

PRÁCTICA 5º:
INSTALACIÓN DE WORDPRESS EN AWS
CON SSH
EVELYN ARCENTALES



Objetivos.....	2
Requisitos Previos.....	2
PASO 1: Preparación en tu PC Local (Localhost).....	3
1.1 Crear directorio SSH.....	3
1.2 – Generar clave SSH con Ed25519.....	3
PARTE 1: CONFIGURACIÓN EN AWS.....	3
1.2 – Transferir la clave descargada a WSL.....	5
1.4 – Configurar reglas de entrada del Security Group.....	6
1.5 – Crear instancia EC2.....	7
1. Accede a EC2 → Instances → Launch instances.....	7
2. Nombre de instancia: wordpress-server.....	7
3. AMI: Ubuntu 24.04 LTS.....	8
6.VPC setting.....	9
7. Security group: Selecciona sg-wordpress-aws.....	9
8. Storage: 15 GiB, gp3.....	9
9. Haz clic en "Launch instance".....	9
PARTE 2: CONEXIÓN SSH DESDE WSL A AWS.....	10
PARTE 3: INSTALACIÓN BASE DEL SERVIDOR (EN AWS).....	11
3.2 – Instalar LAMP Stack.....	11
3.3 – Iniciar servicios.....	12
3.4 – Verificar servicios.....	12
PARTE 4: SCRIPT DE AUTOMATIZACIÓN DE WORDPRESS.....	13
4.1 – Crear script de instalación.....	13
PARTE 5: MIGRACIÓN DE ARCHIVOS CON SCP.....	14
5.1 – Transferir el script a AWS.....	14
5.2 – Dar permisos de ejecución.....	14
5.3 – Ejecutar el script.....	14
PARTE 6: VERIFICACIÓN DE INSTALACIÓN.....	15
6.1 – Verificar servicios.....	15
6.2 – Acceder desde navegador local.....	16
PARTE 7: HACER WORDPRESS ACCESIBLE DESDE INTERNET CON NGROK.....	17
7.1 – Instalar ngrok en AWS.....	17
7.2 – Autenticar ngrok.....	17
7.5 – Probar acceso remoto.....	18

Objetivos

Al finalizar esta práctica, serás capaz de:

- Generar claves SSH con Ed25519/RSA usando keygen
- Configurar Security Groups en AWS
- Conectarte a una instancia EC2 mediante SSH
- Migrar archivos de WordPress usando SCP
- Automatizar la instalación y configuración de WordPress con un script
- Acceder a WordPress desde Internet usando ngrok

Requisitos Previos

- Cuenta en AWS (Free Tier)
- WSL2 o máquina virtual con Ubuntu 22.04 o superior
- Acceso a Internet
- Terminal/PowerShell en Windows o Terminal en Linux/Mac
- Cuenta en ngrok.com

PASO 1: Preparación en tu PC Local (Localhost)

1.1 Crear directorio SSH

```
umno05@A6Alumno05:~$ mkdir -p ~/.ssh
umno05@A6Alumno05:~$ chmod 700 ~/.ssh
umno05@A6Alumno05:~$
```

1.2 – Generar clave SSH con Ed25519

```
ssh-keygen -t ed25519 -f ~/.ssh/wordpress-key -C "mi-usuario@aws"
```

```
lumno05@A6Alumno05:~$ ssh-keygen -t ed25519 -f ~/.ssh/wordpress-key -C "evelyn@aws"
Generating public/private ed25519 key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/lumno05/.ssh/wordpress-key
Your public key has been saved in /home/lumno05/.ssh/wordpress-key.pub
The key fingerprint is:
SHA256:DzSkWwXwYGz8Yjt54byqdqkbxjifrMzM8o7r3r/4CJ8 evelyn@aws
The key's randomart image is:
+--[ED25519 256]--+
  o+.O..
  .+= .
  ...=
  o+o.
  ..*S.
. .o + +o
Bo + + ..
.0==oo .
=*oEXOo.
+----[SHA256]-----+
lumno05@A6Alumno05:~$
```

Verifica que se creó correctamente:

```
chmod 400 ~/.ssh/wordpress-key
```

```
lumno05@A6A1umno05:~$ chmod 400 ~/.ssh/wordpress-key
lumno05@A6A1umno05:~$
```

```
ls -la ~/.ssh/wordpress-key*
```

```
lumno05@A6A1umno05:~$ ls -la ~/.ssh/wordpress-key
-r----- 1 lumno05 lumno05 444 Dec  2 14:12 /home/lumno05/.ssh/wordpress-key
lumno05@A6A1umno05:~$
```

PARTE 1: CONFIGURACIÓN EN AWS

1.1 – Crear par de claves en AWS

1. Accede a AWS Console → EC2 → Key Pairs

EC2

Panel

Vista global de EC2

Eventos

Instancias

Instancias

Tipos de instancia

Plantillas de lanzamiento

Solicitudes de spot

Savings Plans

Instancias reservadas

Alojamientos dedicados

Reservas de capacidad

Recursos

Actualmente, utiliza los siguientes recursos de Amazon EC2 en la región Estados Unidos (Norte de Virginia):

Instancias (en ejecución)	1	Balanceadores de carga	0
Capacity Reservations	0	Direcciones IP elásticas	0
Grupos de escalamiento automático	0	Grupos de seguridad	3
Grupos de ubicación	0	Hosts dedicados	0
Instancias	1	Instantáneas	0
Pares de claves	1	Volúmenes	1

2. Haz clic en "Create key pair"

Pares de claves (1) Información

Acciones

Crear par de claves

Buscar Par de claves por atributo o etiqueta

<input type="checkbox"/>	Nombre	Tipo	Se ha creado	Huella digital
<input type="checkbox"/>	vockey	rsa	2025/11/04 09:05 GMT+1	0e:45:c6:a7:c0:53:49:7

3. Nombre: wordpress-key-aws

4. Type: Ed25519 (o RSA si tu AWS no soporta Ed25519)

5. File format: .pem

6. Clic en "Create key pair" Se descargará un archivo wordpress-key-aws.pem

EC2 > Pares de claves > Crear par de claves

Crear par de claves Información

Par de claves
Un par de claves, compuesto por una clave privada y una clave pública, es un conjunto de credenciales de seguridad que se utilizan para demostrar su identidad cuando se conecta a una instancia.

Nombre
wordpress-key-aws
El nombre puede incluir hasta 255 caracteres ASCII. No puede incluir espacios al principio ni al final.

Tipo de par de claves Información
☐ RSA
 ☒ ED25519

Formato de archivo de clave privada
☒ .pem
Para usar con OpenSSH
☐ .ppk
Para usar con PuTTY

Etiquetas - opcional
No hay etiquetas asociadas a este recurso.


[Agregar nueva etiqueta](#)
Puede agregar hasta 50 etiquetas más.

[Cancelar](#) [Crear par de claves](#)

Confirmación de claves creadas

<input type="checkbox"/>	Nombre	Tipo	Se ha creado	Huella digital	ID
<input type="checkbox"/>	vockey	rsa	2025/11/04 09:05 GMT+1	0e:45:c6:a7:c0:53:49:77:...	key-0e6dd30f4dae7c534
<input type="checkbox"/>	wordpress-key-aws	ed25519	2025/12/02 14:28 GMT+1	1W4HUJpoCbXqXLFNW...	key-0db525a77262a49f3

Se descargará un archivo wordpress-key-aws.pem

▼ hoy			
 wordpress-key-aws.pem	02/12/2025 14:28	Archivo PEM	1 KB

1.2 – Trasferir la clave descargada a WSL

```
lumno05@A6Alumno05:~$ cp /mnt/c/Users/Alumno.DESKTOP-DI5KTUG/Downloads/wordpress-key-aws.pem ~/.ssh/
lumno05@A6Alumno05:~$
```

chmod 400 ~/.ssh/wordpress-key-aws.pem

```
lumno05@A6Alumno05:~$ chmod 400 ~/.ssh/wordpress-key-aws.pem
lumno05@A6Alumno05:~$
```

1.4 – Configurar reglas de entrada del Security Group

Añade las siguientes reglas de entrada:

Reglas de entrada

Regla de entrada 1

TipoSSH

ProtocoloTCP

Intervalo de puertos22

Tipo de origenAnywhere-IPv4

Origen0.0.0.0/0

Descripción: opcional

Eliminar

Regla de entrada 2

TipoHTTP

ProtocoloTCP

Intervalo de puertos80

Tipo de origenAnywhere-IPv4

Origen0.0.0.0/0

Descripción: opcional

Eliminar

Regla de entrada 3

TipoHTTPS

ProtocoloTCP

Intervalo de puertos443

Tipo de origenAnywhere-IPv4

Origen0.0.0.0/0

Descripción: opcional

Eliminar

Agregar regla

Activar Windows
Ve a Configuración para activar Windows.

Grupos creados

sg-0cab689575ea0dbcd - wordpress-aws-sg

Acciones

Detalles

Nombre del grupo de seguridad

wordpress-aws-sg

ID del grupo de seguridad

sg-0cab689575ea0dbcd

Descripción

Security group para WordPress en AWS

ID de la VPC

[vpc-010e2cd53f55d1fac](#)

Propietario

987232549607

Número de reglas de entrada

3 Entradas de permisos

Número de reglas de salida

1 Entrada de permiso

Reglas de entrada

Reglas de salida

Compartiendo : *novedad*

Asociaciones de VPC : *novedad*

Etiquetas

Reglas de entrada (3)

Administrar etiquetas

Editar reglas de entrada

Buscar

< 1 >

<input type="checkbox"/>	Name	ID de la regla del gr...	Versión de IP	Tipo	Protocolo
<input type="checkbox"/>	-	sgr-02f7a7c49b6994b07	IPv4	SSH	TCP
<input type="checkbox"/>	-	sgr-073e99f2abdb8aafb	IPv4	HTTP	TCP
<input type="checkbox"/>	-	sgr-0b2dd53b33d10337d	IPv4	HTTPS	TCP

1.5 – Crear instancia EC2

1. Accede a EC2 → Instances → Launch instances
2. Nombre de instancia: wordpress-server

Lanzar una instancia

Información

Amazon EC2 le permite crear máquinas virtuales, o instancias, que se ejecutan en la nube de AWS. Comience rápidamente siguiendo los sencillos pasos en la continuación.

Nombre y etiquetas

Información

Nombre

wordpress-server

Agregar etiquetas adicionales

3. AMI: Ubuntu 24.04 LTS

▼ **Imágenes de aplicaciones y sistemas operativos (Imagen de máquina de Amazon)**

Una AMI posee el sistema operativo, el servidor de aplicaciones y las aplicaciones de la instancia. Si a continuación no ve búsqueda o elija **Buscar más AMI**.

🔍 *Busque en nuestro catálogo completo que incluye miles de imágenes de sistemas operativos y aplicaciones*

Recientes

Inicio rápido

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Imágenes de máquina de Amazon (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-0ecb62995f68bb549 (64 bits (x86)) / ami-01b9f1e7dc427266e (64 bits (Arm))
Virtualización: hvm Activado para ENA: true Tipo de dispositivo raíz: ebs

4. Instance type: t3.micro (Free Tier)

▼ **Tipo de instancia** [Información](#) | [Obtener asesoramiento](#)

Tipo de instancia

t3.micro

Apto para la capa gratuita

Familia: t3 2 vCPU 1 GiB Memoria Generación actual: true

Bajo demanda Ubuntu Pro base precios: 0.0139 USD por hora Bajo demanda SUSE base precios: 0.0104 USD por hora

Bajo demanda Linux base precios: 0.0104 USD por hora Bajo demanda RHEL base precios: 0.0392 USD por hora

Bajo demanda Windows base precios: 0.0196 USD por hora

[Se aplican costos adicionales a las AMI con software preinstalado](#)

5. Key pair: Selecciona wordpress-key-aws

▼ **Par de claves (inicio de sesión)** [Información](#)

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

wordpress-key-aws

🔄 Crear un nuevo par de claves

6.VPC setting

▼ Configuraciones de red Información

Red | Información

vpc-010e2cd53f55d1fac

Subred | Información

Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

7. Security group: Selecciona sg-wordpress-aws

☐ Crear grupo de seguridad

☒ Seleccionar un grupo de seguridad existente

Grupos de seguridad comunes | Información

Seleccionar grupos de seguridad

wordpress-aws-sg sg-0cab689575ea0dbcd X
VPC: vpc-010e2cd53f55d1fac

Compare reglas de grupo de seguridad

Los grupos de seguridad que agrega o elimine aquí se agregarán a todas las interfaces de red o se eliminarán de ellas.

8. Storage: 15 GiB, gp3

▼ Configurar almacenamiento Información

1x 15 GiB gp3 Volumen raíz, 3000 IOPS, No cifrado

Agregar un nuevo volumen

9. Haz clic en "Launch instance"

Instancias (1/3) Información

Última actualización
Hace less than a minute

Conectar

Estado de la instancia

Acciones

Lanzar instancias

Buscar instancia por atributo o etiqueta (case-sensitive)

Todos los estados

<input checked="" type="checkbox"/>	Name	ID de la instancia	Estado de la i...	Tipo de inst...	Comprobación de	Estado de la al.	Zona de dispon...	DNS de IPv4 pública	Dirección IP...	IP elástica	Direcciones I...	Monit
<input type="checkbox"/>	mi-servidor-web	i-016a4b22d5065ebfb	En ejecución	t3.micro	3/3 comprobación	Ver alarmas +	us-east-1a	ec2-18-209-20-236.co...	18.209.20.236	-	-	disable
<input type="checkbox"/>	wordpress-ser...	i-03c0dbcb6417a809	Terminada	t3.micro	-	Ver alarmas +	us-east-1f	-	-	-	-	disable
<input checked="" type="checkbox"/>	wordpress-ser...	i-083f9fcc42335fb6b1	En ejecución	t3.micro	Iniciando	Ver alarmas +	us-east-1f	ec2-3-238-85-162.com...	3.238.85.162	-	-	disable

1.6 – Obtener la IP pública

1. Selecciona la instancia
2. Copia la "Public IPv4 address" 172.31.78.147

i-083f9fcc42333f6b1 (wordpress-server)

<

Detalles

Estado y alarmas

Monitoreo

Seguridad

Redes

Almacenamiento

E

>

▼ Resumen de instancia

Información

ID de la instancia	Dirección IPv4 pública	Direcciones IPv4 privadas
i-083f9fcc42333f6b1	3.238.85.162 dirección abierta	172.31.78.147

PARTE 2: CONEXIÓN SSH DESDE WSL A AWS

2.1 – Conectar a la instancia

Reemplaza TU-IP-PUBLICA con la IP obtenida:

`ssh -i ~/.ssh/wordpress-key-aws.pem ubuntu@TU-IP-PUBLICA`

Ejemplo: `ssh -i ~/.ssh/wordpress-key-aws.pem ubuntu@54.123.45.67`

La primera vez, te pedirá confirmar la huella digital del servidor. Escribe yes.

```
lumno05@A6A1lumno05:~$ ssh -i ~/.ssh/wordpress-key-aws.pem ubuntu@3.238.85.162
The authenticity of host '3.238.85.162 (3.238.85.162)' can't be established.
ED25519 key fingerprint is SHA256:eqxropvx3B/1TmvFyEB1SdcFWhgIALW4hUhVHH362sI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

```
Warning: Permanently added '3.238.85.162' (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 Tue Dec 9 08:18:35 UTC 2025
```

```
System load:  0.0      Temperature:    -273.1 C
Usage of /:   12.8% of 13.49GB Processes:      111
Memory usage: 22%     Users logged in: 0
Swap usage:   0%      IPv4 address for ens5: 172.31.78.147
```

```
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.
```

```
ubuntu@ip-172-31-78-147:~$
```

"El navegador muestra que el endpoint de Ngrok no está disponible (Tunnel not found), indicando que no hay una sesión activa para esa dirección específica."

PARTE 3: INSTALACIÓN BASE DEL SERVIDOR (EN AWS)

3.1 – Actualizar el sistema

sudo apt update

```
ubuntu@ip-172-31-78-147:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1627 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [305 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [175 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [15.7 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1501 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [304 kB]
```

sudo apt upgrade -y

```
ubuntu@ip-172-31-78-147:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
  linux-aws-6.14-headers-6.14.0-1017 linux-aws-6.14-tools-6.14.0-1017 linux-headers-6.14.0-1017-aws
  linux-image-6.14.0-1017-aws linux-modules-6.14.0-1017-aws linux-tools-6.14.0-1017-aws
The following upgrades have been deferred due to phasing:
  apparmor libapparmor1
The following packages will be upgraded:
  bind9-dnsutils bind9-host bind9-libs gir1.2-glib-2.0 gir1.2-packagekitglib-1.0 intel-microcode
  landscape-common libdrm-common libdrm2 libglib2.0-0t64 libglib2.0-bin libglib2.0-data libnetplan1
  libpackagekit-glib2-18 libpython3-stdlib libpython3.12-minimal libpython3.12-stdlib libpython3.12t64
  libssh-4 libxml2 linux-aws linux-headers-aws linux-image-aws linux-tools-common netplan-generator
  netplan.io packagekit packagekit-tools python3 python3-minimal python3-netplan python3.12
  python3.12-minimal snapd ubuntu-pro-client ubuntu-pro-client-l10n
36 upgraded, 6 newly installed, 0 to remove and 2 not upgraded.
18 standard LTS security updates
Need to get 133 MB of archives.
After this operation, 179 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 ubuntu-pro-client-l10n amd64 37.1ubuntu0~24.04 [19.8 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 ubuntu-pro-client amd64 37.1ubuntu0~24.04 [259 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 python3-minimal amd64 3.12.3-0ubuntu2.1 [27.4 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 python3 amd64 3.12.3-0ubuntu2.1 [23.0 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libpython3.12t64 amd64 3.12.3-1ubuntu0.9 [2342 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 python3.12 amd64 3.12.3-1ubuntu0.9 [651 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libpython3.12-stdlib amd64 3.12.3-1ubuntu0.9 [2068 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 python3.12-minimal amd64 3.12.3-1ubuntu0.9 [2332 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libpython3.12-minimal amd64 3.12.3-1ubuntu0.9 [836 kB]
```

3.2 – Instalar LAMP Stack

`sudo apt install apache2 php php-mysql libapache2-mod-php php-curl php-gd php-mbstring php-xml php-xmllrpc php-intl php-zip mysql-server -y`

```
ubuntu@ip-172-31-78-147:~$ sudo apt install apache2 php php-mysql libapache2-mod-php php-curl php-gd
php-mbstring php-xml php-xmllrpc php-intl php-zip mysql-server -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils fontconfig-config fonts-dejavu-core fonts-dejavu-mono libaom3
  libapache2-mod-php8.3 libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 libde265-0
  libdeflate0 libfontconfig1 libgd3 libheif-plugin-aomdec libheif-plugin-aomenc libheif-plugin-libde265
  libheif1 libjpeg-turbo8 libjpeg8 liblerc4 liblua5.4-0 libsharpvuv0 libtiff6 libwebp7 libxpm4
  php-common php8.3 php8.3-cli php8.3-common php8.3-curl php8.3-gd php8.3-mysql php8.3-opcache
  php8.3-readline ssl-cert
```

Configuración de las URLs en la base de datos para permitir el acceso correcto a WordPress a través del túnel seguro de ngrok.

3.3 – Iniciar servicios

`sudo systemctl start apache2`
`sudo systemctl start mysql`

```
ubuntu@ip-172-31-78-147:~$ sudo systemctl start mysql
Failed to start mysql.service: Unit mysql.service not found.
ubuntu@ip-172-31-78-147:~$ sudo apt install mysql-server -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcgi-fast-perl libcgi-pm-perl libclone-perl libencode-locale-perl libevent-pthreads-2.1-7t64 libfcgi-bin
  libfcgi-perl libfcgi0t64 libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl
  libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite32t64 libtimedate-perl
  liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common
  mysql-server-8.0 mysql-server-core-8.0
```

Como se verifica el error no tenia instalado y se procede a la instalación
`sudo systemctl enable apache2`

`sudo systemctl enable mysql`

```
ubuntu@ip-172-31-78-147:~$ sudo systemctl start mysql
ubuntu@ip-172-31-78-147:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
ubuntu@ip-172-31-78-147:~$
```

3.4 – Verificar servicios

`sudo systemctl status apache2`

```
ubuntu@ip-172-31-78-147:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-12-09 08:28:34 UTC; 18min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 19667 (apache2)
    Tasks: 7 (limit: 1008)
  Memory: 13.5M (peak: 15.1M)
     CPU: 113ms
  CGroup: /system.slice/apache2.service
          └─19667 /usr/sbin/apache2 -k start
          └─19672 /usr/sbin/apache2 -k start
          └─19673 /usr/sbin/apache2 -k start
          └─19674 /usr/sbin/apache2 -k start
          └─19675 /usr/sbin/apache2 -k start
          └─19676 /usr/sbin/apache2 -k start
          └─19754 /usr/sbin/apache2 -k start

Dec 09 08:28:34 ip-172-31-78-147 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Dec 09 08:28:34 ip-172-31-78-147 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@ip-172-31-78-147:~$
```

PARTE 4: SCRIPT DE AUTOMATIZACIÓN DE WORDPRESS

4.1 – Crear script de instalación

```
ubuntu@ip-172-31-78-147:~$ nano install-wordpress.sh
ubuntu@ip-172-31-78-147:~$

GNU nano 7.2                                install-wordpress.sh *
#!/bin/bash
set -e
echo "=== Iniciando instalación automatizada de WordPress ==="

# Variables (¡Esto es lo que faltaba!)
DB_NAME="wordpress"
DB_USER="wpuser"
DB_PASSWORD=$(openssl rand -base64 12)
DB_ROOT_PASSWORD=$(openssl rand -base64 12)
WP_HOME="http://localhost"
WP_SITEURL="http://localhost"

# Paso 1: Configurar MySQL
echo "Configurando MySQL..."
sudo mysql -e "ALTER USER 'root'@'localhost' IDENTIFIED BY '${DB_ROOT_PASSWORD}';"
sudo mysql -e "DELETE FROM mysql.user WHERE User='';"
sudo mysql -e "DELETE FROM mysql.user WHERE User='root' AND Host NOT IN ('localhost', '127.0.0.1', '::1');"
sudo mysql -e "DROP DATABASE IF EXISTS test;"
sudo mysql -e "DELETE FROM mysql.db WHERE Db='test' OR Db='test\\_%';"
sudo mysql -e "FLUSH PRIVILEGES;"

# Paso 2: Crear base de datos y usuario de WordPress
echo "Creando base de datos y usuario..."
sudo mysql -u root -p"${DB_ROOT_PASSWORD}" -e "CREATE DATABASE ${DB_NAME};"
sudo mysql -u root -p"${DB_ROOT_PASSWORD}" -e "CREATE USER '${DB_USER}'@'localhost' IDENTIFIED BY '${DB_PASSWORD}';"
sudo mysql -u root -p"${DB_ROOT_PASSWORD}" -e "GRANT ALL PRIVILEGES ON ${DB_NAME}.* TO '${DB_USER}'@'localhost';"
sudo mysql -u root -p"${DB_ROOT_PASSWORD}" -e "FLUSH PRIVILEGES;"

# Paso 3: Descargar WordPress
echo "Descargando WordPress..."
cd /tmp
wget https://wordpress.org/latest.tar.gz -q
tar -xzf latest.tar.gz

# Paso 4: Instalar WordPress
echo "Copiando archivos a /var/www/html..."
sudo rm -rf /var/www/html/*
sudo cp -r wordpress/* /var/www/html/

# Paso 5: Configurar wp-config.php
echo "Configurando wp-config.php..."
sudo cp /var/www/html/wp-config-sample.php /var/www/html/wp-config.php
sudo sed -i "s/database_name_here/${DB_NAME}/g" /var/www/html/wp-config.php
sudo sed -i "s/username_here/${DB_USER}/g" /var/www/html/wp-config.php
sudo sed -i "s/password_here/${DB_PASSWORD}/g" /var/www/html/wp-config.php

# Paso 6: Permisos
echo "Configurando permisos..."
sudo chown -R www-data:www-data /var/www/html/
sudo chmod -R 755 /var/www/html/

# Paso 7: Habilitar mod_rewrite en Apache
echo "Habilitando mod_rewrite..."
sudo a2enmod rewrite
sudo systemctl restart apache2

# Paso 8: Guardar credenciales
```

```
# Paso 8: Guardar credenciales
echo "Guardando credenciales en archivo..."
cat > ~/wordpress-credentials.txt << EOF
=== CREDENCIALES DE WORDPRESS ===
Base de datos: ${DB_NAME}
Usuario BD: ${DB_USER}
Contraseña BD: ${DB_PASSWORD}
Usuario root MySQL: root
Contraseña root MySQL: ${DB_ROOT_PASSWORD}
Acceso local: http://localhost
EOF

echo "=== Instalación completada ==="
echo "Credenciales guardadas en ~/wordpress-credentials.txt"
echo "Accede a http://3.238.85.162 para finalizar la instalacion de Wordpress"
```

Se crea un script en nano para realizar la configuración respectiva

PARTE 5: MIGRACIÓN DE ARCHIVOS CON SCP

5.1 – Transferir el script a AWS

Desde tu máquina local:

```
ubuntu@ip-172-31-78-147:~$ scp -i ~/.ssh/wordpress-key-aws.pem install-wordpress.sh ubuntu@3.238.85.162:~/
Warning: Identity file /home/ubuntu/.ssh/wordpress-key-aws.pem not accessible: No such file or directory.
The authenticity of host '3.238.85.162 (3.238.85.162)' can't be established.
ED25519 key fingerprint is SHA256:eqxropvx3B/1TmvFyEB1SdcFWHgIALW4hUhVHH362sI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.238.85.162' (ED25519) to the list of known hosts.
ubuntu@3.238.85.162: Permission denied (publickey).
scp: Connection closed
Connection to 3.238.85.162 closed.
lumn05@A6Alumn05:~$
```

5.2 – Dar permisos de ejecución

```
ubuntu@ip-172-31-78-147:~$ nano install-wordpress.sh
ubuntu@ip-172-31-78-147:~$ chmod +x install-wordpress.sh
ubuntu@ip-172-31-78-147:~$ ./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 a /var/www/html...
Configurando wp-config.php...
sed: -e expression #1, char 25: unknown option to `s'
ubuntu@ip-172-31-78-147:~$
```

Desde aquí se da la opción de ejecución con permisos de wordpress , en esta captura se aprecia que hay un error por no tener instalado wordpress.sh

5.3 – Ejecutar el script

```
./install-wordpress.sh
```



```

ubuntu@ip-172-31-78-147:~$ ./install-wordpress.sh
=== Iniciando instalación automatizada de WordPress (Versión Segura) ===
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...
Enabling module rewrite.
To activate the new configuration, you need to run:
    systemctl restart apache2
=== Instalación completada ===
Accede a http://3.238.85.162 para finalizar
ubuntu@ip-172-31-78-147:~$

```

aquí se aprecia la verificación, de la instalación completa

cat ~/wordpress-credentials.txt

Aquí el sistema saca las credenciales creadas

```

ubuntu@ip-172-31-78-147:~$ cat ~/wordpress-credentials.txt
=== CREDENCIALES ===
Base de datos: wordpress
Usuario BD: wpuser
Contraseña BD: kaHHTfIzxcsaJn/s
Usuario root MySQL: root
Contraseña root MySQL: 6sW1l77JC49iK0PM
ubuntu@ip-172-31-78-147:~$

```

con el comando indicado se puede verificar las credenciales necesarias para el siguiente acceso

PARTE 6: VERIFICACIÓN DE INSTALACIÓN

6.1 – Verificar servicios

sudo systemctl status apache2

```

ubuntu@ip-172-31-78-147:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-12-09 09:27:50 UTC; 10min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Process: 21177 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 21181 (apache2)
    Tasks: 6 (limit: 1008)
   Memory: 15.6M (peak: 15.7M)
      CPU: 78ms
   CGroup: /system.slice/apache2.service
           └─21181 /usr/sbin/apache2 -k start
             └─21184 /usr/sbin/apache2 -k start
               └─21185 /usr/sbin/apache2 -k start
                 └─21186 /usr/sbin/apache2 -k start
                   └─21187 /usr/sbin/apache2 -k start
                     └─21188 /usr/sbin/apache2 -k start



Dec 09 09:27:50 ip-172-31-78-147 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Dec 09 09:27:50 ip-172-31-78-147 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@ip-172-31-78-147:~$

```

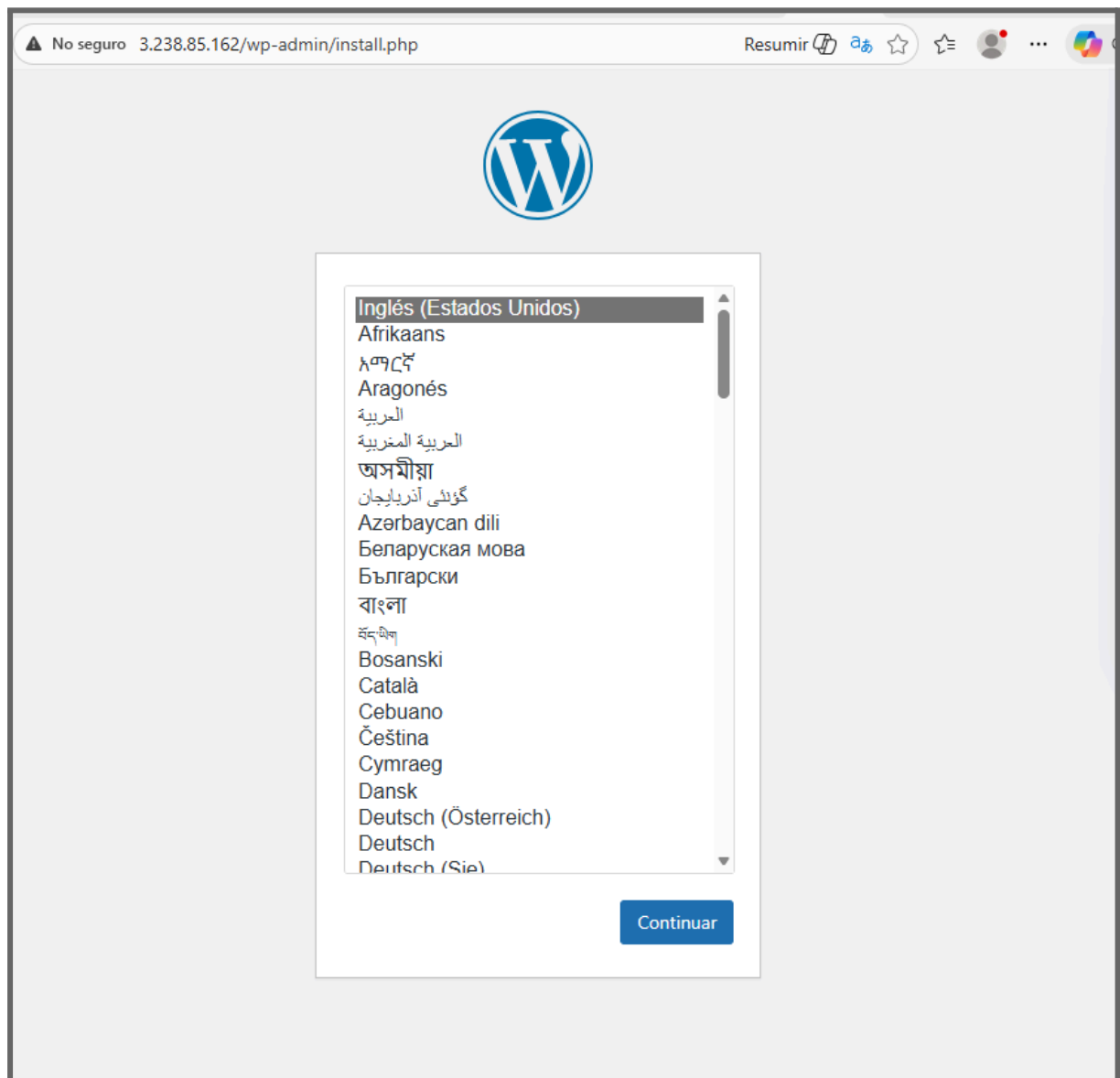
sudo systemctl status mysql

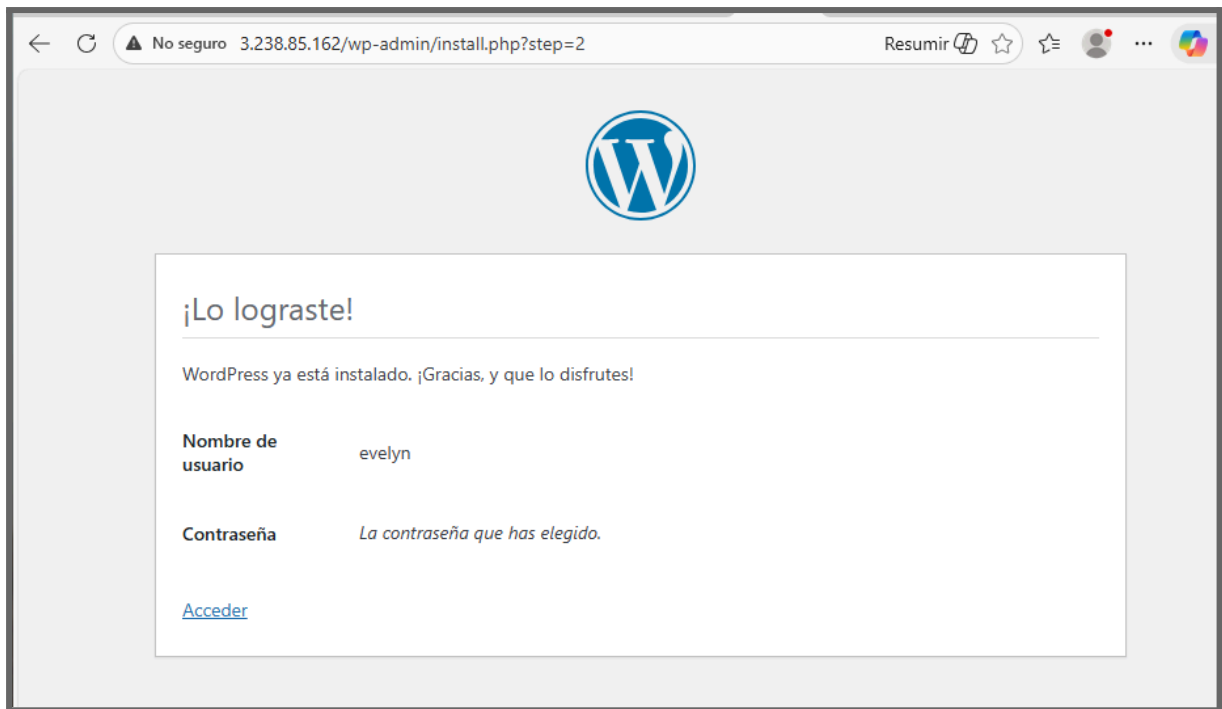
6.2 – Acceder desde navegador local

Dirección IPv4 pública

 3.238.85.162 | [dirección abierta](#) 

http://3.238.85.162





PARTE 7: HACER WORDPRESS ACCESIBLE DESDE INTERNET CON NGROK

7.1 – Instalar ngrok en AWS

```
ubuntu@ip-172-31-78-147:~$ cd ~
ubuntu@ip-172-31-78-147:~$ wget https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tgz
--2025-12-09 09:48:26-- https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tgz
Resolving bin.equinox.io (bin.equinox.io)... 99.83.220.108, 13.248.244.96, 35.71.179.82, ...
Connecting to bin.equinox.io (bin.equinox.io)|99.83.220.108|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11284206 (11M) [application/octet-stream]
Saving to: 'ngrok-v3-stable-linux-amd64.tgz'

ngrok-v3-stable-linux-amd64. 100%[=====] 10.76M --.-KB/s in 0.06s

2025-12-09 09:48:26 (172 MB/s) - 'ngrok-v3-stable-linux-amd64.tgz' saved [11284206/11284206]

ubuntu@ip-172-31-78-147:~$ tar -xvzf ngrok-v3-stable-linux-amd64.tgz
ngrok
ubuntu@ip-172-31-78-147:~$ sudo mv ngrok /usr/local/bin/
ubuntu@ip-172-31-78-147:~$
```

Verificación de instalación correcta

```
ubuntu@ip-172-31-78-147:~$ ngrok --version
ngrok version 3.34.0
ubuntu@ip-172-31-78-147:~$
```

7.2 – Autenticar ngrok

```
ubuntu@ip-172-31-78-147:~$ ngrok config add-authtoken 3514BpFlc2NsNPe3EmSMuf9zJEB_4dXR1vBe9V4sFUQyM2Bj4
authtoken saved to configuration file: /home/ubuntu/.config/ngrok/ngrok.yml
```

```
ngrok (Ctrl+C to quit)
! Free Users: Agents ≤3.18.x stop connecting 12/17/25. Update or upgrade: https://ngrok.com/pricing

Session Status      online
Account             evelyn14-1984 (Plan: Free)
Version             3.34.0
Region              United States (us)
Latency             11ms
Web Interface       http://127.0.0.1:4040
Forwarding           https://unprefigured-lilliana-unspecifiable.ngrok-free.dev -> http://localhost:80

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

pantalla con el primer puente para acceder con el ssh

Contraseña de credenciales

kaHHTflzxcsaJn/s

```
ubuntu@ip-172-31-78-147:~$ mysql -u wpuser -p -D wordpress
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 59
Server version: 8.0.44-0ubuntu0.24.04.2 (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.
```

Ejecuta (reemplaza la URL):

```
UPDATE wp_options SET
option_value='https://unprefigured-lilliana-unspecifiable.ngrok-free.dev' WHERE
option_name='siteurl';
UPDATE wp_options SET
option_value='https://unprefigured-lilliana-unspecifiable.ngrok-free.dev' WHERE
option_name='home';
```

```
mysql> UPDATE wp_options SET option_value='https://unprefigured-lilliana-unspecifiable.ngrok-free.dev' WHERE o
option_name='siteurl';
on_value='https://unprefigured-lilliana-unspecifiable.ngrok-free.dev' WHERE option_name='home';Query OK, 1 row
affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

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

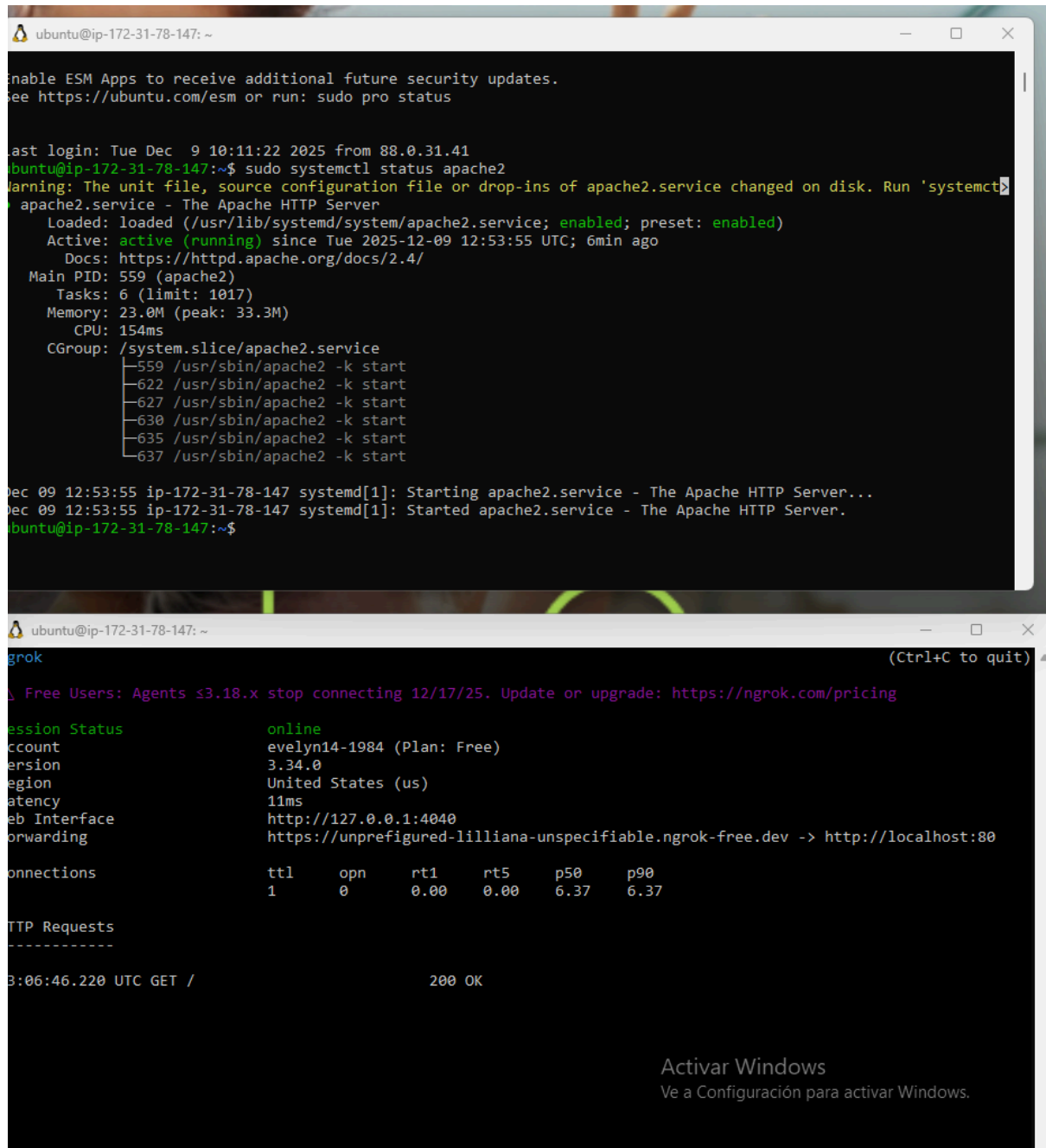
mysql>
```

desde aquí ya se ha podido realizar los comandos respectivos .

7.5 – Probar acceso remoto

En las capturas se aprecian dos maquinas de swl para hacer la comprobacion que ngrok pasa a traves del tunel de ngrok http 80 y ya con

<https://unprefigured-lilliana-unspecifiable.ngrok-free.dev> se visualiza la pantalla con url



The image consists of two terminal screenshots. The top screenshot shows the command `sudo systemctl status apache2` being executed on an Ubuntu machine. The output displays the status of the `apache2.service`, which is `loaded` and `enabled`. It also shows the `systemctl start apache2` command being run, which successfully starts the service. The bottom screenshot shows the `ngrok` command being executed. The output displays the ngrok tunnel status, including the tunnel ID `evelyn14-1984`, the plan `Free`, and the public URL `https://unprefigured-lilliana-unspecifiable.ngrok-free.dev`. It also shows the local interface `http://127.0.0.1:4040` and the forwarding rule `https://unprefigured-lilliana-unspecifiable.ngrok-free.dev -> http://localhost:80`. The bottom screenshot also shows a list of connections and a table of HTTP requests.

```
ubuntu@ip-172-31-78-147: ~  
enable ESM Apps to receive additional future security updates.  
see https://ubuntu.com/esm or run: sudo pro status  
  
Last login: Tue Dec 9 10:11:22 2025 from 88.0.31.41  
ubuntu@ip-172-31-78-147:~$ sudo systemctl status apache2  
Warning: The unit file, source configuration file or drop-ins of apache2.service changed on disk. Run 'systemctl daemon-reload' to reload units.  
● apache2.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)  
   Active: active (running) since Tue 2025-12-09 12:53:55 UTC; 6min ago  
     Docs: https://httpd.apache.org/docs/2.4/  
   Main PID: 559 (apache2)  
     Tasks: 6 (limit: 1017)  
  Memory: 23.0M (peak: 33.3M)  
     CPU: 154ms  
   CGroup: /system.slice/apache2.service  
           └─559 /usr/sbin/apache2 -k start  
             └─622 /usr/sbin/apache2 -k start  
               └─627 /usr/sbin/apache2 -k start  
                 └─630 /usr/sbin/apache2 -k start  
                   └─635 /usr/sbin/apache2 -k start  
                     └─637 /usr/sbin/apache2 -k start  
  
Dec 09 12:53:55 ip-172-31-78-147 systemd[1]: Starting apache2.service - The Apache HTTP Server...  
Dec 09 12:53:55 ip-172-31-78-147 systemd[1]: Started apache2.service - The Apache HTTP Server.  
ubuntu@ip-172-31-78-147:~$  
  
ngrok (Ctrl+C to quit)  
Free Users: Agents ≤3.18.x stop connecting 12/17/25. Update or upgrade: https://ngrok.com/pricing  
  
Session Status      online  
Account             evelyn14-1984 (Plan: Free)  
Version             3.34.0  
Region              United States (us)  
Latency             11ms  
Web Interface       http://127.0.0.1:4040  
Forwarding          https://unprefigured-lilliana-unspecifiable.ngrok-free.dev -> http://localhost:80  
  
Connections         ttl    opn    rt1    rt5    p50    p90  
                    1      0      0.00   0.00   6.37   6.37  
  
HTTP Requests  
-----  
3:06:46.220 UTC GET / 200 OK  
  
Activar Windows  
Ve a Configuración para activar Windows.
```

Blog

¡Hola, mundo!

Te damos la bienvenida a WordPress. Esta es tu primera entrada. Edítala o bórrala, ¡luego empieza a escribir!

9 de diciembre de 2025

resultado final con la URL respectiva .

