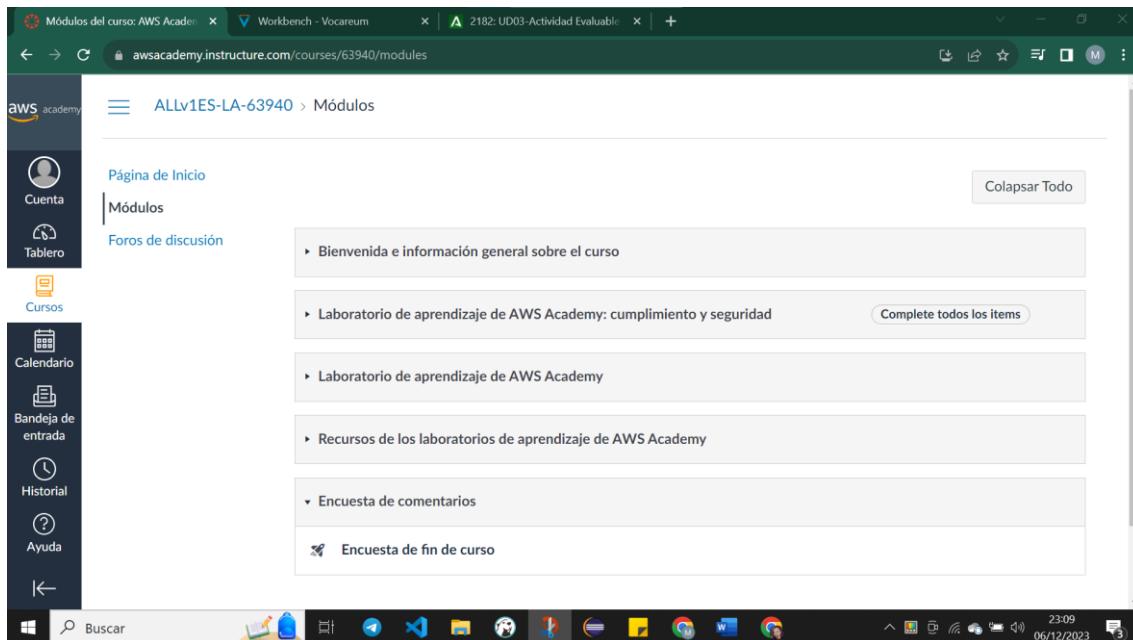
El objetivo de este despliegue es implementar una aplicación web en un servidor EC2 de AWS para que sea accesible a través de internet.

Requisitos Previos

Cuenta de AWS Educacional: Asegúrate de tener una cuenta de AWS. Si no tienes una, pídesela al profesor encargado.

Acceso a la Consola de AWS: Inicia sesión en la Consola de AWS con tus credenciales educativas.

Mediante módulos



The screenshot shows a web browser window with three tabs open: 'Módulos del curso: AWS Academy', 'Workbench - Vocareum', and '2182: UD03-Actividad Evaluable'. The main content area displays the 'Módulos' section for the course 'ALLv1ES-LA-63940'. On the left is a sidebar with navigation links: Cuenta, Tablero, Cursos, Calendario, Bandeja de entrada, Historial, Ayuda, and a back arrow. The 'Cursos' link is highlighted. The main content area shows a list of modules:

- Bienvenida e información general sobre el curso
- Laboratorio de aprendizaje de AWS Academy: cumplimiento y seguridad Complete todos los ítems
- Laboratorio de aprendizaje de AWS Academy
- Recursos de los laboratorios de aprendizaje de AWS Academy
- Encuesta de comentarios
- Encuesta de fin de curso

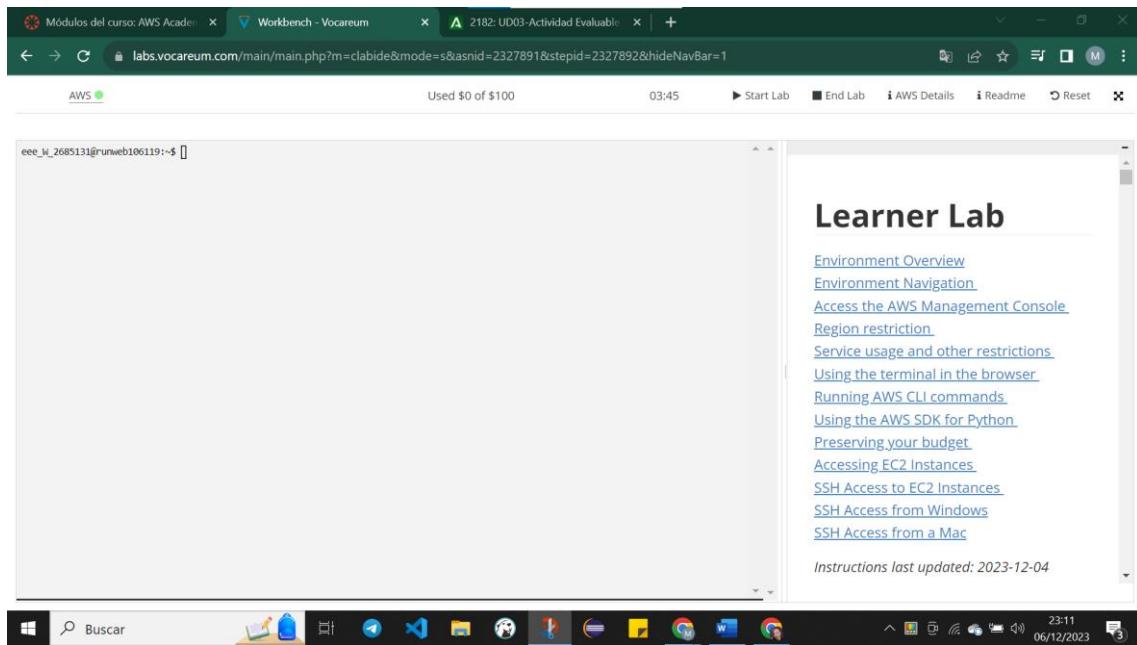
The status bar at the bottom shows the date as 06/12/2023 and the time as 23:09.

Conectarse a la Instancia EC2

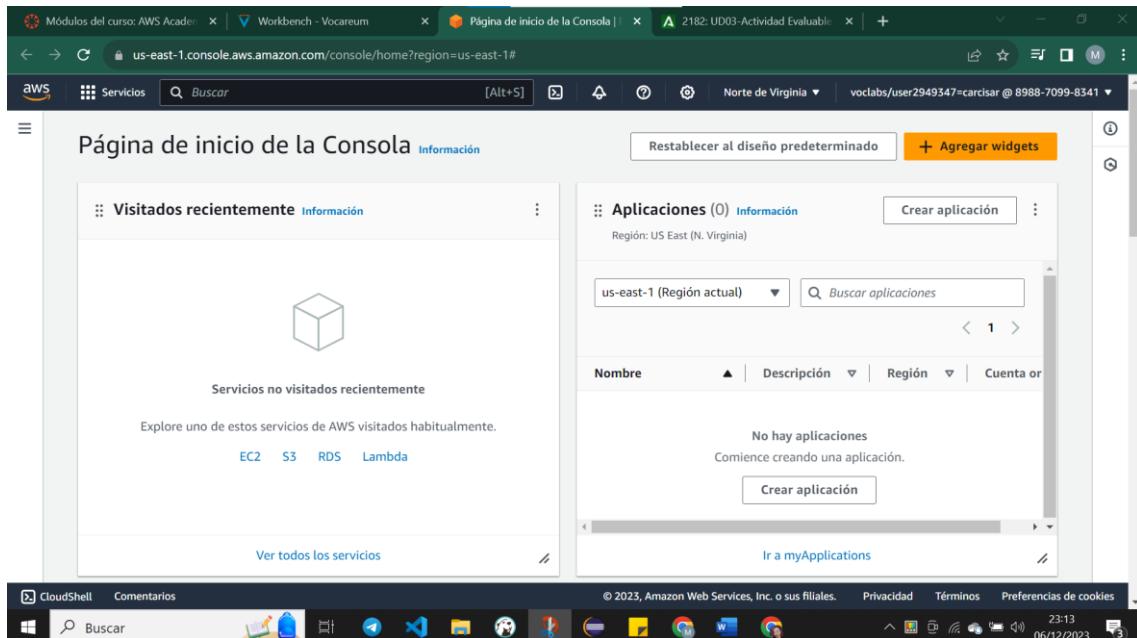
Paso 1: Crear una Instancia EC2

Ingresá a la consola de AWS con tu cuenta educativa.

Comenzamos con Start Lab

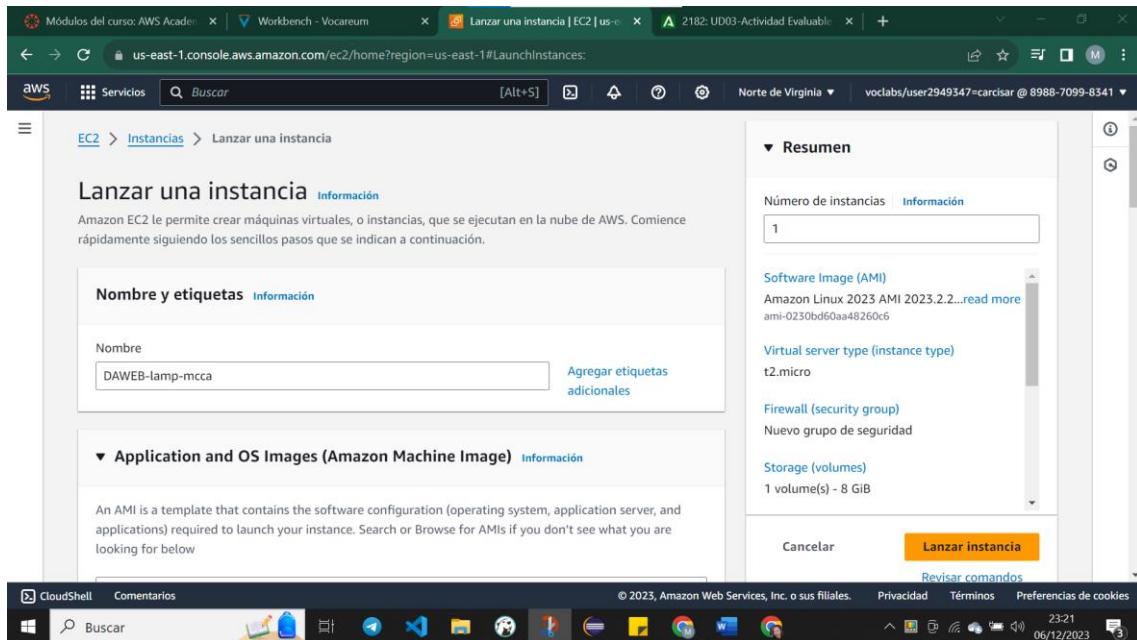


En la esquina superior izquierda, hacemos clic en AWS, y nos llevará a la página de inicio de la consola

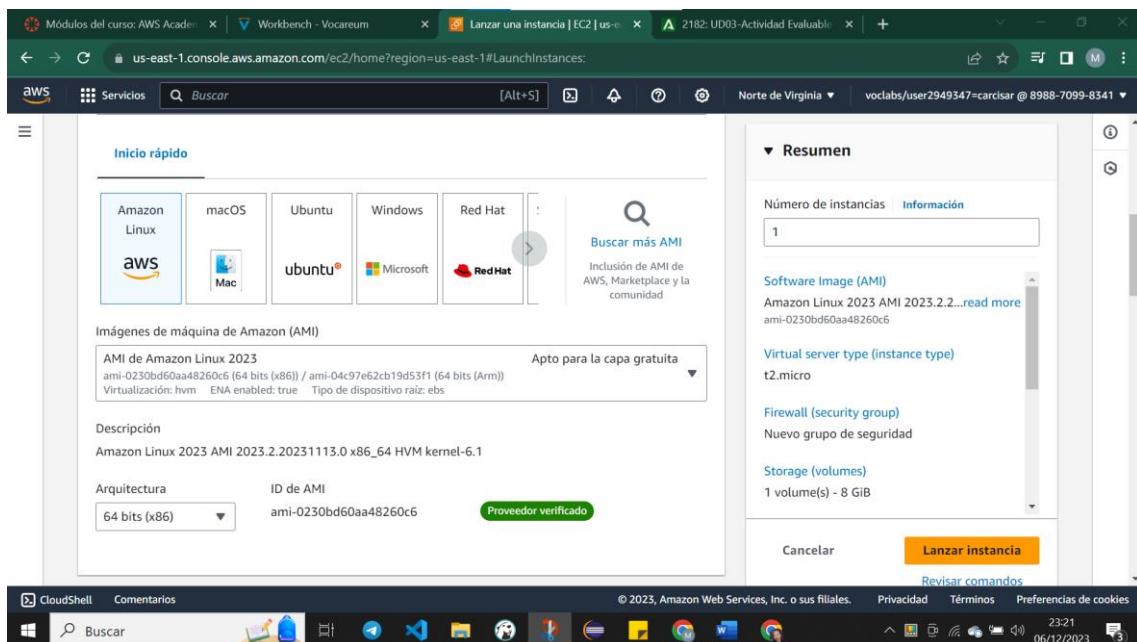


Ve a "EC2" en la sección de "Servicios".

Haz clic en "Launch Instance" para crear una nueva instancia.



Selecciona una imagen de Amazon Linux.



Elige una instancia t2.micro (puedes cambiarla según tus necesidades).

Tipo de instancia

t2.micro Apto para la capa gratuita
Familia: t2 1 vCPU 1 GiB Memoria Generación actual: true
Bajo demanda Windows base precios: 0.0162 USD por hora
Bajo demanda SUSE base precios: 0.0116 USD por hora
Bajo demanda RHEL base precios: 0.0716 USD por hora
Bajo demanda Linux base precios: 0.0116 USD por hora

Par de claves (inicio de sesión)

Puede utilizar un par de claves para conectarse de forma segura a la instancia. Asegúrese de que tiene acceso al par de claves seleccionado antes de lanzar la instancia.

Nombre del par de claves - obligatorio
Seleccionar

Resumen

Número de instancias: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.2.2...
ami-0230bd60aa48260c6

Virtual server type (instance type): t2.micro

Firewall (security group): Nuevo grupo de seguridad

Storage (volumes): 1 volume(s) - 8 GiB

En la configuración de instancias, configura la red y las opciones de seguridad según sea necesario.

Configuraciones de red

Red: Información
vpc-04c1d0529d291299e

Subred: Información
Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

Asignar automáticamente la IP pública: Información

Habilitar

Firewall (grupos de seguridad): Información
Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

Crear grupo de seguridad Seleccionar un grupo de seguridad existente

We'll create a new security group called 'launch-wizard-1' with the following rules:

Allow SSH traffic from
Helps you connect to your instance Cualquier lugar 0.0.0.0/0

Permitir el tráfico de HTTPS desde Internet
Para configurar un punto de enlace, por ejemplo, al crear un servidor web

Resumen

Número de instancias: 1

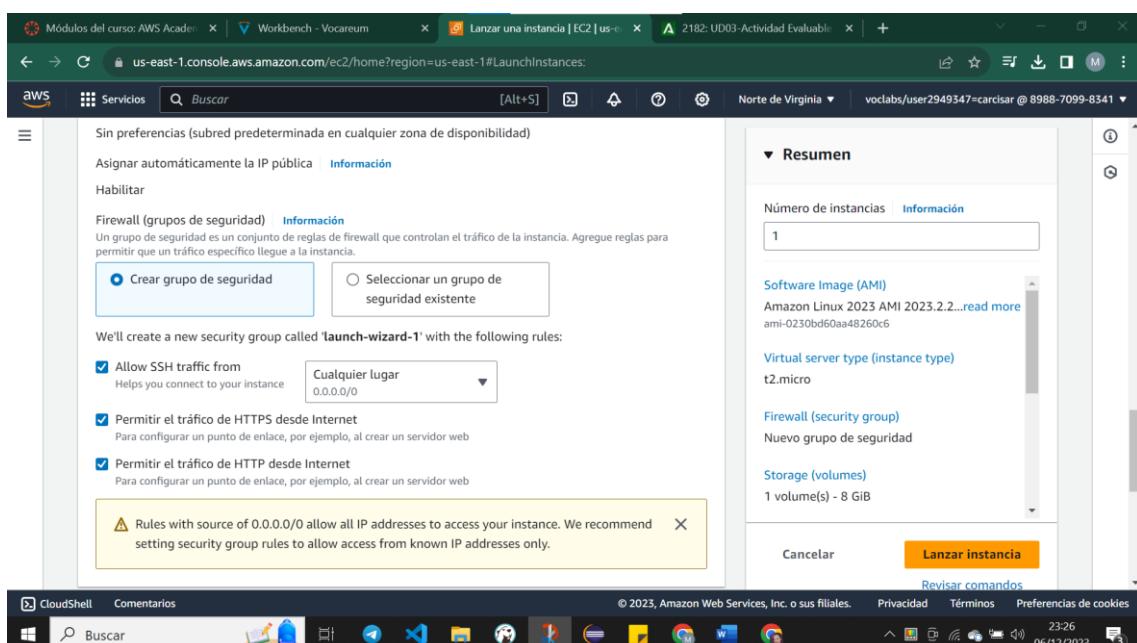
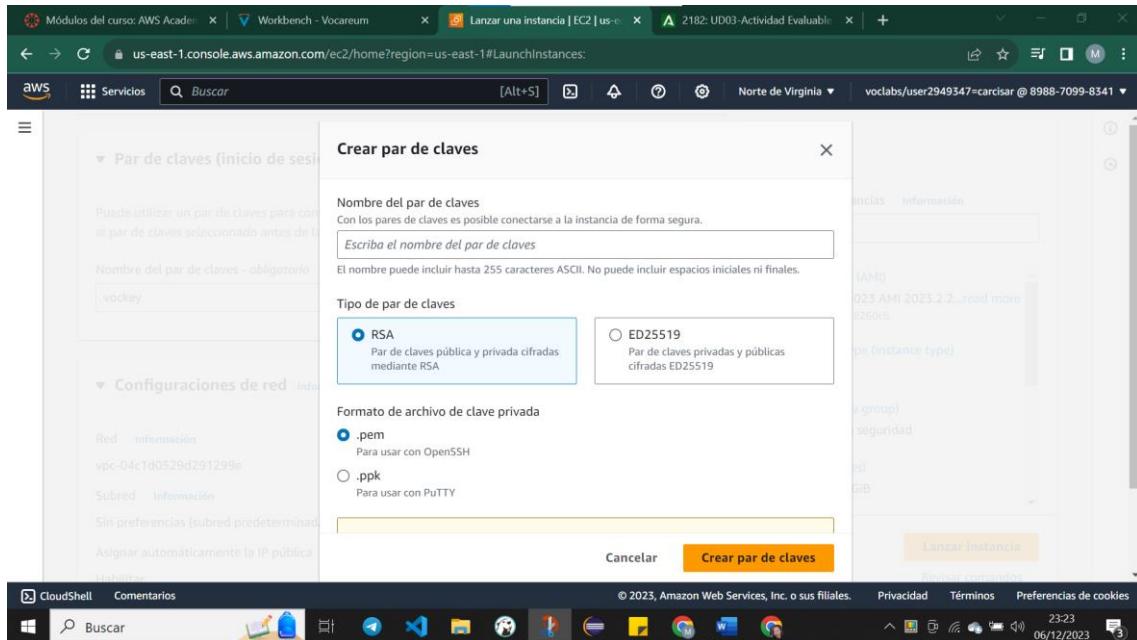
Software Image (AMI): Amazon Linux 2023 AMI 2023.2.2...
ami-0230bd60aa48260c6

Virtual server type (instance type): t2.micro

Firewall (security group): Nuevo grupo de seguridad

Storage (volumes): 1 volume(s) - 8 GiB

Revisa la configuración y haz clic en "Launch". Selecciona un par de claves existente o crea uno nuevo.



Lanzamos la instancia

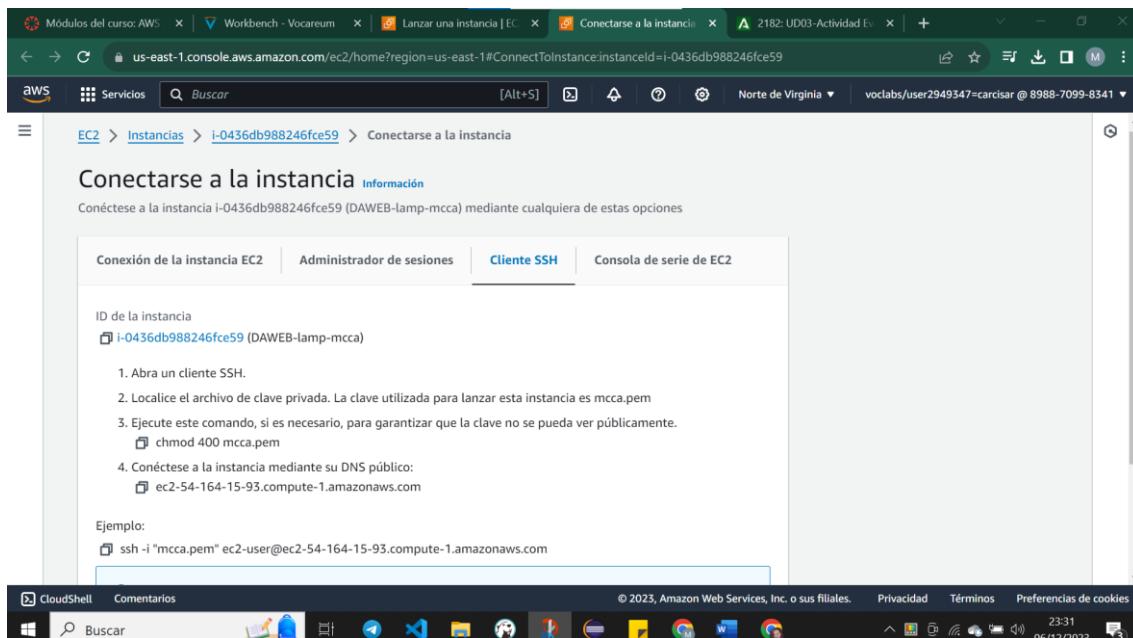
The screenshot shows the AWS Cloud Console interface. The top navigation bar includes tabs for 'Módulos del curso: AWS Academy', 'Workbench - Vocareum', 'Lanzar una instancia | EC2 | us-east-1', and '2182: UD03-Actividad Evaluable'. The main content area displays a green success message: 'Correcto' (Correct) and 'El lanzamiento de la instancia se inició correctamente (i-0436db988246fce59)' (The instance launch started successfully). Below this, there's a section titled 'Pasos siguientes' (Next steps) with three items: 'Crear alertas de uso del nivel gratuito y facturación' (Create usage alerts for free tier and billing), 'Conectarse a la instancia' (Connect to the instance), and 'Conectar una base de datos de RDS' (Connect to an RDS database). The bottom of the screen shows the Windows taskbar with various pinned icons.

Paso 2: Conectarse a la Instancia EC2

Una vez que la instancia está en ejecución, selecciona la instancia y haz clic en "Connect".

This screenshot is similar to the previous one but focuses on the 'Connect to instance' step. The 'Conectarse a la instancia' section is highlighted, showing the 'Conectar a la instancia' button. Below it, there are links for 'Más información' (More information) and 'Crear una nueva base de datos de RDS' (Create a new RDS database). At the bottom right of the page, there is a prominent orange button labeled 'Ver todas las instancias' (View all instances).

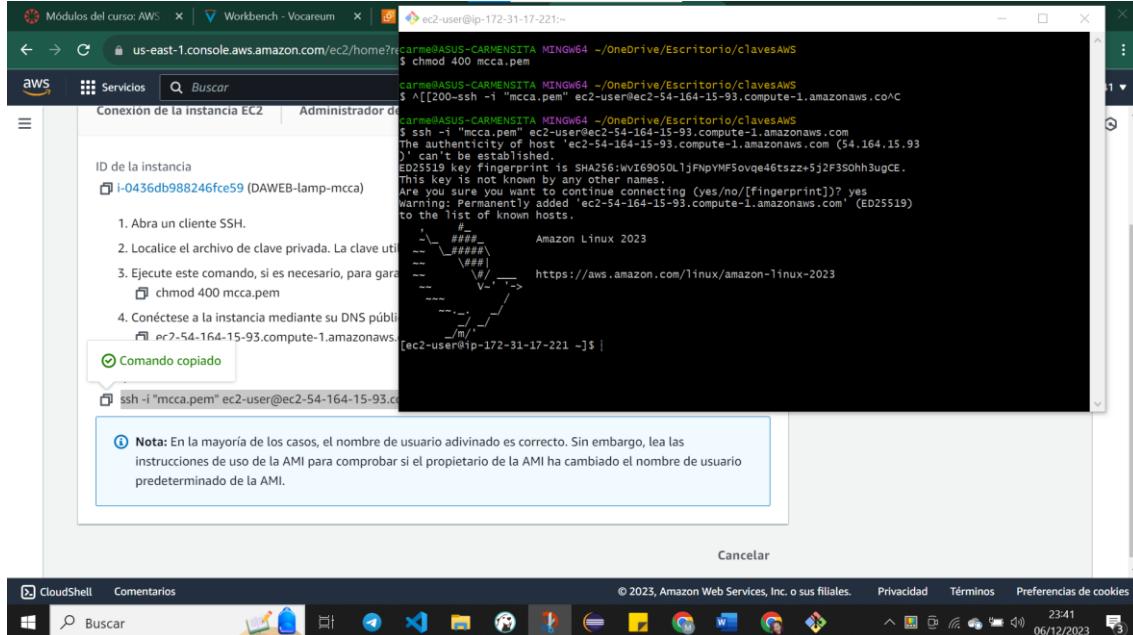
Sigue las instrucciones para conectarte a la instancia a través de SSH. Puedes usar una terminal de git bash si estás con Windows.



The screenshot shows the AWS CloudShell interface with the following details:

- Page Title:** Conectarse a la instancia (Information)
- Section:** Conexión de la instancia EC2
- Content:**
 - Abra un cliente SSH.
 - Localice el archivo de clave privada. La clave utilizada para lanzar esta instancia es mcca.pem
 - Ejecute este comando, si es necesario, para garantizar que la clave no se pueda ver públicamente:
chmod 400 mcca.pem
 - Conéctese a la instancia mediante su DNS público:
ec2-54-164-15-93.compute-1.amazonaws.com

Ejemplo:
ssh -i "mcca.pem" ec2-user@ec2-54-164-15-93.compute-1.amazonaws.com
- Bottom Status Bar:** © 2023, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies



The screenshot shows the AWS CloudShell terminal session with the following details:

- Terminal Output:**

```
carme@ASUS-CARMENSTA MINGW64 ~/OneDrive/Escritorio/clavesAWS
$ ssh -i "mcca.pem" ec2-user@ec2-54-164-15-93.compute-1.amazonaws.com
The authenticity of host 'ec2-54-164-15-93.compute-1.amazonaws.com (54.164.15.93)' can't be established.
ECDSA key fingerprint is SHA256:WjE905OLljFnPyMF5ovqe46tszz+5j2F35Ohh3ugCE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
warning: Permanently added 'ec2-54-164-15-93.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-172-31-17-221 ~]$
```
- Message:** Comando copiado
- Info Message:** Nota: En la mayoría de los casos, el nombre de usuario adivinado es correcto. Sin embargo, lea las instrucciones de uso de la AMI para comprobar si el propietario de la AMI ha cambiado el nombre de usuario predeterminado de la AMI.
- Bottom Status Bar:** © 2023, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

The screenshot shows the AWS Management Console with the EC2 Instances page open. The instance details for 'i-0436db988246fce59 (DAWEB-lamp-mcca)' are displayed. The instance is running and has a public IP of 54.164.15.93 and a private IP of 172.31.17.221. The AWS taskbar at the bottom shows various icons and the date/time.

Paso 3: Instalar y Configurar la Pila LAMP

En la instancia EC2, actualiza la lista de paquetes:

```
sudo dnf update -y
```

```
sudo dnf install -y httpd wget php-fpm php-mysql php-json php php-devel
```

```
sudo dnf install mariadb105-server
```

```
sudo dnf info package_name
```

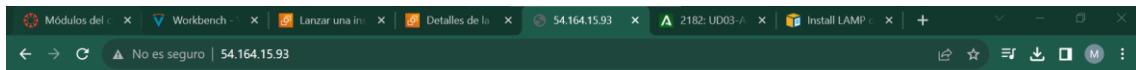
```
[ec2-user@ip-172-31-17-221:~]$ sudo dnf install -y httpd wget php-fpm php-mysql php-json php php-devel
Last metadata expiration check: 0:17:13 ago on wed Dec 6 22:09:16 2023.
Package wget-1.21.3-1.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
=====
Available Packages
Name        Arch      Version           Repository      Size
=====
Installing:
httpd       x86_64   2.4.58-1.amzn2023    amazonlinux   47 k
httpd-devel x86_64   8.2.9-1.amzn2023.0.3  amazonlinux   13 k
php8.2-devel x86_64   8.2.9-1.amzn2023.0.3  amazonlinux   208 k
php8.2-fpm  x86_64   8.2.9-1.amzn2023.0.3  amazonlinux   1.9 M
php8.2-mysqli x86_64   8.2.9-1.amzn2023.0.3  amazonlinux   150 k
Installing dependencies:
annobin-docs      noarch  10.93-1.amzn2023.0.1  amazonlinux   92 k
annobin-plugin-gcc x86_64  10.93-1.amzn2023.0.1  amazonlinux   887 k
apr-util          x86_64  1.7.7-2.amzn2023.0.2  amazonlinux   129 k
autoconf          x86_64  1.6.3-1.amzn2023.0.1  amazonlinux   98 k
automake          noarch  1.16.5-9.amzn2023.0.3  amazonlinux   666 k
cmake             x86_64  3.7.2-2.amzn2023.0.4  amazonlinux   677 k
cmake-filesystem  x86_64  1.1.2-2.amzn2023.0.2  amazonlinux   16 k
cpp               x86_64  1.28.2-3.amzn2023.0.6  amazonlinux   9.5 M
gcc               x86_64  8.0.4-5.amzn2023.0.2  amazonlinux   105 k
gcc-c++           x86_64  11.4.1-2.amzn2023.0.2  amazonlinux   32 M
generic-logos-https x86_64  18.0.0-12.amzn2023.0.3  amazonlinux   12 M
generic-logos     noarch  2.3.0-1.amzn2023.0.1  amazonlinux   4 K
glib              x86_64  2.64-52.amzn2023.0.7  amazonlinux   446 k
glib2              x86_64  2.2.7-7.amzn2023.0.3  amazonlinux   6.4 M
httpd-core        x86_64  2.4.58-1.amzn2023  amazonlinux   1.4 M
httpd-filesystem  noarch  2.4.58-1.amzn2023  amazonlinux   14 k
httpd-tools       x86_64  2.4.58-1.amzn2023  amazonlinux   81 k
kernel-headers    x86_64  6.1.0-85.amzn2023  amazonlinux   1.4 M
krb5-devel        x86_64  1.21-3.amzn2023.0.3  amazonlinux   55 k
krb5              x86_64  1.21-3.amzn2023.0.3  amazonlinux   136 k
libbrotli         x86_64  1.0.9-4.amzn2023.0.2  amazonlinux   315 k
libcom_err-devel  x86_64  1.46.5-2.amzn2023.0.2  amazonlinux   17 k
libkadm5          x86_64  1.21-3.amzn2023.0.3  amazonlinux   80 k
libmpc            x86_64  1.1.2-2.amzn2023.0.2  amazonlinux   62 k
libnslinux-devel  x86_64  1.1-1.amzn2023.0.1  amazonlinux   113 k
libsspol-devel   x86_64  3.4-3.amzn2023.0.3  amazonlinux   42 k
libssodium         x86_64  1.0.18-13.amzn2023.0.1  amazonlinux   166 k
libstdc++-devel   x86_64  11.4.1-2.amzn2023.0.2  amazonlinux   2.2 M
libtbo0           x86_64  2.4.7-1.amzn2023.0.3  amazonlinux   596 k
libtbo0-tl          x86_64  2.4.7-1.amzn2023.0.3  amazonlinux   38 k
libverto-devel    x86_64  0.1.1-1.amzn2023.0.2  amazonlinux   10 k
libxml2-devel     x86_64  4.4.33-1.amzn2023  amazonlinux   32 k
libxml2-devel     x86_64  2.10.4-1.amzn2023.0.6  amazonlinux   500 k

```

Comprueba la pila LAMP (Linux, Apache, MySQL, PHP):

```
sudo systemctl start httpd
```

```
sudo systemctl enable httpd
```



It works!

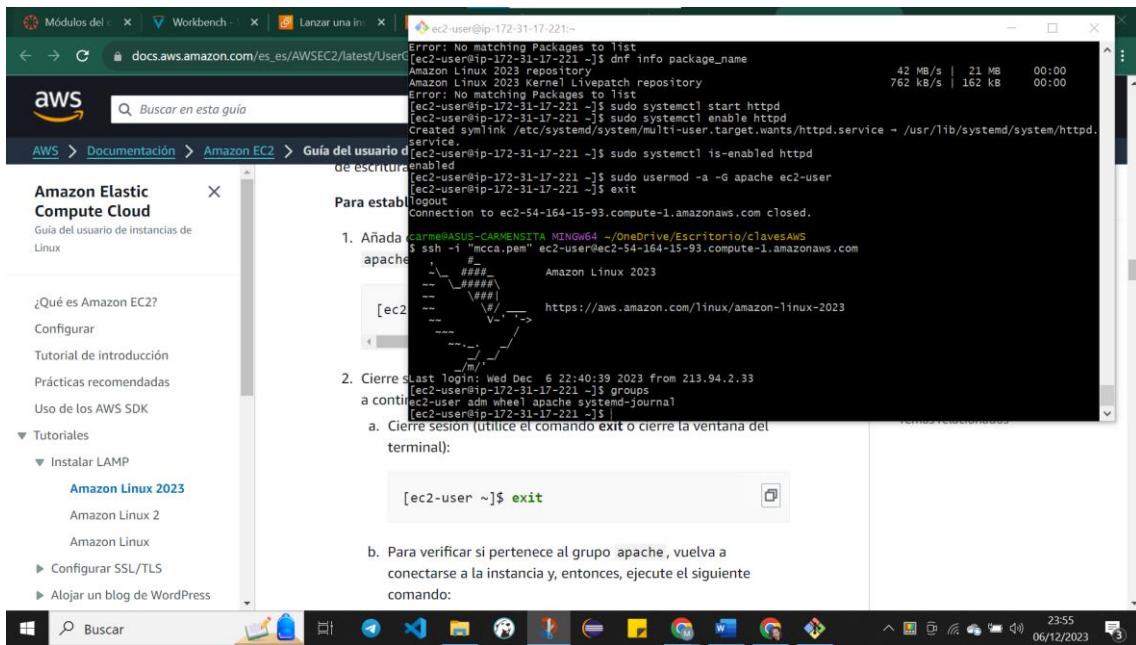


Configura los usuarios y grupos:

```
sudo usermod -a -G apache ec2-user
```

```
exit
```

```
groups
```

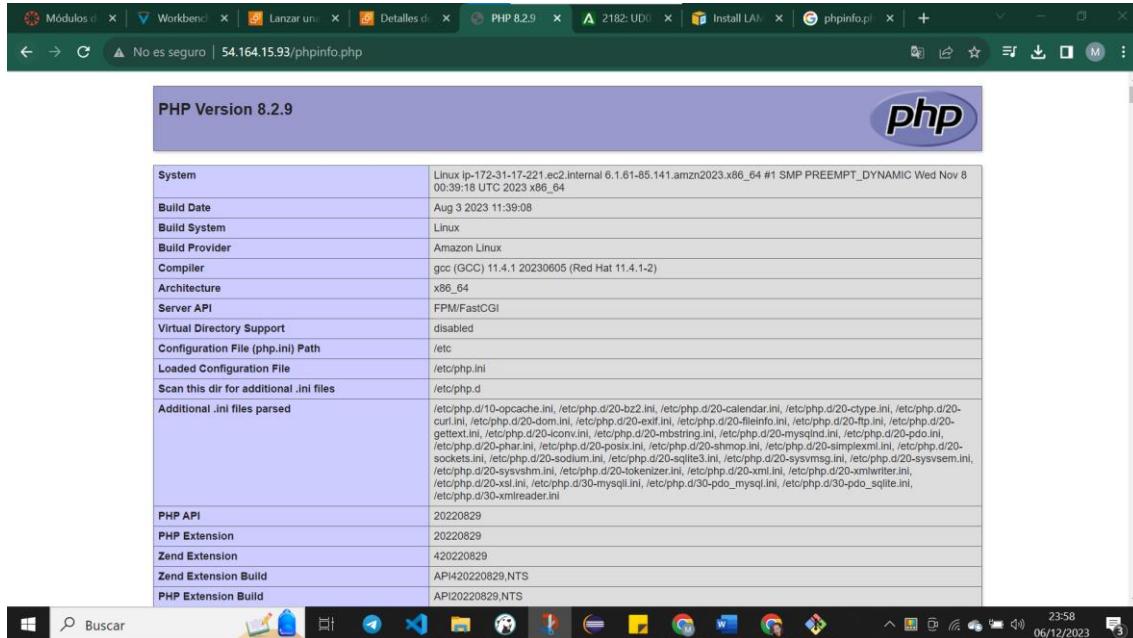


```
sudo chown -R ec2-user:apache /var/www
sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
find /var/www -type f -exec sudo chmod 0664 {} \;
```

Inicia y habilita los servicios:

Probamos php

```
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```



Sigue los comandos para configurar la seguridad de MySQL.

```
sudo systemctl start mariadb
```

```
sudo mysql_secure_installation
```

```
Administrator:~ ip-172-31-17-221 ~
Password updated successfully!
Reloading privilege tables...
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into the database without having to guess a password. This
is intended only for testing, and to help get MariaDB up and
running so it's smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [y/n] y
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
[ec2-user@ip-172-31-17-221 ~]$ sudo systemctl stop mariadb
[ec2-user@ip-172-31-17-221 ~]$ sudo systemctl enable mariadb
Created symlink /etc/systemd/system/mysql.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /usr/lib/systemd/system/mariadb.service.
[ec2-user@ip-172-31-17-221 ~]$
```

```
sudo systemctl stop mariadb  
sudo systemctl enable mariadb
```

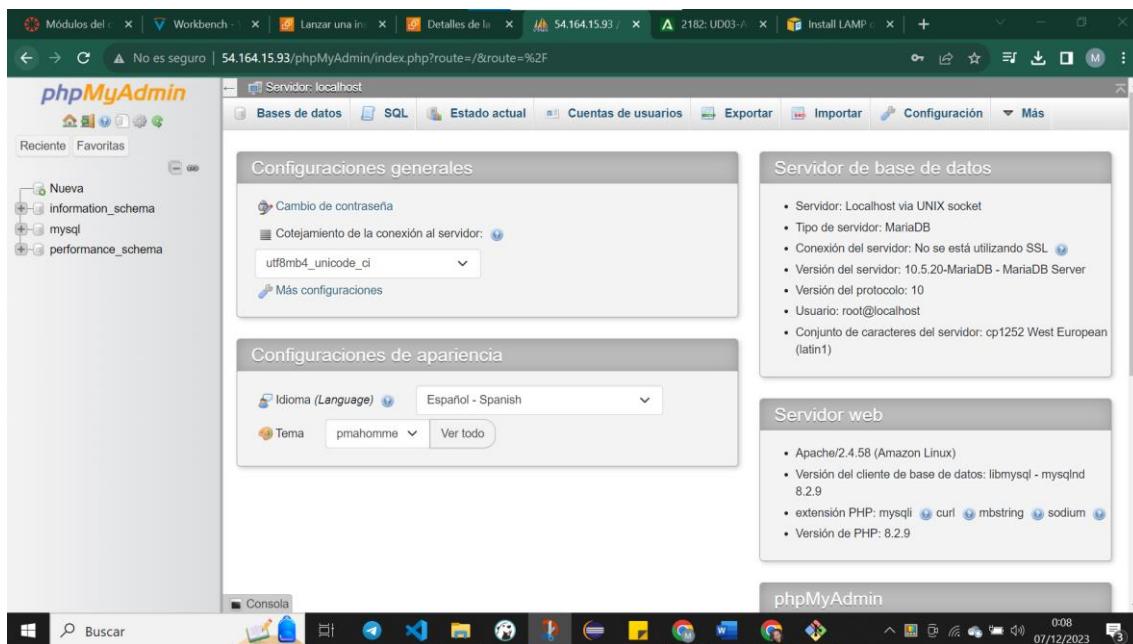
Instalar phpMyAdmin

```
sudo dnf install php-mbstring php-xml -y  
sudo systemctl restart httpd  
cd /var/www/html  
wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz  
mkdir phpMyAdmin && tar -xvf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1  
rm phpMyAdmin-latest-all-languages.tar.gz
```

Importante que la base de datos este habilitada, si no podrás entrar.

```
sudo systemctl start mariadb
```

<http://my.public.dns.amazonaws.com/phpMyAdmin>



Paso 4: Habilitar Certificado SSL

```
sudo dnf install openssl mod_ssl
```

```
ec2-user@ip-172-31-17-221:/var/www/html Modified
GNU nano 5.8 /etc/httpd/conf.d/ssl.conf

User agents such as web browsers are not configured for the user's own preference of either security or performance, therefore this must be the prerogative of the web server administrator who manages # cpu load versus confidentiality, so enforce the server's cipher order.
SSLHonorCipherOrder on

# SSL Cipher Suite:
# List the cipher suites that the client is permitted to negotiate.
# See the mod_ssl documentation for a complete list.
# The OpenSSL system profile is configured by default. See
# update-crypto-policies(8) for more details.
SSLCipherSuite PROFILE=SYSTEM
SSLProxyCipherSuite PROFILE=SYSTEM

Point SSLCertificateFile at a PEM encoded certificate. If the certificate is encrypted, then you will be prompted for a pass phrase. Note that restarting httpd will prompt again. Keep in mind that if you have both an RSA and a DSA certificate, you can configure both in parallel (to also allow the use of DSA ciphers, etc.)
Some ECC cipher suites (http://www.ietf.org/rfc/rfc4492.txt) require an ECC certificate which can also be configured in parallel
SSLCertificateFile /etc/pki/tls/certs/apache-selfsigned.crt

# Server Private Key:
# If the key is not combined with the certificate, use this directive to point to the key file. Keep in mind that if you have both an RSA and a DSA private key, you can configure both in parallel (to also allow the use of DSA ciphers, etc.)
# ECC keys, when in use, can also be configured in parallel
SSLCertificateKeyFile /etc/pki/tls/private/apache-selfsigned.key

Server Certificate Chain:
Point SSLCertificateChainFile at a file containing the concatenated list of PEM encoded CA certificates which form the certificate chain for the server certificate. Alternatively the referenced file can be the same as SSLCertificateFile when the CA certificates are directly appended to the server certificate for convenience.
SSLCertificateChainFile /etc/pki/tls/certs/server-chain.crt

Certificate Authority (CA):
# Set the CA certificate verification path where to find CA
```



La conexión no es privada

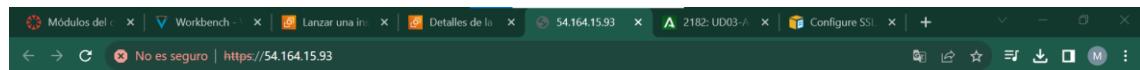
Es posible que los atacantes estén intentando robar tu información de **54.164.15.93** (por ejemplo, contraseñas, mensajes o tarjetas de crédito). [Más información](#)

NET::ERR_CERT_AUTHORITY_INVALID

💡 Para disfrutar del máximo nivel de seguridad en Chrome, [activa la protección mejorada.](#)

[Configuración avanzada](#)

[Volver para estar a salvo](#)



It works!



Paso 5: Instalar WordPress

Descarga e instala WordPress:

Descargue e instale estos paquetes con el siguiente comando.

```
dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json php php-devel -y
```

Descargue el último paquete de instalación de WordPress con el comando wget. El comando siguiente debería descargar siempre la última versión.

```
wget https://wordpress.org/latest.tar.gz
```

Descomprima y desarchive el paquete de instalación. La carpeta de instalación se descomprime en una carpeta llamada wordpress.

```
tar -xzf latest.tar.gz
```

```
If you enter '.', the field will be left blank.
Country Name (2 letter code) [XX]:ES
State or Province Name (full name) []:SEVILLA
Locality Name (eg, city) [Default City]:SEVILLA
Organization Name (eg, company) [Default Company Ltd]:IES ALIXAR
Organizational Unit Name (eg, section) []
Common Name (eg, your name or your server's hostname) []:ec2-54-164-15-93.compute-1.amazonaws.com
Email Address [email@mcisarr2280.educaand.es]
DNS Name(s) [www]:www
HTTP Port [80]:80
SSLCertFile [etc/httpd/conf.d/ssl.conf]
[ec2-user@ip-172-31-17-221 ~]$ sudo systemctl restart httpd
[ec2-user@ip-172-31-17-221 ~]$ sudo dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json php php-devel -y
Error: This command must be run with superuser privileges (under the root user on most systems).
[ec2-user@ip-172-31-17-221 ~]$ sudo dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json php php-devel -y
Last metadata expiration check: 0:51:57 ago on Wed Dec 6 22:29:16 2023.
Package wget-2.38-1.amzn2023.0.3.x86_64 is already installed.
Package httpd-2.4.58-1.amzn2023.x86_64 is already installed.
Package php8.2-fpm-8.2.9-1.amzn2023.0.3.x86_64 is already installed.
Package mariadb105-server-3:10.5.20-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-mysqlnd-8.2.9-1.amzn2023.0.3.x86_64 is already installed.
Package mariadb105-server-3:10.5.20-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-fpm-8.2.9-1.amzn2023.0.3.x86_64 is already installed.
Package php8.2-dev-8.2.9-1.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-17-221 ~]$ wget https://wordpress.org/latest.tar.gz
--2023-12-06 23:22:05-- https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24479697 (23M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz          [=====] 23.34M 31.2MB/s   in 0.7s
2023-12-06 23:22:05 (31.2 MB/s) - 'latest.tar.gz' saved [24479697/24479697]

[ec2-user@ip-172-31-17-221 ~]$ Descomprima y desarchive el paquete de instalación. La carpeta de instalación se descomprime en una carpeta llamada wordpress.

[ec2-user ~]$ tar -xzf latest.tar.gz
-bash: Descomprima: command not found
-bash: [ec2-user: command not found
[ec2-user@ip-172-31-17-221 ~]$ tar -xzf latest.tar.gz
[ec2-user@ip-172-31-17-221 ~]$ ls
[latest.tar.gz] phpMyAdmin phpInfo.php wordpress
[ec2-user@ip-172-31-17-221 ~]$ |
```

Inicie la base de datos y el servidor web.

```
sudo systemctl start mariadb httpd
```

```
mysql -u root -p
```

Cree un usuario y una contraseña para la base de datos MySQL.

```
CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'your_strong_password';
```

Cree la base de datos.

```
CREATE DATABASE `wordpress-db`;
```

Conceda privilegios completos para la base de datos al usuario de WordPress que ha creado antes.

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
```

```
FLUSH PRIVILEGES;
```

exit

```
Nothing to do.
Complete!
[ec2-user@ip-172-31-17-221 ~]$ wget https://wordpress.org/latest.tar.gz
--2023-12-06 23:22:04-- https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24479697 (23M) [application/octet-stream]
Saving to: "latest.tar.gz"

latest.tar.gz          100%[=====] 23.34M 31.2MB/s   in 0.7s

2023-12-06 23:22:05 (31.2 MB/s) - 'latest.tar.gz' saved [24479697/24479697]

[ec2-user@ip-172-31-17-221 ~]$ Descomprima y desarchivé el paquete de instalación. La carpeta de instalación se descomprime en una carpeta llamada wordpress.

[ec2-user ~]$ tar -xvf latest.tar.gz
tar: --decompress: command not found
tar: --decompress: command not found
[ec2-user@ip-172-31-17-221 ~]$ tar -xvf latest.tar.gz
[ec2-user@ip-172-31-17-221 ~]$ ls
latest.tar.gz  phpMyAdmin  phinfo.php  wordpress
[ec2-user@ip-172-31-17-221 ~]$ sudo systemctl start mariadb httpd
[ec2-user@ip-172-31-17-221 ~]$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 17
Server version: 10.5.20-MariaDB MariaDB Server

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)]> CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY '2203';
Query OK, 0 rows affected (0.006 sec)

MariaDB [(none)]> CREATE DATABASE `wordpress-db`;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> exit
Bye
[ec2-user@ip-172-31-17-221 ~]$ |
```

Para crear y modificar el archivo wp-config.php:

```
cp wordpress/wp-config-sample.php wordpress/wp-config.php
```

generando las claves de wordpress, cambiamos todos los datos requeridos.

```
rc2-user@ip-172-31-17-221:/var/www/html
GNU nano 5.8
wordpress/wp-config.php
Modified

/* @package WordPress
*/
/** Database settings - You can get this info from your web host ** */
/** The name of the database for WordPress */
define( 'DB_NAME', 'Wordpress-db' );

/** Database username */
define( 'DB_USER', 'wordpress-user' );

/** Database password */
define( 'DB_PASSWORD', '2203' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

/*#+#
 * Authentication unique keys and salts.
 *
 * Change these to different unique phrases! You can generate these using
 * the (@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service).
 *
 * You can change these at any point in time to invalidate all existing cookies.
 * This will force all users to have to log in again.
 *
 * Since 2.6.0
 */
define('AUTH_KEY', 'gfE@08nKvxi|oJ6pm|+C8yjFJyv8l-HGToage|1n-9-wf#FT7y0s7Vuuc(C`5p0');
define('SECURE_AUTH_KEY', 'GpP^+WzDgXm9sVg+9wvBxRz-54dRcTj720MhIu-1kqLjZu-');
define('LOGGED_IN_KEY', 'YX04SM5(R,=)j-2AU1B1dWm2Z{=18KR1;XkWmM6b1;PTC05-NEbdff');
define('NONCE_KEY', 'h_P`eo-oh{jid-f5$+}-;n7BpgjUW+7FBTp#7d!ao_#k<8n5n?eq/RvR3');
define('AUTH_SALT', 'l{eqwx+p!wdEVa)3dt%6nOV21+RU_jrz?kap_q#8R6_-Pa>:5hqrh+czP%PR');
define('SECURE_AUTH_SALT', 'AVsVs+aEKvevFW26-RWbQdR8,%_HSCN.sh.wLO-~$X8pE}f0|-j>2(f');
define('LOGGED_IN_SALT', '42#neJW-ATPTOC $t|R1ow[?=sh|bzoU!nzsooyIIfvus97LT_=vh9jkC8U');
define('NONCE_SALT', 'ar75w-zgev)5.r q-0D p3.3U9GvGOB>80sJ5mA#Gbt-Lh5:!2H8mKF]Fw12+');

/*#+#
 * Help Write Out Read File Where Is Replace Cut Paste Execute Justify Location Go To Line Undo Redo Set Mark To Bracket Where Was Previous Next
 * Exit Buscar
```

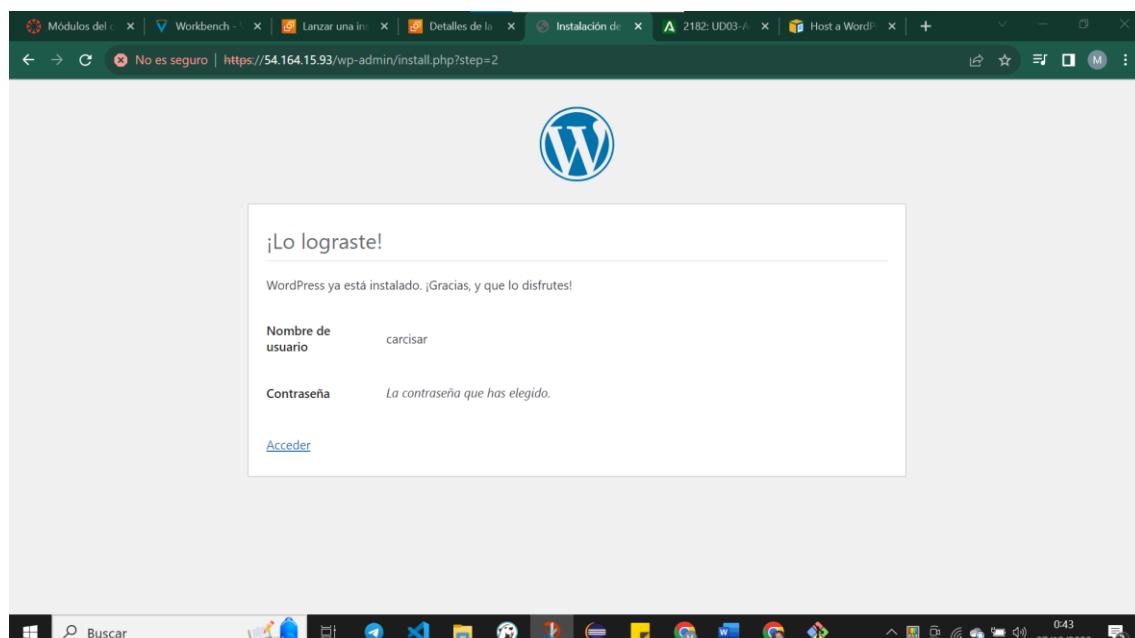
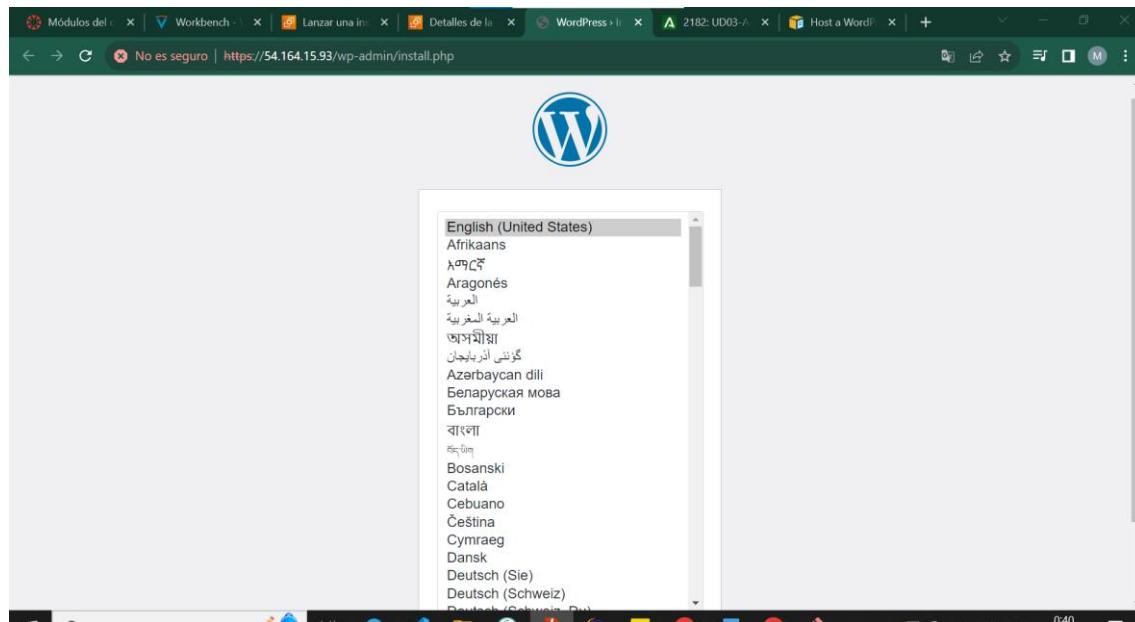
Verifique que el servidor de base de datos se está ejecutando.

```
sudo systemctl status mariadb
```

Verifique que el servidor web Apache (httpd) se está ejecutando.

```
sudo systemctl status httpd
```

Una vez verificados, si vamos a la dirección de aws, debe de aparecer el instalador de wordpress.



¡Listo!

Ahora puedes acceder a tu sitio WordPress mediante el dominio o la IP de tu instancia EC2.