

# Guía para Compilar PostgreSQL en Ubuntu Server 24.04

## 1 Actualizar el Sistema

Antes de empezar, es recomendable actualizar el sistema para asegurarnos de tener las últimas versiones de los paquetes.

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

## 2 Instalar Dependencias Necesarias

PostgreSQL requiere varias bibliotecas y herramientas para compilarse correctamente.

```
sudo apt install -y build-essential libreadline-dev zlib1g-dev \  
flex bison libxml2-dev libxslt1-dev \  
libssl-dev libperl-dev libpam0g-dev \  
libldap2-dev python3-dev uuid-dev
```

## 3 Descargar la Última Versión de PostgreSQL

Obtenemos la versión más reciente desde el sitio oficial de PostgreSQL.

```
cd /usr/local/src  
wget https://ftp.postgresql.org/pub/source/v16.1/postgresql-16.1.  
tar.gz
```

(Sustituye '16.1' por la última versión disponible si hay una más reciente).

## 4 Extraer el Código Fuente

Descomprimos el archivo descargado.

```
tar -xvzf postgresql-16.1.tar.gz  
cd postgresql-16.1
```

## 5 Configurar la Compilación

Ejecutamos el script de configuración con las opciones necesarias.

```
./configure --prefix=/usr/local/pgsql --with-openssl
```

Opcionalmente, puedes añadir otras opciones como ‘-with-python’, ‘-with-libxml’, etc.

## 6 Compilar PostgreSQL

Ahora compilamos PostgreSQL utilizando ‘make’.

```
make -j$(nproc)
```

## 7 Instalar PostgreSQL

Si la compilación fue exitosa, instalamos PostgreSQL en el sistema.

```
sudo make install
```

## 8 Crear el Usuario y Configurar Directorios

Creamos un usuario específico para PostgreSQL y configuramos los directorios.

```
sudo useradd -m -d /usr/local/pgsql -s /bin/bash postgres
sudo mkdir -p /usr/local/pgsql/data
sudo chown postgres:postgres /usr/local/pgsql/data
```

## 9 Inicializar la Base de Datos

Iniciamos PostgreSQL con el usuario ‘postgres’.

```
sudo -i -u postgres
/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
exit
```

## 10 Configurar el Servicio de PostgreSQL

Creamos un archivo de servicio para ‘systemd’.

```
sudo nano /etc/systemd/system/postgresql.service
```

Pegamos el siguiente contenido:

```
[Unit]
Description=PostgreSQL database server
After=network.target

[Service]
User=postgres
Group=postgres
ExecStart=/usr/local/pgsql/bin/postgres -D /usr/local/pgsql/data
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
KillSignal=SIGINT
Type=forking

[Install]
WantedBy=multi-user.target
```

Guardamos y salimos (‘Ctrl + X’, luego ‘Y’ y ‘Enter’).

## 11 Iniciar y Habilitar PostgreSQL

Recargamos ‘systemd’ y habilitamos PostgreSQL para que se inicie automáticamente.

```
sudo systemctl daemon-reload
sudo systemctl enable postgresql
sudo systemctl start postgresql
```

## 12 Verificar el Estado del Servidor

Para confirmar que PostgreSQL está corriendo correctamente, usamos:

```
sudo systemctl status postgresql
```

Si PostgreSQL está funcionando bien, deberíamos ver un mensaje indicando que está activo.

## 13 Acceder a PostgreSQL

Iniciamos sesión en PostgreSQL con el usuario ‘postgres’.

```
sudo -i -u postgres
/usr/local/pgsql/bin/psql
```

Para salir de la sesión de ‘psql’, usamos:

```
\q
```

Y para salir del usuario ‘postgres’:

```
exit
```

## 14 ¡PostgreSQL está listo para usarse!

Ahora puedes configurar usuarios, bases de datos y tunear PostgreSQL según tus necesidades.