

# Instalación de WordPress en AWS con SSH

David Arbelaez Mutis

25/11/2025

2º ASIR - Arquitectura en la nube  
Prometeo-Caja Mágica

<b>Instalación de WordPress en AWS con SSH</b>	<b>2</b>
PARTE 0: PREPARACIÓN DEL ENTORNO LOCAL (WSL/Linux)	2
PARTE 1: CONFIGURACIÓN EN AWS	3
PARTE 2 Y 3: CONEXIÓN E INSTALACIÓN BASE	4
PARTE 4 Y 5: AUTOMATIZACIÓN Y MIGRACIÓN	5
PARTE 6: VERIFICACIÓN	7
PARTE 7: ACCESO EXTERNO CON NGROK	7
NOTAS FINALES	9

## PARTE 0: PREPARACIÓN DEL ENTORNO LOCAL (WSL/Linux)

### 1. Generación de Claves SSH (Ed25519)

Se crea el directorio .ssh seguro y se genera un par de claves SSH moderno (algoritmo **Ed25519**) en la máquina local para la autenticación segura con AWS. Se ajustan los permisos de la clave privada (chmod 400) para evitar errores de seguridad.

#### Comandos Clave:

```
ssh-keygen -t ed25519 -f ~/.ssh/wordpress-key -C "david@aws"  
ls -la ~/.ssh/wordpress-key*
```

```
root@A6Alumno10:/home/lumno10/.ssh# ssh-keygen -t ed25519 -f ~/.ssh/wordpress-key -C "David@aws"  
Generating public/private ed25519 key pair.  
/root/.ssh/wordpress-key already exists.  
Overwrite (y/n)? y  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /root/.ssh/wordpress-key  
Your public key has been saved in /root/.ssh/wordpress-key.pub  
The key fingerprint is:  
SHA256:b8/NQZwUh6xtkXz5sAZ5sNcOusLTN5gZBLZryf0Dabg David@aws  
The key's randomart image is:  
+--[ED25519 256]--  
| o ..oo |  
| . oo.0= |  
| . o+=o |  
| . *.*=+ |  
S * Bo+ |  
+ + X |  
E * * |  
. = + + |  
o o |  
+---[SHA256]---+  
root@A6Alumno10:/home/lumno10/.ssh#
```

<Imagen 1. Terminal local mostrando la generación de las claves y el listado de archivos (privada y pública) en el directorio .ssh.>

## PARTE 1: CONFIGURACIÓN EN AWS

### 2. Creación del Key Pair y Security Group

En la consola de AWS, se crea un **Key Pair** (wordpress-key-aws) y se descarga el archivo .pem. Luego, se configura un **Security Group** (wordpress-aws-sg) que actúa como firewall virtual, permitiendo el tráfico necesario:

- **SSH (22):** Para administración remota.
- **HTTP (80) y HTTPS (443):** Para el acceso web y el funcionamiento de Ngrok.

Inbound rules (3)						
<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range
<input type="checkbox"/>	-	sgr-0f0d3f28b97422b54	IPv4	HTTP	TCP	80
<input type="checkbox"/>	-	sgr-0056f3373ded0f8d2	IPv4	SSH	TCP	22
<input type="checkbox"/>	-	sgr-041c7b7076c48f433	IPv4	HTTPS	TCP	443

<Imagen 2. Captura de AWS Console mostrando las reglas de entrada (Inbound rules) del Security Group con los puertos 22, 80 y 443 abiertos.>

### 3. Despliegue de la Instancia EC2

Se lanza una instancia EC2 con **Ubuntu 24.04 LTS** y tipo **t3.micro** (Free Tier). Se asocia el par de claves y el grupo de seguridad creados anteriormente. Una vez que la instancia está en estado running, se obtiene la **IP Pública**.

▼ Instance details <a href="#">Info</a>	
<b>AMI ID</b>	<a href="#">AMI details</a> ami-0ecb62995f68bb549
<b>Platform details</b>	<a href="#">Platform details</a> Linux/UNIX
<b>Allowed image</b>	-
<b>Stop protection</b>	Disabled
<b>AMI location</b>	<a href="#">AMI location</a> amazon/ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-20251022
<b>Instance auto-recovery</b>	Default
<b>Stop-hibernate behavior</b>	Disabled
<b>Key pair assigned at launch</b>	<a href="#">Key pair assigned at launch</a> wordpress-key-aws
<b>Credit specification</b>	unlimited
<b>Monitoring</b>	disabled
<b>AMI name</b>	<a href="#">AMI name</a> ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-20251022
<b>Termination protection</b>	Disabled
<b>Launch time</b>	<a href="#">Launch time</a> Mon Nov 24 2025 11:13:34 GMT+0100 (hora estándar de Europa central) (1 minute)
<b>Instance reboot migration</b>	Default (On)
<b>Lifecycle</b>	normal
<b>AMI Launch index</b>	0
<b>State transition reason</b>	-
<b>Kernel ID</b>	-

<Imagen 3. Panel de instancias EC2 mostrando la instancia 'wordpress-server' en ejecución y su dirección IP pública visible.>

## PARTE 2 Y 3: CONEXIÓN E INSTALACIÓN BASE

### 4. Conexión SSH y Despliegue LAMP

- Se establece la conexión SSH desde la máquina local hacia la instancia AWS utilizando la clave .pem. Una vez dentro, se actualiza el sistema y se instala el stack **LAMP** (Linux, Apache, MySQL, PHP) y las librerías necesarias para WordPress.

**Comando de Conexión:**

```
ssh -i ~/.ssh/wordpress-key-aws.pem ubuntu@35.171.244.54
```

```

root@A6Alumno10:/home/lumno10/.ssh# ssh -i ~/.ssh/wordpress-key-aws.pem ubuntu@35.171.244.54
The authenticity of host '35.171.244.54 (35.171.244.54)' can't be established.
ED25519 key fingerprint is SHA256:funYDbJLpUZcfJQGq7/feXBtIcm+f0TsKaeaVsaOr8E.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '35.171.244.54' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1015-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon Nov 24 10:24:16 UTC 2025

System load: 0.0          Temperature:      -273.1 C
Usage of /:   25.8% of 6.71GB Processes:        111
Memory usage: 22%          Users logged in:   0
Swap usage:   0%           IPv4 address for ens5: 172.31.67.108

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

```

<Imagen 4. Terminal mostrando la conexión SSH exitosa al servidor AWS (prompt ubuntu@35.171.244.54) y el estado de los servicios Apache/MySQL.>

## PARTE 4 Y 5: AUTOMATIZACIÓN Y MIGRACIÓN

### 5. Creación y Transferencia del Script de Automatización

Se crea localmente el script install-wordpress.sh. Este script automatiza la configuración segura de MySQL, la creación de la BBDD y usuario, la descarga de WordPress, la configuración de wp-config.php y los permisos de Apache. Luego, se transfiere el script al servidor AWS usando SCP.

#### Comando SCP:

```
scp -i ~/.ssh/wordpress-key-aws.pem install-wordpress.sh ubuntu@35.171.244.54:~/
```

```

root@A6Alumno10:~/.ssh# touch install-wordpress.sh
root@A6Alumno10:~/.ssh# nano install-wordpress.sh
root@A6Alumno10:~/.ssh# scp -i ~/.ssh/wordpress-key-aws.pem install-wordpress.sh ubuntu@35.171.244.54:~/install-wordpress.sh
                                                               100% 2234     19.7KB/s   00:00
root@A6Alumno10:~/.ssh#

```

<Imagen 5. Captura mostrando el comando SCP de transferencia y el fichero install-wordpress.sh listado en el servidor AWS.>

## 6. Ejecución del Script y Credenciales

Se ejecutan los permisos de ejecución y se lanza el script en el servidor. Al finalizar, el script genera un archivo `wordpress-credentials.txt` con las contraseñas generadas aleatoriamente y los datos de acceso.

**Comando:** `./install-wordpress.sh`

```
ubuntu@ip-172-31-67-108:~$ chmod +x ~/install-wordpress.sh
ubuntu@ip-172-31-67-108:~$ ./install-wordpress.sh
*** Iniciando instalación automatizada de WordPress ***
Configurando MySQL...
Creando base de datos y usuario...
mysql: [Warning] Using a password on the command line interface can be insecure.
mysql: [Warning] Using a password on the command line interface can be insecure.
mysql: [Warning] Using a password on the command line interface can be insecure.
mysql: [Warning] Using a password on the command line interface can be insecure.
Descargando WordPress...
Copiando archivos...
Configurando wp-config.php...
Configurando permisos...
Habilitando mod_rewrite...
Enabling module rewrite.
To activate the new configuration, you need to run:
    systemctl restart apache2
*** Instalación completada ***
Credenciales guardadas en ~/wordpress-credentials.txt
```

Activar Windows  
Ve a Configuración p

<Imagen 6. Salida final del script en la terminal mostrando "Instalación completada" y el contenido del archivo de credenciales.>

```
ubuntu@ip-172-31-67-108:~$ cat ~/wordpress-credentials.txt
*** CREDENCIALES DE WORDPRESS ***
Base de datos: wordpress
Usuario BD: wpuser
Contraseña BD: +r4Uxt93EnpvSkfh
Usuario root MySQL: root
Contraseña root MySQL: Y7ETQuBxurAPggc1
```

<Imagen 7. Acceso al contenido del archivo de credenciales.>

## PARTE 6: VERIFICACIÓN

### 7. Acceso Web y Finalización de Instalación

Se accede a través del navegador a la **IP Pública** de AWS (`http://54.x.x.x`). Se muestra el asistente de instalación de WordPress, confirmando que Apache y PHP están funcionando correctamente.



<Imagen 7. Navegador accediendo a la IP pública de AWS mostrando la pantalla de bienvenida o el panel de administración de WordPress.>

## PARTE 7: ACCESO EXTERNO CON NGROK

### 8. Instalación y Ejecución de Ngrok en AWS

Se descarga e instala **ngrok** dentro de la instancia EC2. Se autentica con el token y se inicia un túnel HTTP en el puerto 80. Esto genera una URL pública segura (<https://...ngrok-free.app>) para acceder al servidor sin depender de la IP de AWS o dominios.

**Comando:** ngrok http 80

```
ubuntu@ip-172-31-67-108: ~ + v
ngrok (Ctrl+C to quit)
容忍 your secrets in vaults and (re)use them to transform traffic: https://ngrok.com/r/secrets

Session Status      online
Account            David Arbelaez (Plan: Free)
Version            3.33.0
Region             United States (us)
Web Interface     http://127.0.0.1:4040
Forwarding         https://triunitarian-multiflorous-paulette.ngrok-free.dev -> http://localhost:80

Connections        ttl     opn      rt1      rt5      p50      p90
                    0       0       0.00    0.00    0.00    0.00
```

<Imagen 8. Terminal de AWS con ngrok ejecutándose, mostrando la URL pública generada (<https://triunitarian-multiflorous-paulette.ngrok-free.app>).>

### 9. Actualización de Base de Datos (URLs de WordPress)

Para que WordPress funcione correctamente con la URL de ngrok, es necesario actualizar las opciones siteurl y home en la base de datos MySQL, reemplazando localhost por el dominio

de ngrok.

### Comandos SQL:

SQL

```
UPDATE wp_options SET option_value='https://...ngrok-free.app' WHERE option_name='siteurl';
UPDATE wp_options SET option_value='https://...ngrok-free.app' WHERE option_name='home';
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 57
Server version: 8.0.44-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> UPDATE wp_options SET option_value='https://triunitarian-multiflorous-paulette.ngrok-free.app'
WHERE option_name='siteurl';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> UPDATE wp_options SET option_value='https://triunitarian-multiflorous-paulette.ngrok-free.dev.ngrok-free.app' WHERE option_name='home';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

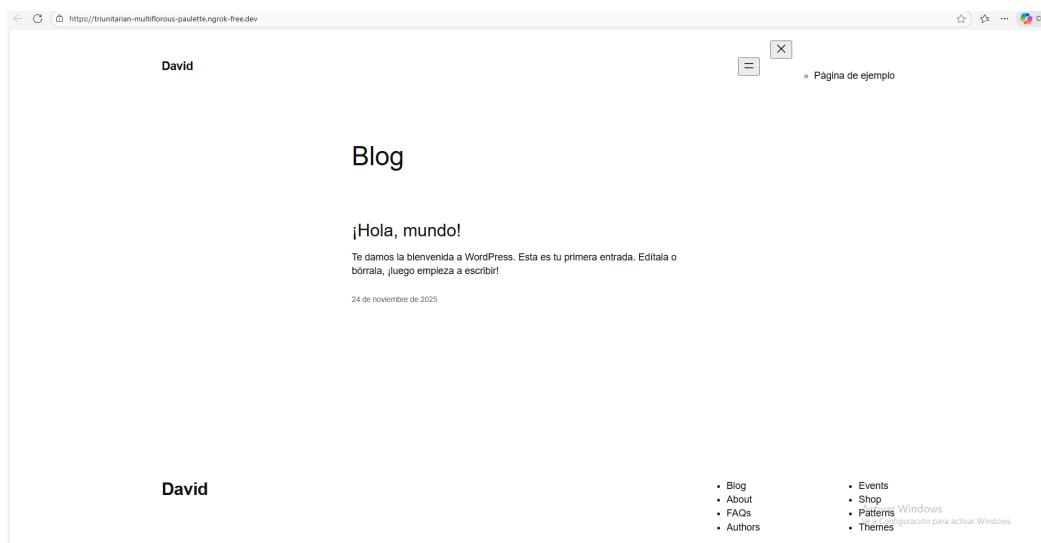
mysql> exit
```

Activar Windows  
Ve a Configuración para activar Windows.

<Imagen 9. Captura de la sesión MySQL mostrando los comandos UPDATE ejecutados correctamente para cambiar las URLs.>

## 10. Prueba Final de Acceso Remoto

Se accede a la URL proporcionada por ngrok desde el navegador local. Se verifica que el sitio carga correctamente con HTTPS (candado seguro).



<Imagen 10. Navegador mostrando el sitio WordPress cargado a través de la URL de ngrok con conexión segura HTTPS.>

También se verifica el certificado SSL que se está usando y los diferentes parámetros.



<Imagen 11. Revisión del Certificado SSL .>

## NOTAS FINALES

- **Seguridad:** Se ha utilizado autenticación basada en claves SSH (Ed25519/RSA) en lugar de contraseñas, y grupos de seguridad restrictivos.
- **Automatización:** El uso de scripts Bash reduce el error humano y acelera el despliegue.
- **Ngrok:** Se utiliza como solución temporal para exponer el servicio sin configurar DNS o Elastic IPs estáticas, ideal para entornos de desarrollo.