**PARTE 0: PREPARACIÓN DEL ENTORNO LOCAL:**

1. Crear Directorio SSH

```
root@juan-VirtualBox:/home/juan# mkdir -p ~/.ssh
root@juan-VirtualBox:/home/juan# chmod 700 ~/.ssh
```

2. General clave SSH

```
root@juan-VirtualBox:/home/juan# ssh-keygen -t ed25519 -f ~/.ssh/wordpress-key
C "Juan@aws"
Generating public/private ed25519 key pair.
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:ERGd7A195cESiHH+RhByT1Crnpc0niisKz6zYXDwjA4 Juan@aws
The key's randomart image is:
+--[ED25519 256]--+
|        +BoO=+oo.|
|         oX.=.oo.|
|   .    .. + =. .|
|    =    .. =    |
| E o +  S  . =   |
|  o o   . . * +  |
|   . o   o + =   |
|    .+. . . .    |
|    .o=o.        |
+----[SHA256]-----+
```

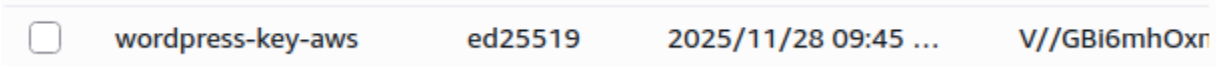
```
root@juan-VirtualBox:/home/juan# ls -la ~/.ssh/wordpress-key*
-rw------- 1 root root 399 nov 28 09:25 /root/.ssh/wordpress-key
-rw-r--r-- 1 root root  90 nov 28 09:25 /root/.ssh/wordpress-key.pub
```

3. Ajustar permisos clave privada

```
root@juan-VirtualBox:/home/juan# chmod 400 ~/.ssh/wordpress-key
root@juan-VirtualBox:/home/juan# ls -la ~/.ssh/wordpress-key
-r-------- 1 root root 399 nov 28 09:25 /root/.ssh/wordpress-key
```

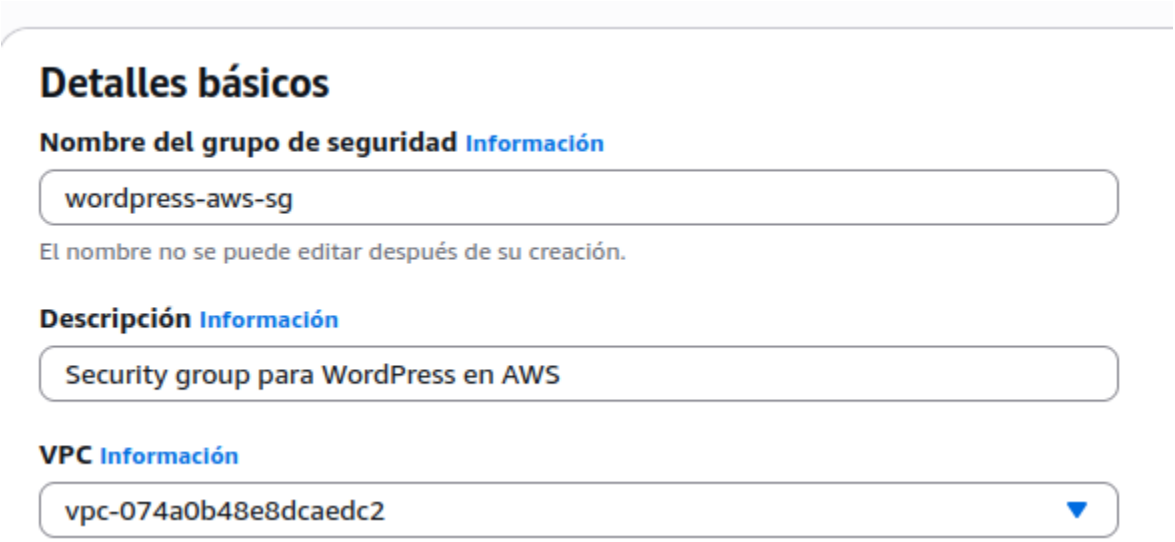
**PARTE 1: CONFIGURACIÓN EN AWS**

1. Crear par de claves en AWS



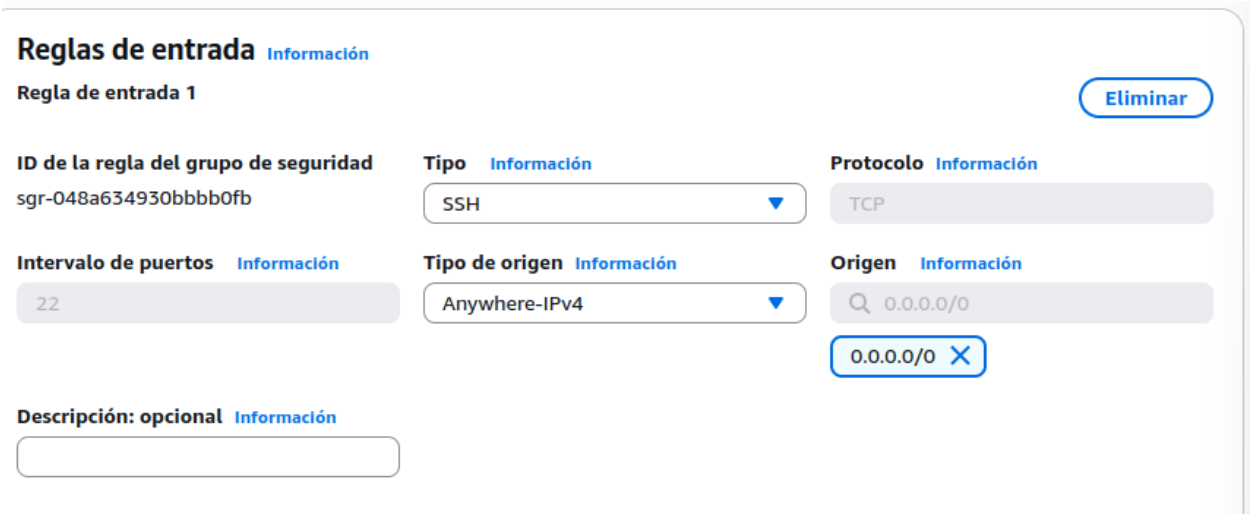

2. Transferir la descarga a AWS
   Como lo hemos hecho en AWS directamente, no tendremos que transferir la descarga

3. Crear el Security Group



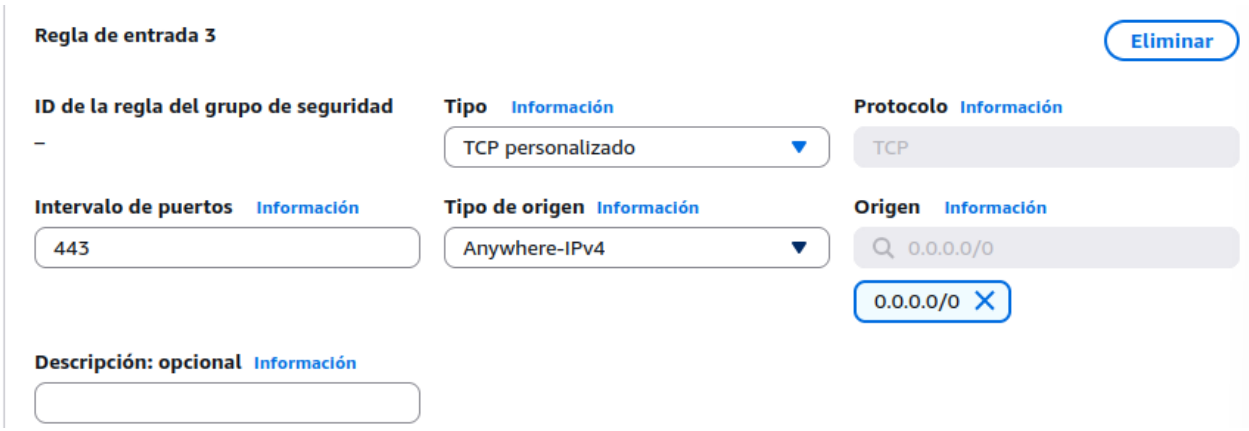

4. Configurar reglas de entrada del Security Group

   REGLA 1:

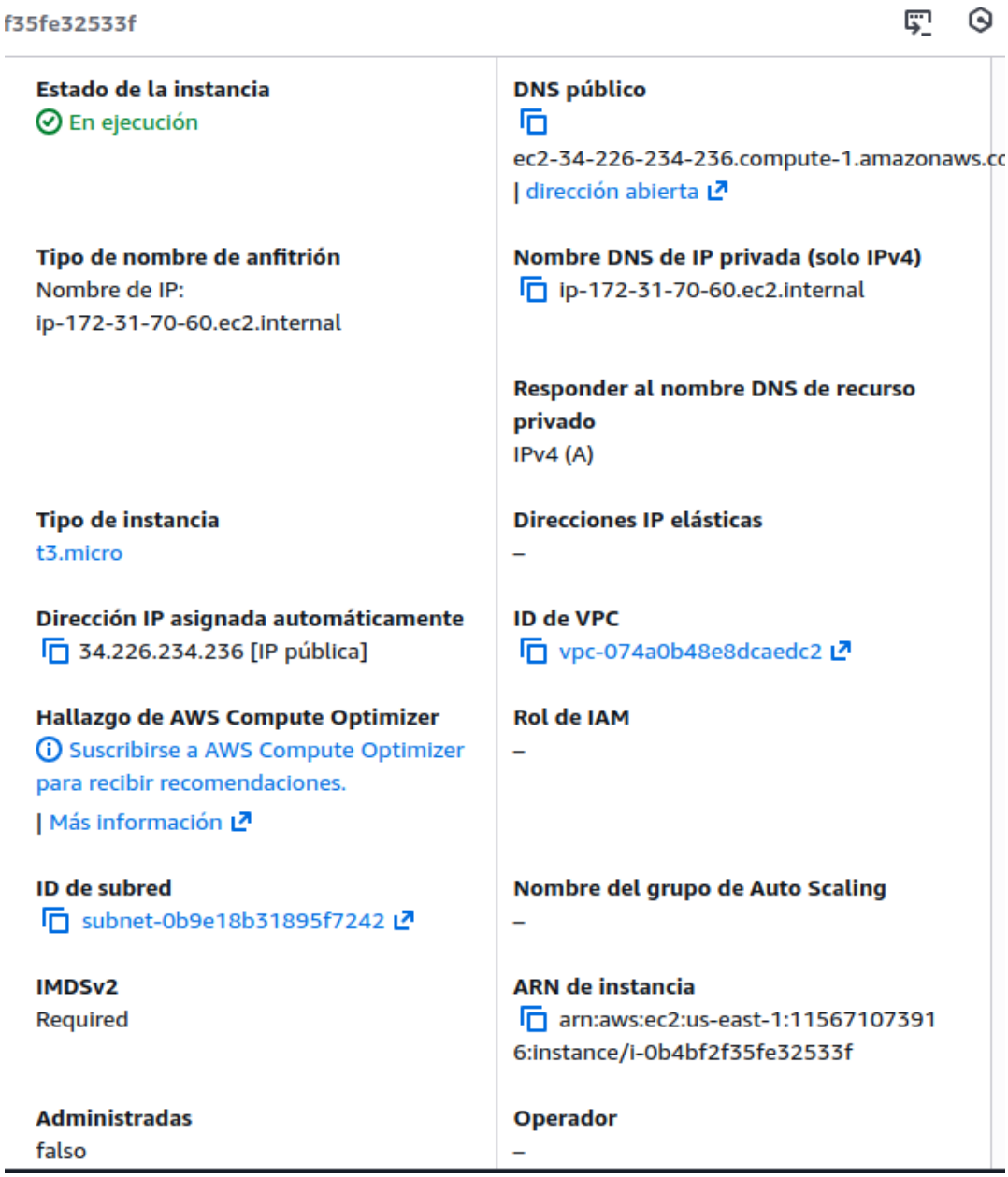



   REGLA 2:


REGLA 3:




5. Crear instancia EC2




6. Obtener IP pública
   34.226.234.236

**PARTE 2: CONEXIÓN SSH DESDE WSL A AWS**

1. Conectar a la instancia

```
root@juan-VirtualBox:/home/juan# ssh -i ~/Descargas/wordpress-key-aws.pem ubuntu
@34.226.234.236
Warning: Identity file /root/Descargas/wordpress-key-aws.pem not accessible: No
such file or directory.
ubuntu@34.226.234.236: Permission denied (publickey).
root@juan-VirtualBox:/home/juan# ssh -i ~/.ssh/wordpress-key-aws.pem ubuntu@34.2
26.234.236
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 Fri Nov 28 09:41:20 UTC 2025

  System load:  0.0               Temperature:           -273.1 C
  Usage of /:   12.9% of 13.49GB  Processes:             110
  Memory usage: 22%               Users logged in:       0
  Swap usage:   0%                IPv4 address for ens5: 172.31.70.60

Expanded Security Maintenance for Applications is not enabled.
```

2. Verificar conexión

```
ubuntu@ip-172-31-70-60:~$
```

**PARTE 3: INSTALACIÓN BASE DEL SERVIDOR (EN AWS)**

1. Actualizar sistema

```
27 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-70-60:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
27 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

```
ubuntu@ip-172-31-70-60:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

2. Instalar LAMP Stack

```
ubuntu@ip-172-31-70-60:~$ sudo apt install apache2 php php-mysql libapache2-mod-
php php-curl php-gd php-mbstring php-xml php-xmlrpc php-intl php-zip mysql-serve
r -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 libcgi-fast-perl
  libcgi-pm-perl libclone-perl libde265-0 libdeflate0 libencode-locale-perl
  libevent-pthreads-2.1-7t64 libfcgi-bin libfcgi-perl libfcgi0t64
  libfontconfig1 libgd3 libheif-plugin-aomdec libheif-plugin-aomenc
  libheif-plugin-libde265 libheif1 libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
  libjbig0 libjpeg-turbo8 libjpeg8 liblerc4 liblua5.4-0 liblwp-mediatypes-perl
  libmecab2 libprotobuf-lite32t64 libsharpyuv0 libtiff6 libtimedate-perl
  liburi-perl libwebp7 libxmlrpc-epi0t64 libxpm4 libzip4t64 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 php-common php8.3
```

3. Iniciar Servicios

```
ubuntu@ip-172-31-70-60:~$ sudo systemctl start apache2
ubuntu@ip-172-31-70-60:~$ sudo systemctl start mysql
ubuntu@ip-172-31-70-60:~$ sudo systemctl enable mysql
Synchronizing state of mysql.service with SysV service script with /usr/lib/syst
emd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable mysql
^[[Aubuntu@ip-172-31-70-60:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/sy
stemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
ubuntu@ip-172-31-70-60:~$
```

4. Verificar servicios

```
ubuntu@ip-172-31-70-60:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >
     Active: active (running) since Fri 2025-11-28 09:57:53 UTC; 2min 51s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 23467 (apache2)
      Tasks: 6 (limit: 1008)
     Memory: 14.7M (peak: 16.7M)
        CPU: 83ms
     CGroup: /system.slice/apache2.service
             ├─23467 /usr/sbin/apache2 -k start
             ├─23472 /usr/sbin/apache2 -k start
             ├─23473 /usr/sbin/apache2 -k start
             ├─23474 /usr/sbin/apache2 -k start
             ├─23475 /usr/sbin/apache2 -k start
             └─23476 /usr/sbin/apache2 -k start

Nov 28 09:57:53 ip-172-31-70-60 systemd[1]: Starting apache2.service - The Apac>
Nov 28 09:57:53 ip-172-31-70-60 systemd[1]: Started apache2.service - The Apach>
lines 1-18/18 (END)
```

**PARTE 4: SCRIPT DE AUTOMATIZACIÓN DE WORDPRESS**

1. Crear script de instalación

```
 GNU nano 7.2                    install-wordpress.sh
echo "Habilitando mod_rewrite..."
sudo a2enmod rewrite
sudo systemctl restart apache2
# 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
Acceso remoto: (se configurará con ngrok)
EOF
echo "=== Instalación completada ==="
echo "Credenciales guardadas en ~/wordpress-credentials.txt"
echo "Accede a http://TU-IP-PUBLICA para finalizar la instalación de WordPress"

                          [ Wrote 67 lines ]
```

PARTE 5: MIGRACIÓN DE ARCHIVOS CON SCP

1. Transferir Script a AWS

```
root@mail:/home/juan# scp -i ~/.ssh/wordpress-key-aws.pem install-wordpress.sh ubuntu@9
8.92.241.209
root@mail:/home/juan#
```

2. Permisos de ejecución

```
ubuntu@ip-172-31-68-246:~$ scp -i ~/.ssh/wordpress-key-aws.pem install-wordpress.sh ubuntu@172.31.68.2
46
```

3. Ejecutar Script

```
ubuntu@ip-172-31-68-246:~$ chmod +x ~/install-wordpress.sh
ubuntu@ip-172-31-68-246:~$ ./install-wordpress.sh
```

PARTE 6: Verificación

1. Verificar servicios

```
root@mail:/home/juan# 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 Fri 2025-12-05 09:40:29 CET; 1h 4min ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 1423 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 1671 (apache2)
      Tasks: 6 (limit: 4600)
     Memory: 19.1M (peak: 19.5M)
        CPU: 966ms
     CGroup: /system.slice/apache2.service
             ├─1671 /usr/sbin/apache2 -k start
             ├─1679 /usr/sbin/apache2 -k start
             ├─1680 /usr/sbin/apache2 -k start
             ├─1681 /usr/sbin/apache2 -k start
             ├─1682 /usr/sbin/apache2 -k start
             └─1683 /usr/sbin/apache2 -k start

dic 05 09:40:28 mail.mi-startup.lan systemd[1]: Starting apache2.service - The Apache >
dic 05 09:40:29 mail.mi-startup.lan systemd[1]: Started apache2.service - The Apache H>
lines 1-19/19 (END)
```

```
● mysql.service - MySQL Community Server
```

2. Navegador local

```
ubuntu@ip-172-31-68-246:~$ nano install-wordpress.sh
ubuntu@ip-172-31-68-246:~$ ./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...
```

PARTE 7: Hacer wordpress accesible con ngrok

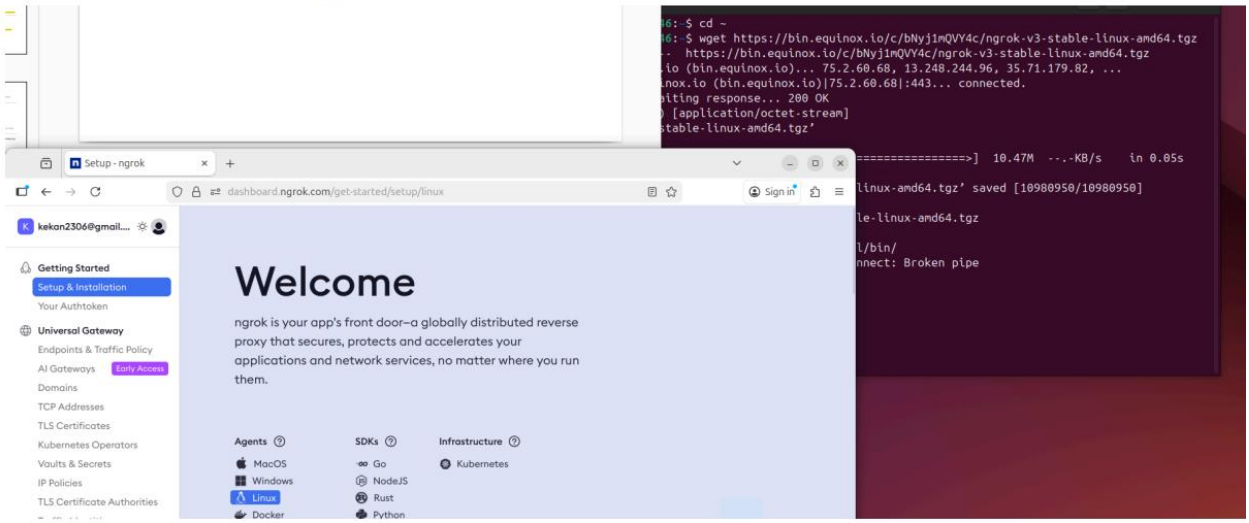
1.Instalar ngrok

```
ubuntu@ip-172-31-68-246:~$ wget https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tg
--2025-11-28 10:02:56--  https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tgz
Resolving bin.equinox.io (bin.equinox.io)... 75.2.60.68, 13.248.244.96, 35.71.179.82, ...
Connecting to bin.equinox.io (bin.equinox.io)|75.2.60.68|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10980950 (10M) [application/octet-stream]
Saving to: 'ngrok-v3-stable-linux-amd64.tgz'

ngrok-v3-stable-linux-amd 100%[===================================>]  10.47M  --.-KB/s    in 0.05s

2025-11-28 10:02:56 (202 MB/s) - 'ngrok-v3-stable-linux-amd64.tgz' saved [10980950/10980950]

ubuntu@ip-172-31-68-246:~$ tar -xvzf ngrok-v3-stable-linux-amd64.tgz
ngrok
ubuntu@ip-172-31-68-246:~$ sudo mv ngrok /usr/local/bin/
```




2. Autenticar ngrok


3. Puerto 80