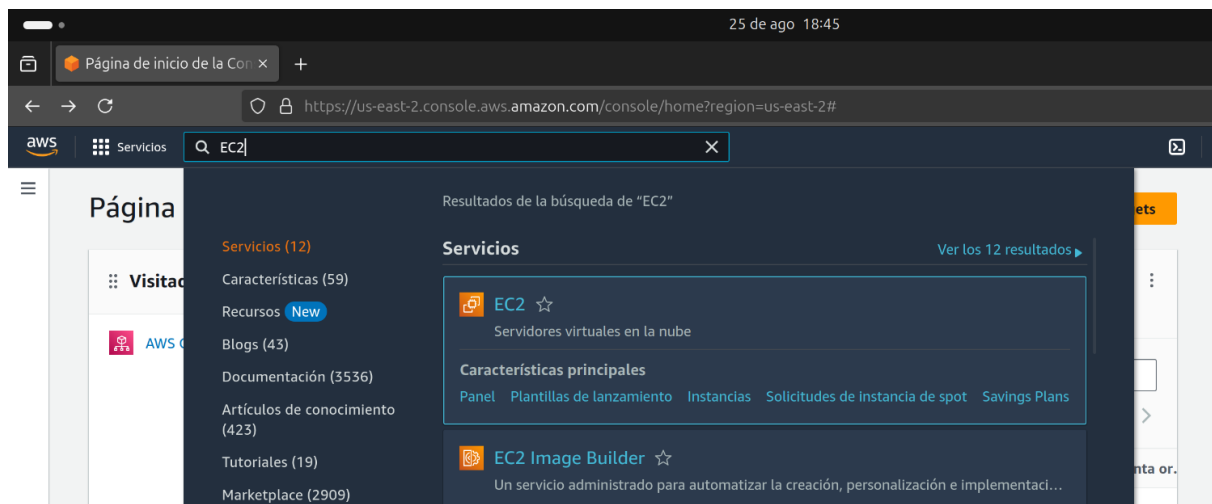


WebDev - Ayudantía 01

Crear Instancia EC2

En caso que no hayan podido crear su instancia durante las clases de cátedra, lo explicaremos brevemente acá. Una instancia EC2 de Amazon es básicamente una máquina virtual, es decir, un computador virtual que está corriendo en un computador físico en los servidores de Amazon, en dónde podemos ejecutar aplicaciones, almacenar datos, entre otros.

1. Buscar EC2 en el navegador, y abrir "Panel".



2. En el panel, apretar donde dice "Lanzar Instancia".

25 de ago 18:51

Inicio | EC2 | us-east-2 x Recibidos - nicolas.chirino x +

← → ↺ https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#Home:

aws Servicios 🔍 Buscar [Alt+S]

Panel de EC2 X

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

[Novedad](#)

▼ **Imágenes**

AMI

Catálogo de AMI

▼ **Elastic Block Store**

Volúmenes

Instantáneas

Administrador del ciclo de

Recursos [EC2 Global View](#) ⌵ ⚙️ ↺

Actualmente, utiliza los siguientes recursos de Amazon EC2 en la región EE.UU. Este (Ohio):

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

Lanzar la instancia

Para comenzar, lance una instancia de Amazon EC2, que es un servidor virtual en la nube.

[Lanzar la instancia](#) ▼

[Migrar un servidor](#) ⌵

Nota: Sus instancias se lanzarán en la región EE.UU. Este (Ohio)

Estado del servicio [Panel de AWS Health](#) ⌵ ↺

Región

EE.UU. Este (Ohio)

Estado

🟢 Este servicio funciona con normalidad.

Zonas

3. Poner el nombre que ustedes quieran a la instancia, y seleccionar una imagen de Ubuntu de 64 bits.

aws Servicios [Alt+S]

webdev [Agregar etiquetas adicionales](#)

▼ Imágenes de aplicaciones y sistemas operativos (Imagen de máquina de Amazon) [Información](#)

Una AMI es una plantilla que contiene la configuración de software (sistema operativo, servidor de aplicaciones y aplicaciones) necesaria para lanzar la instancia. Busque o examine las AMI si no ve lo que busca a continuación.

Inicio rápido

Amazon Linux
aws

macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Linux
SUS

Incluir de AMI de AWS, Marketplace y la comunidad

Imágenes de máquina de Amazon (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
 ami-085f9c64a9b75eed5 (64 bits (x86)) / ami-0b8e245d9e93c05fa (64 bits (Arm))
 Virtualización: hvm Activado para ENA: true Tipo de dispositivo raíz: ebs

Apto para la capa gratuita ▼

Descripción
 Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Arquitectura **ID de AMI** Proveedor verificado

4. Seleccionar la instancia más pequeña que puedan, en este caso seleccioné "t2.micro". Esto dado que una página web no necesita de muchos recursos.

aws Servicios [Alt+S]

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

Tipo de instancia

t2.micro
 Familia: t2 1 vCPU 1 GiB Memoria Generación actual: true
 Bajo demanda Linux base precios: 0.0116 USD por hora
 Bajo demanda SUSE base precios: 0.0116 USD por hora
 Bajo demanda Windows base precios: 0.0162 USD por hora
 Bajo demanda RHEL base precios: 0.026 USD por hora

Apto para la capa gratuita ▼

☐ Todas las generaciones

[Comparar tipos de instancias](#)

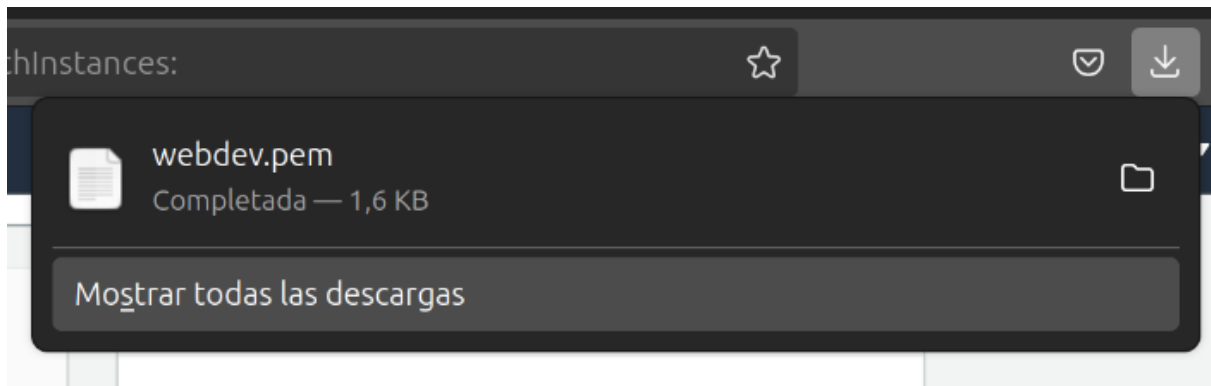
[Se aplican costos adicionales a las AMI con software preinstalado](#)

5. Generar un nuevo par de llaves, esto generará un archivo .PEM que deberemos guardar como hueso santo. Al apretar el botón les saldrá la siguiente interfaz, pueden poner el nombre que ustedes quieran.

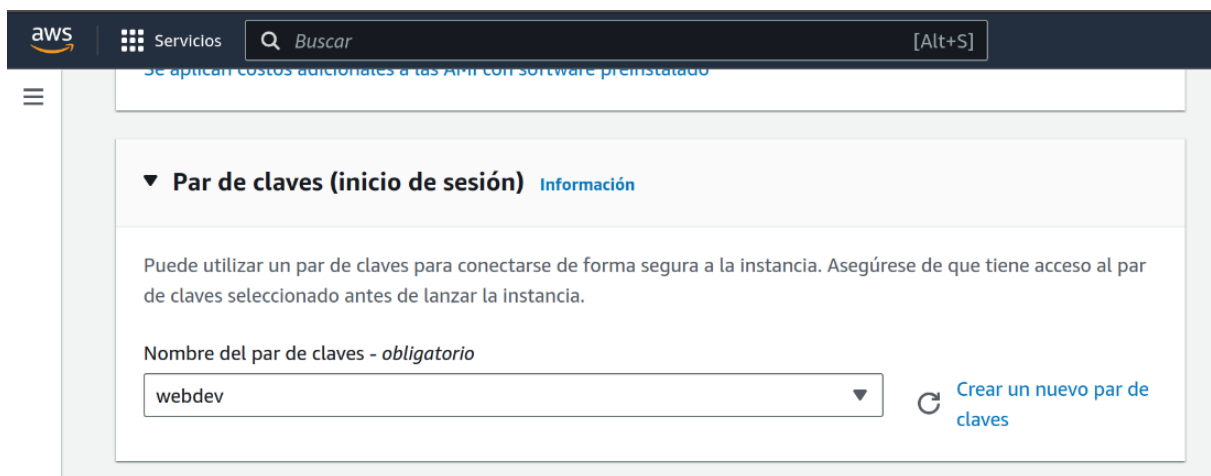
The screenshot shows a modal window titled "Crear par de claves" (Create key pair) with a close button (X) in the top right corner. The window contains the following sections:

- Nombre del par de claves** (Key pair name): A text input field containing "webdev". Below the field, a note states: "Con los pares de claves es posible conectarse a la instancia de forma segura." (With key pairs, it is possible to connect to the instance securely). Below the note, a warning says: "El nombre puede incluir hasta 255 caracteres ASCII. No puede incluir espacios al principio ni al final." (The name can include up to 255 ASCII characters. It cannot include spaces at the beginning or end).
- Tipo de par de claves** (Key pair type): Two radio button options.
 - ☒ **RSA**: Par de claves pública y privada cifradas mediante RSA (Public and private key pair encrypted using RSA).
 - ☐ **ED25519**: Par de claves privadas y públicas cifradas ED25519 (ED25519 encrypted private and public key pair).
- Formato de archivo de clave privada** (Private key file format): Two radio button options.
 - ☒ **.pem**: Para usar con OpenSSH (For use with OpenSSH).
 - ☐ **.ppk**: Para usar con PuTTY (For use with PuTTY).
- Warning box**: A yellow box with a warning icon containing the text: "Cuando se le solicite, almacene la clave privada en un lugar seguro y accesible del equipo. Lo necesitará más adelante para conectarse a la instancia. [Más información](#)" (When prompted, store the private key in a secure and accessible location on the device. You will need it later to connect to the instance. [More information](#)).
- Buttons**: "Cancelar" (Cancel) and "Crear par de claves" (Create key pair).

6. Guardamos el archivo .PEM en un lugar dónde no se nos vaya a olvidar.



7. Luego, se seleccionará automáticamente el par de claves como "webdev", que es el nombre que yo le puse al archivo .PEM.



8. Modificamos la configuración de red, y seleccionamos para permitir el tráfico SSH, HTTPS y HTTP.

aws

Servicios

Q Buscar

[Alt+S]

≡

▼ Configuraciones de red [Información](#)

Editar

Red [Información](#)

vpc-0262f63d7b0074c61

Subred [Información](#)

Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

Asignar automáticamente la IP pública [Información](#)

Habilitar

Se aplican cargos adicionales cuando no se cumplen los límites del nivel gratuito

Firewall (grupos de seguridad) [Información](#)

Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

☒ Crear grupo de seguridad

☐ Seleccionar un grupo de seguridad existente

Crearemos un nuevo grupo de seguridad denominado "launch-wizard-1" con las siguientes reglas:

☒ Permitir el tráfico de SSH desde

Ayuda a establecer conexión con la instancia

Cualquier lugar
0.0.0.0/0

☒ Permitir el tráfico de HTTPS desde Internet

Para configurar un punto de enlace, por ejemplo, al crear un servidor web

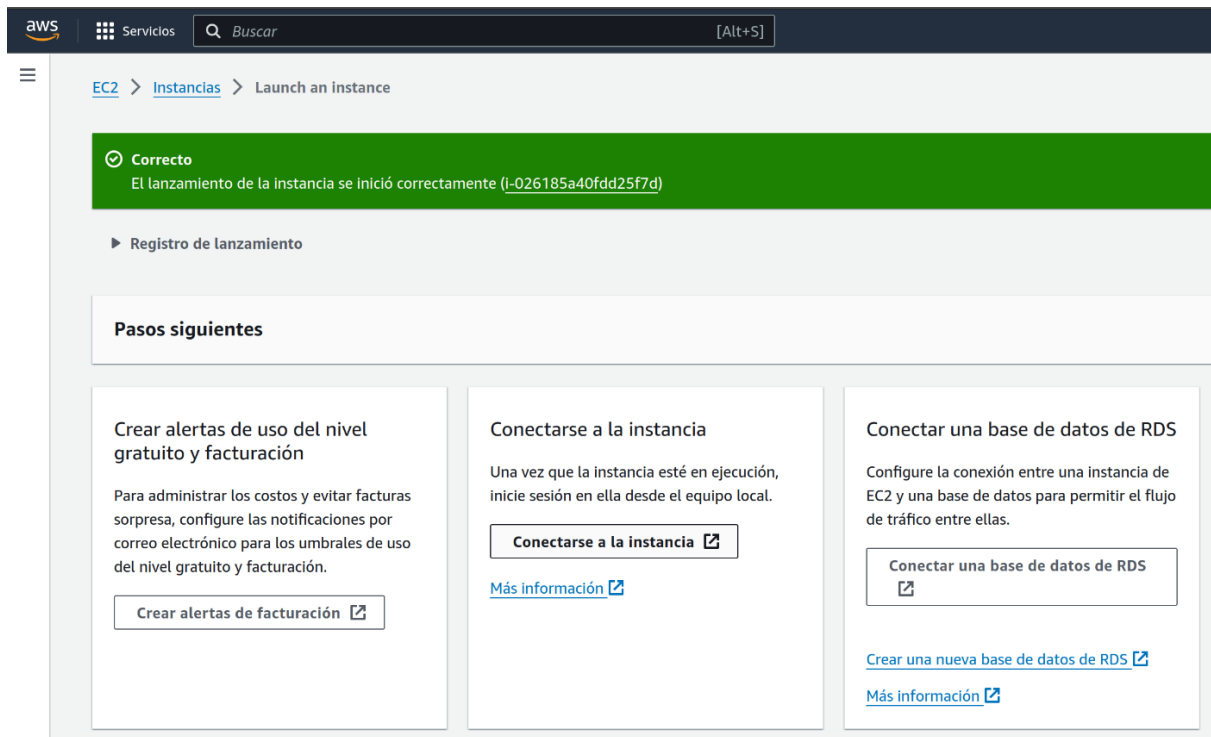
☒ Permitir el tráfico de HTTP desde Internet

Para configurar un punto de enlace, por ejemplo, al crear un servidor web

⚠ Las reglas con origen 0.0.0.0/0 permiten que todas las direcciones IP tengan acceso a la instancia. Le recomendamos que configure las reglas del grupo de seguridad para permitir el acceso únicamente desde direcciones IP conocidas.

✕

9. Una vez listo, creamos la instancia en el apartado derecho. Esto nos mostrará la siguiente página, donde apretaremos "Conectarse a una instancia".



10. Para conectarnos a la instancia, apretaremos "Cliente SSH", si tienen acceso a una terminal desde el computador. Acá debemos abrir una terminal dentro de la carpeta en dónde hayamos guardado el archivo .PEM, copiar el comando de ejemplo, y lo pegamos en la terminal.

aws Servicios [Alt+S]

EC2 > Instancias > i-026185a40fdd25f7d > Conectarse a la instancia

Conectarse a la instancia [Información](#)

Conéctese a la instancia i-026185a40fdd25f7d (webdev) mediante cualquiera de estas opciones

Conexión de la instancia EC2

Administrador de sesiones

Cliente SSH

Consola de serie de EC2

ID de la instancia
i-026185a40fdd25f7d (webdev)

1. Abra un cliente SSH.
2. Localice el archivo de clave privada. La clave utilizada para lanzar esta instancia es webdev.pem
3. Ejecute este comando, si es necesario, para garantizar que la clave no se pueda ver públicamente.
chmod 400 "webdev.pem"
4. Conéctese a la instancia mediante su DNS público:
ec2-3-143-142-134.us-east-2.compute.amazonaws.com

Ejemplo:
ssh -i "webdev.pem" ubuntu@ec2-3-143-142-134.us-east-2.compute.amazonaws.com

Nota: En la mayoría de los casos, el nombre de usuario adivinado es correcto. Sin embargo, lea las instrucciones de uso de la AMI para comprobar si el propietario de la AMI ha cambiado el nombre de usuario predeterminado de la AMI.

Y listo! ya estamos conectados a nuestra instancia EC2 mediante SSH. Acá podemos ejecutar comandos y configurar nuestra instancia para levantar el servidor.

Levantar servidor web NGINX

Nuestra instancia ya está en la web, pero si accedemos desde nuestro navegador no nos mostrará nada. Para levantar un servidor web y mostrar páginas al mundo, lo haremos utilizando NGINX. 1.

1. Para esto, ejecutaremos los siguientes comandos en la terminal.

```
sudo apt update  
sudo apt install nginx -y
```

Esto actualiza los programas y paquetes de nuestra instancia, y luego instala el paquete "NGINX", el cuál será nuestro servidor web.

2. Una vez instalado, tendremos que ejecutar este servicio, utilizando los comandos:

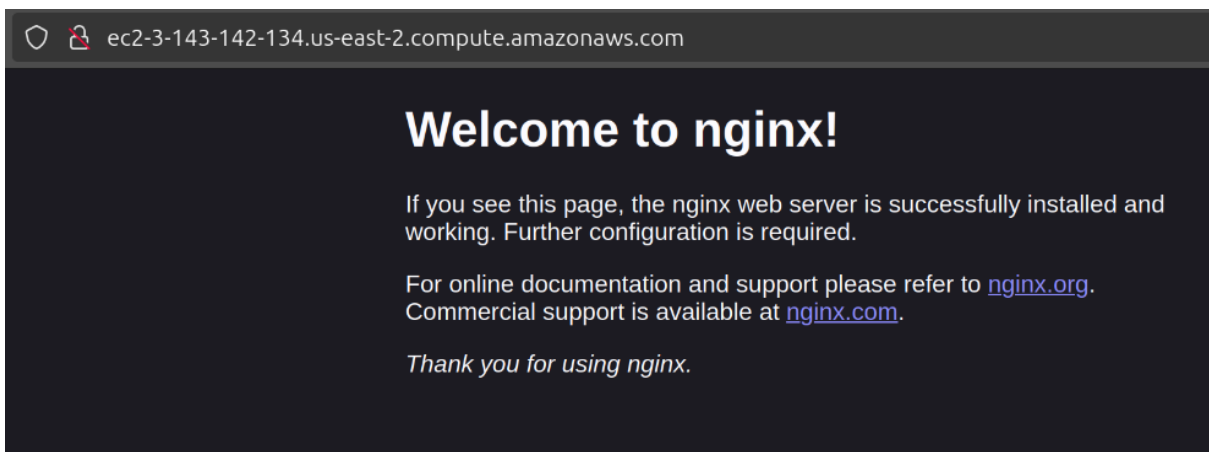
```
sudo systemctl start nginx
sudo systemctl enable nginx
```

```
ubuntu@ip-172-31-29-152:~$ sudo systemctl start nginx
ubuntu@ip-172-31-29-152:~$ sudo systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /usr/lib/sy
emd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable nginx
ubuntu@ip-172-31-29-152:~$
```

3. Por último, modificaremos los permisos de la carpeta en la cuál se alojarán nuestros archivos de la página web, de forma que sólo nosotros podamos modificarlos.

```
sudo chown -R $USER:$USER /var/www/
sudo chmod -R 755 /var/www
```

Una vez listo, estará listo nuestro servidor web, y podremos acceder a nuestra instancia desde el navegador. Para esto iremos al panel de la instancia en Amazon AWS, y apretaremos donde dice dirección abierta. **Esto no nos mostrará nada, porque debemos sacarle la "S" de HTTPS al link.** Si todo salió bien, les mostrará la siguiente página.

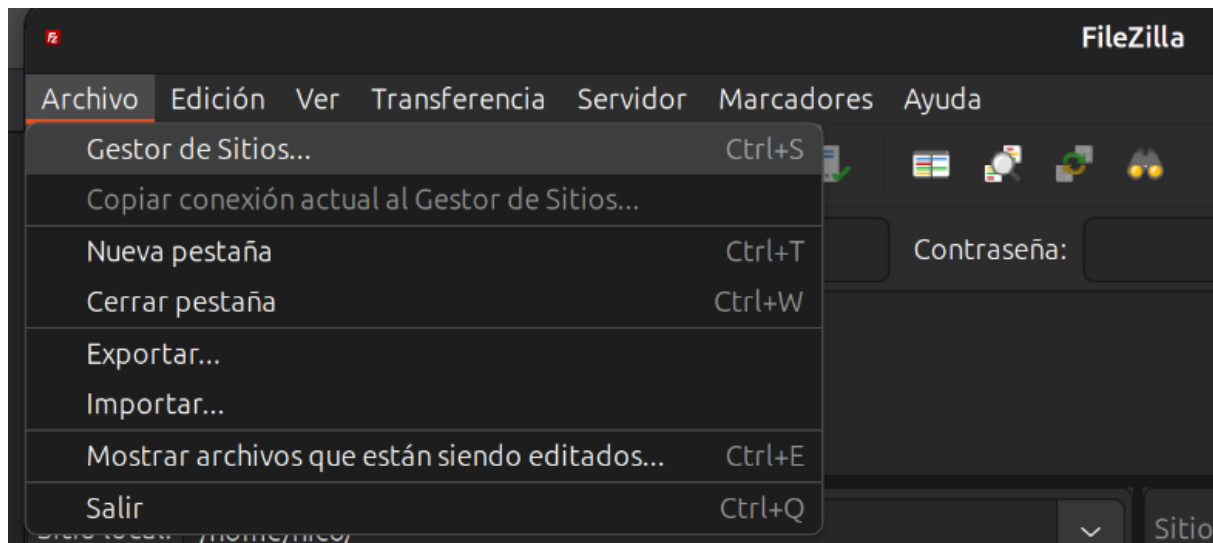


Ahora, sólo nos falta mostrar los archivos correspondientes a nuestra página.

Subir archivos con FileZilla

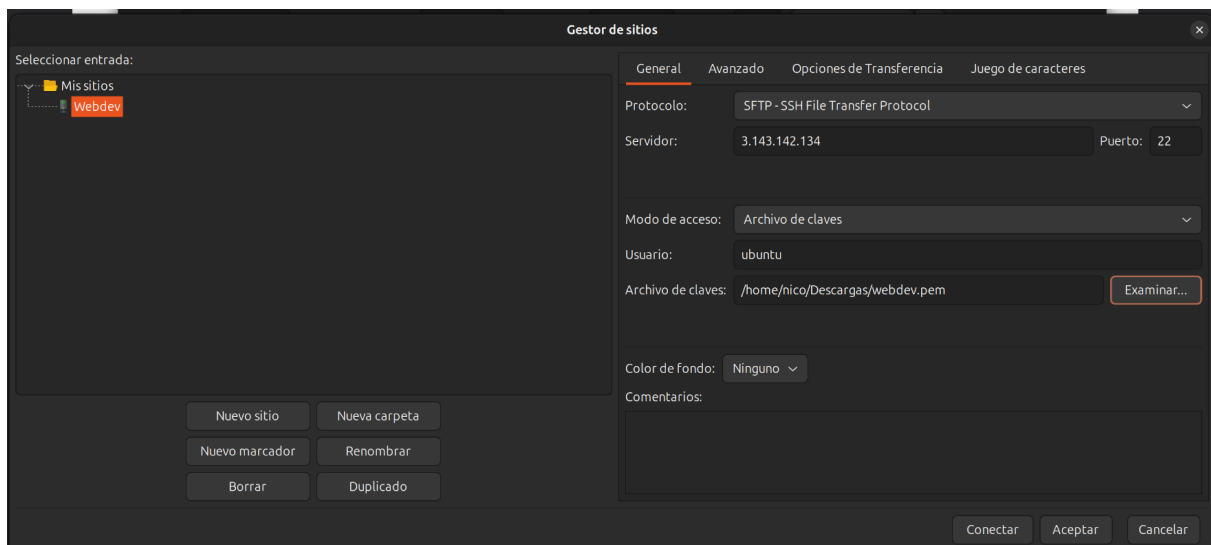
FileZilla es un cliente SFTP para poder enviar archivos de forma segura y visual mediante SSH, utilizando una interfaz gráfica.

1. Primero nos conectaremos a la instancia, mediante el gestor de sitios.

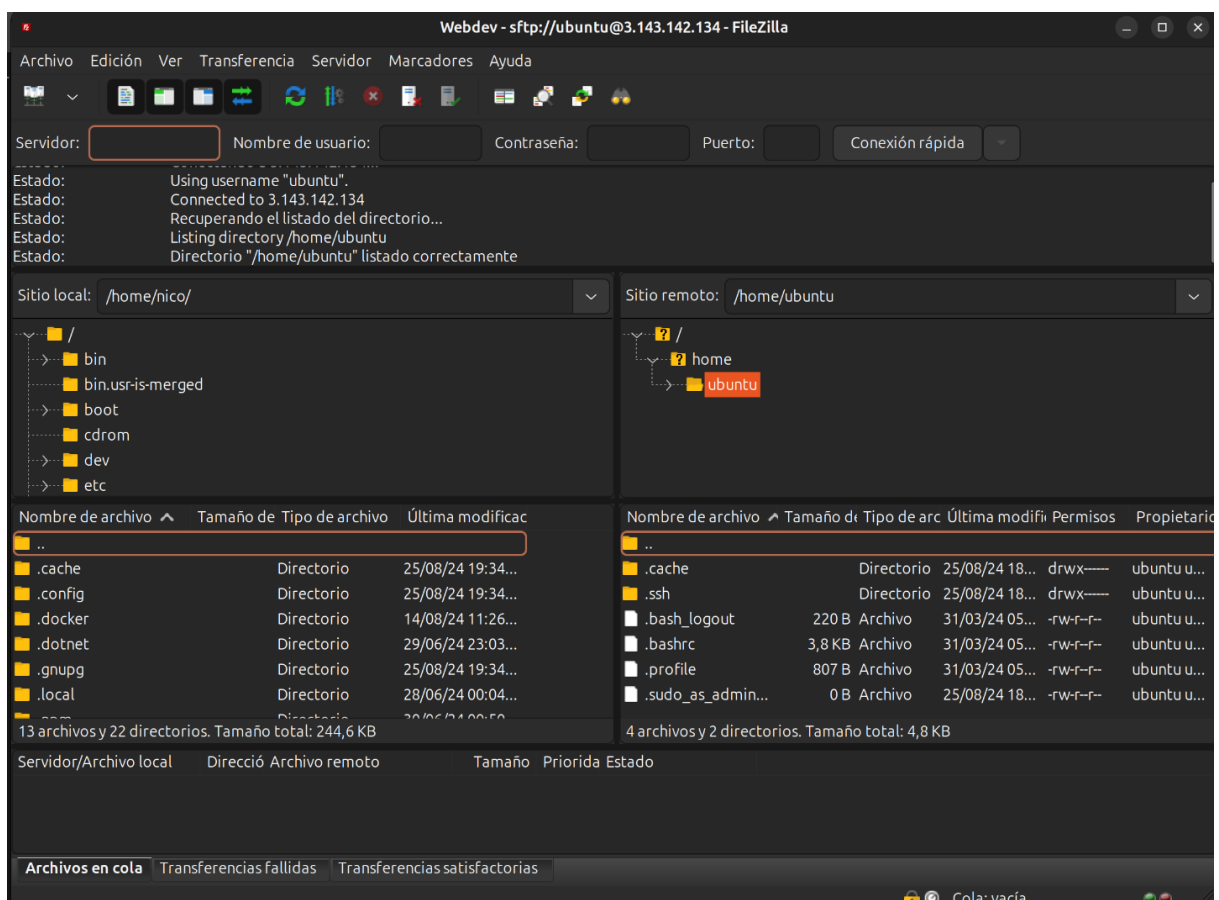


2. Acá, seleccionaremos "Nuevo Sitio", le pondremos el nombre que queramos y llenamos los campos de la derecha.
 - a. Protocolo: SFTP
 - b. Servidor: IPv4 pública de la instancia (Se obtiene desde el panel de AWS)
 - c. Puerto: 22
 - d. Modo de acceso: Archivo de claves
 - e. Usuario: ubuntu
 - f. Archivo de claves: Archivo .PEM

Y le damos a Conectar y Aceptar.

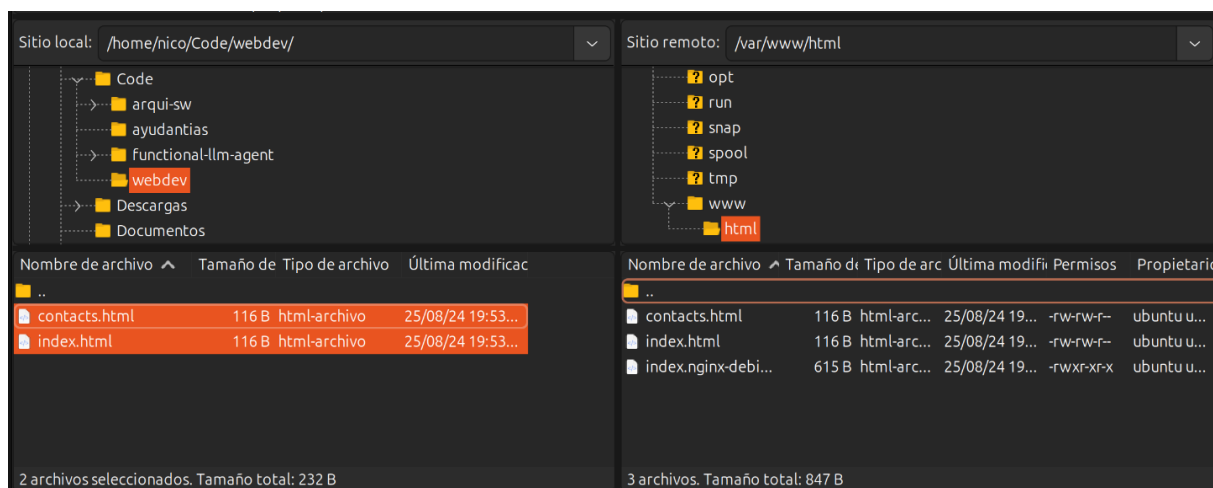


3. No es necesario guardar una contraseña maestra. Si todo salió bien, nos aparecerá la siguiente interfaz:

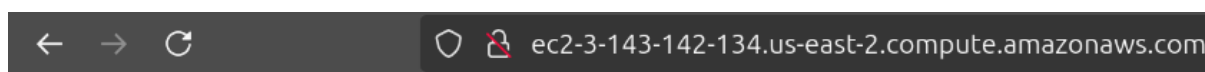


4. Luego, buscamos el directorio en dónde se hospedan nuestros archivos del servidor web. Para esto, dentro de "Sitio remoto" buscamos

"/var/www/html", y moveremos nuestros archivos .html aquí, seleccionando los archivos y arrastrándolos.



5. Si todo salió bien, podemos refrescar la página correspondiente al servidor web desde el navegador, y aparecerá nuestro nuevo index.html



Hello, World!

Si se dan cuenta, yo subí dos archivos a la instancia, "index.html" y "contacts.html", pero si voy a la página "ipinstancia/contacts" no me aparece nada, entonces, ¿Cómo arreglo esto?



404 Not Found

nginx/1.24.0 (Ubuntu)

Routing de archivos estáticos

El error anterior ocurrió ya que no pusimos .html al final de la ruta, es decir, debió ser /contacts.html. Pero esto se ve feo y además, no es lo que se hace en el día a día. Entonces, para poder hacer que nuestro servidor muestre la página HTML, sin poner directamente .HTML en la URL, tendremos que modificar un poco nuestro servidor web de NGINX.

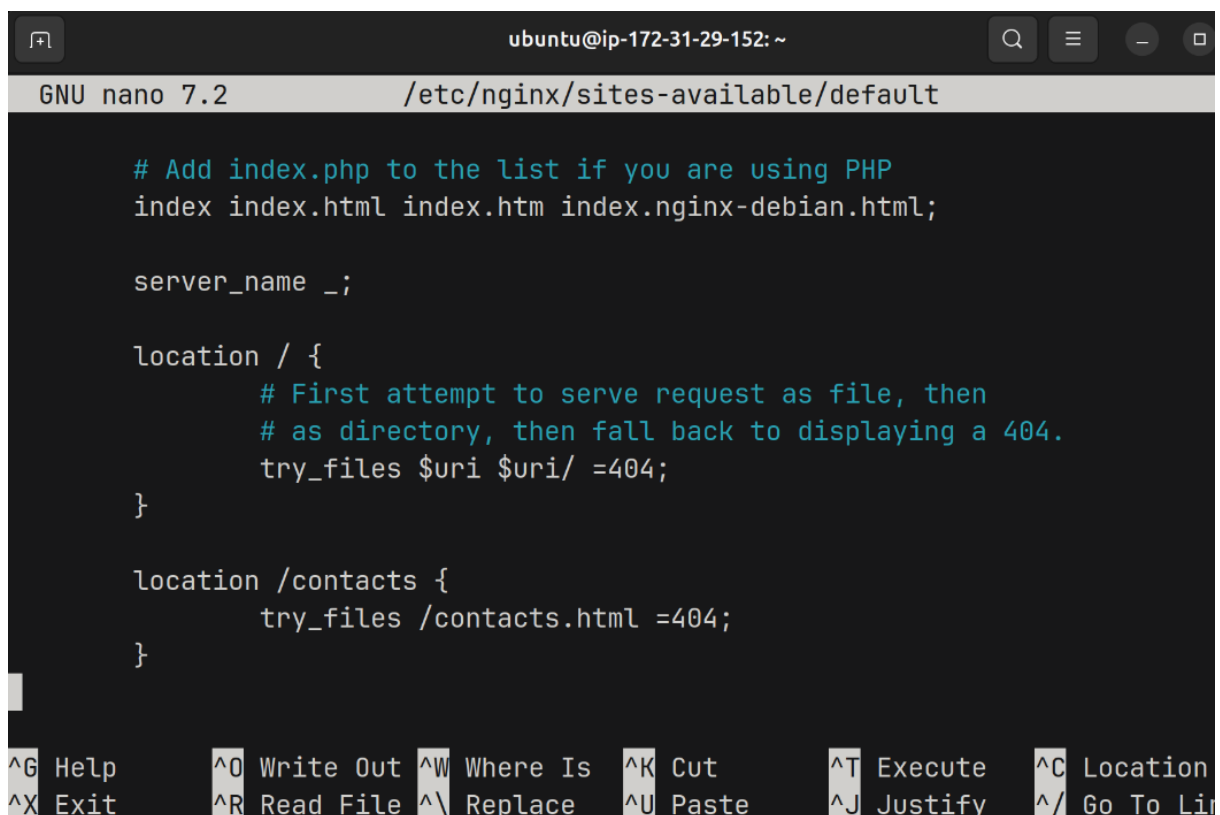
1. Dentro de la terminal la cuál está conectada a nuestra instancia por SSH, ejecutaremos el siguiente comando:

```
sudo nano /etc/nginx/sites-available/default
```

2. Esto nos abrirá el archivo de configuración de NGINX, utilizando el editor de texto "nano". Bajaremos hasta donde dice "location" y agregaremos lo siguiente:

```
location /contacts {  
    try_files /contacts.html =404;  
}
```

3. Debería quedar de la siguiente forma:



The screenshot shows a terminal window with the title 'ubuntu@ip-172-31-29-152: ~'. The terminal is running the 'nano' text editor, editing the file '/etc/nginx/sites-available/default'. The editor's status bar at the top shows 'GNU nano 7.2'. The configuration file content is as follows:

```
# Add index.php to the list if you are using PHP  
index index.html index.htm index.nginx-debian.html;  
  
server_name _;  
  
location / {  
    # First attempt to serve request as file, then  
    # as directory, then fall back to displaying a 404.  
    try_files $uri $uri/ =404;  
}  
  
location /contacts {  
    try_files /contacts.html =404;  
}
```

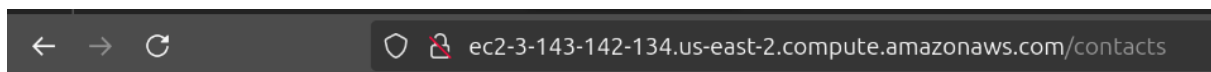
At the bottom of the terminal, there is a navigation bar with various keyboard shortcuts for the nano editor:

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location
^X Exit	^R Read File	^_\ Replace	^U Paste	^J Justify	^/ Go To Lin

4. Finalmente, guardamos los cambios en el archivo utilizando "CTRL + O", apretamos "ENTER" y finalmente salimos utilizando "CTRL + X". Luego, para que hagan efecto los cambios, reiniciamos el servidor web usando el comando:

```
sudo systemctl restart nginx
```

Sí todo salió bien, ahora podemos refrescar la página en la URL `"/contacts"` y nos mostrará el archivo respectivo para `contacts.html`.



Contacts