

PRÁCTICA DE INSTALACIÓN DE WORDPRESS EN LA NUBE

Luis Peregrina Ojeda 2ºASIR Caja Mágica

Parte1 Instalar servidor LAMP

Comenzamos actualizando el sistema

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

Ahora instalamos Apache para despues visualizar las paginas de wordpress

```
-sudo apt install apache2 -y  
-sudo systemctl status apache2
```

```
luis@luis-VirtualBox:~$ sudo systemctl status apache2  
● apache2.service - The Apache HTTP Server  
    Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >  
    Active: active (running) since Tue 2025-11-11 08:53:41 CET; 41s ago  
      Docs: https://httpd.apache.org/docs/2.4/  
    Main PID: 14801 (apache2)  
       Tasks: 55 (limit: 4666)  
     Memory: 5.4M (peak: 6.0M)  
        CPU: 44ms
```

Ahora instalamos MySQL, para guardar toda la información de wordpress:

```
-sudo apt install mysql-server -y
```

```
luis@luis-VirtualBox:~$ sudo apt install mysql-server -y  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias... Hecho  
Leyendo la información de estado... Hecho
```

```
-sudo mysql_secure_installation
```

```
luis@luis-VirtualBox:~$ sudo mysql_secure_installation  
  
Securing the MySQL server deployment.  
  
Connecting to MySQL using a blank password.  
  
VALIDATE PASSWORD COMPONENT can be used to test passwords  
and improve security. It checks the strength of password  
and allows the users to set only those passwords which are  
secure enough. Would you like to setup VALIDATE PASSWORD component?  
  
Press y|Y for Yes, any other key for No: N
```

Despues instalamos php, php se instala de normal con apache2 pero en este caso nos ayuda con wordpress ya que si no tengo mal entendido Wordpress usa php de lenguaje

```
luis@luis-VirtualBox:~$ sudo apt install php php-mysql libapache2-mod-php php-curl php-gd php-mbstring php-xml php-xmlrpc php-intl php-zip -y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
  libllvm19
Utilice «sudo apt autoremove» para eliminarlo.
Se instalarán los siguientes paquetes adicionales:
```

Y ahora verificamos php con

```
-php -v
```

```
luis@luis-VirtualBox:~$ php -v
PHP 8.3.6 (cli) (built: Jul 14 2025 18:30:55) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.6, Copyright (c) Zend Technologies
```

Despues reiniciamos el sistema:

```
-sudo systemctl restart apache2
```

PARTE 2 Crear Base de Datos para WordPress

Accedemos con

```
-sudo mysql
```

Y creamos una base de datos con usuario

```
mysql> CREATE DATABASE wordpress;
Query OK, 1 row affected (0,05 sec)

mysql> CREATE USER 'wpuser'@'localhost' IDENTIFIED BY 'WordPress123!';
Query OK, 0 rows affected (0,05 sec)

mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'wpuser'@'localhost'
      -> '^C'
mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'wpuser'@'localhost';
Query OK, 0 rows affected (0,04 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,05 sec)

mysql> EXIT;
Bye
luis@luis-VirtualBox:~$
```

Probamos que mysql funciona

```
luis@luis-VirtualBox:~$ sudo systemctl status mysql
[sudo] contraseña para luis:
● mysql.service - MySQL Community Server
   Loaded: loaded (/usr/lib/systemd/system/mysql.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-11-11 19:03:10 CET; 1h 8min ago
     Main PID: 16448 (mysqld)
    Status: "Server is operational"
```

PARTE 3 Instalar WordPress

Accedemos a /tmp y descargamos wordpress

```
luis@luis-VirtualBox:~$ cd /tmp
luis@luis-VirtualBox:/tmp$ wget https://wordpress.org/latest.tar.gz
--2025-11-11 19:28:11--  https://wordpress.org/latest.tar.gz
Resolviendo wordpress.org (wordpress.org)... 198.143.164.252, 2607:f978:5:8002::c68f:a4fc
Conectando con wordpress.org (wordpress.org)[198.143.164.252]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 26928488 (26M) [application/octet-stream]
Guardando como: 'latest.tar.gz'

latest.tar.gz      100%[=====] 25,68M  17,9MB/s  en 1,4s

2025-11-11 19:28:13 (17,9 MB/s) - 'latest.tar.gz' guardado [26928488/26928488]

luis@luis-VirtualBox:/tmp$ tar -xzf latest.tar.gz
luis@luis-VirtualBox:/tmp$
```

Primero vaciamos los contenidos de apache y copiamos los archivos en wordpress

-sudo rm -rf /var/www/html/*

-sudo cp -r wordpress/* /var/www/html/

```
luis@luis-VirtualBox:/tmp$ sudo rm -rf /var/www/html/*
luis@luis-VirtualBox:/tmp$ sudo cp -r wordpress/* /var/www/html/
luis@luis-VirtualBox:/tmp$
```

Ponemos los permisos con

-sudo chown -R www-data:www-data /var/www/html/

-sudo chmod -R 755 /var/www/html/

```
luis@luis-VirtualBox:/tmp$ sudo chown -R www-data:www-data /var/www/html/
luis@luis-VirtualBox:/tmp$ sudo chmod -R 755 /var/www/html/
luis@luis-VirtualBox:/tmp$
```

Ahora configuramos wp-config.php

-sudo cp /var/www/html/wp-config-sample.php /var/www/html/wp-config.php

-sudo nano /var/www/html/wp-config.php

```
luis@luis-VirtualBox:/tmp$ sudo cp /var/www/html/wp-config-sample.php /var/www/html/wp-config.php
luis@luis-VirtualBox:/tmp$ sudo nano /var/www/html/wp-config.php
```

Modificamos estos valores:

```
define( 'DB_NAME', 'wordpress' );
define( 'DB_USER', 'wpuser' );
define( 'DB_PASSWORD', 'WordPress123!' );
define( 'DB_HOST', 'localhost' );
```

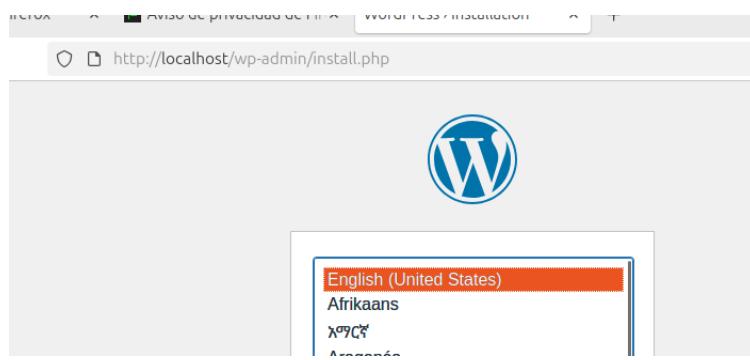
```
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'wpuser' );

/** Database password */
define( 'DB_PASSWORD', 'WordPress123!' );

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

Terminamos la config de wordpress en el navegador con “localhost”



Parte 4

Instalamos ngrok en Ubuntu

```
luis@luis-VirtualBox:~$ wget https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tgz
--2025-11-11 19:42:01--  https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tgz
Resolving bin.equinox.io (bin.equinox.io)... 75.2.60.68, 35.71.179.82, 13.248.244.96.
```

Extraemos el archivo

```
luis@luis-VirtualBox:~$ tar -xvzf ngrok-v3-stable-linux-amd64.tgz
ngrok
```

Movemos ngrok a una ubicación del sistema

-sudo mv ngrok /usr/local/bin/

```
luis@luis-VirtualBox:~$ sudo mv ngrok /usr/local/bin
luis@luis-VirtualBox:~$
```

Verificamos la instalación

-ngrok versión

```
luis@luis-VirtualBox:~$ sudo ./ngrok version
luis@luis-VirtualBox:~$ ngrok version
ngrok version 3.32.0
luis@luis-VirtualBox:~$
```

Autenticar ngrok

-ngrok config add-authtoken (tu token)

```
luis@luis-VirtualBox:~$ ngrok config add-authtoken 35LRc5TiXlrtD7ZAKbNDDueMRAi_6DRVVL2kv98DqMoRRMWZu
Auth token saved to configuration file: /home/luis/.config/ngrok/ngrok.yml
luis@luis-VirtualBox:~$
```

Probamos el puerto 80

-ngrok http 80

```
ngrok
(Ctrl+C to quit)

⚡ Create instant endpoints for local containers within Docker Desktop → https://ngrok.com/r/docker

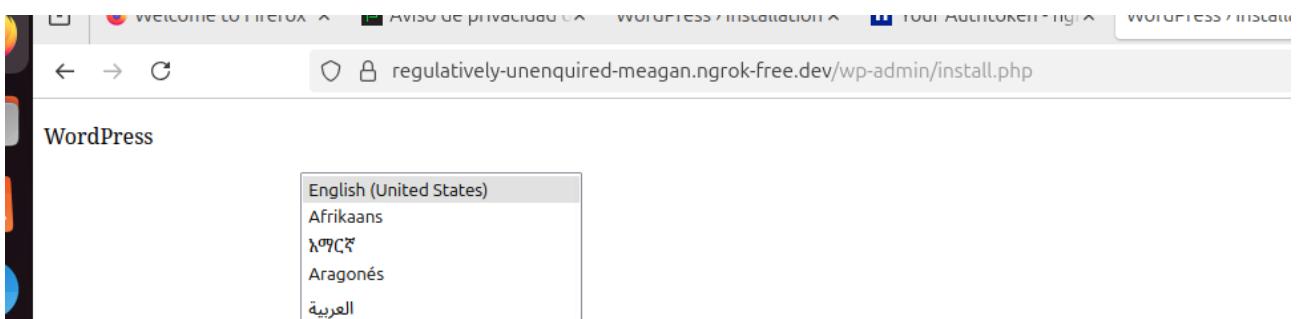
Session Status      online
Account            Luis-440 (Plan: Free)
Version             3.32.0
Region              Europe (eu)
Web Interface       http://127.0.0.1:4040
Forwarding          https://regulatively-unenquired-meagan.ngrok-free.dev -> http://localhost

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

En este caso nos proporciona esta url:

<https://regulatively-unenquired-meagan.ngrok-free.dev> -> <http://localhost:80>

Usaremos la primera.



Saltamos la advertencia y veremos eso ^^