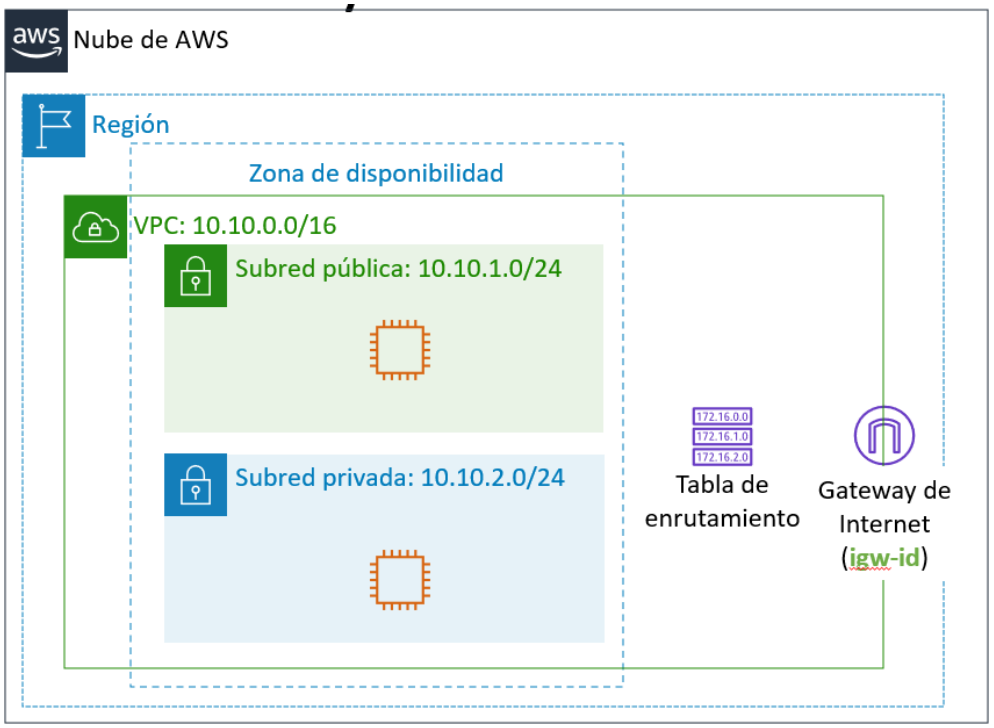
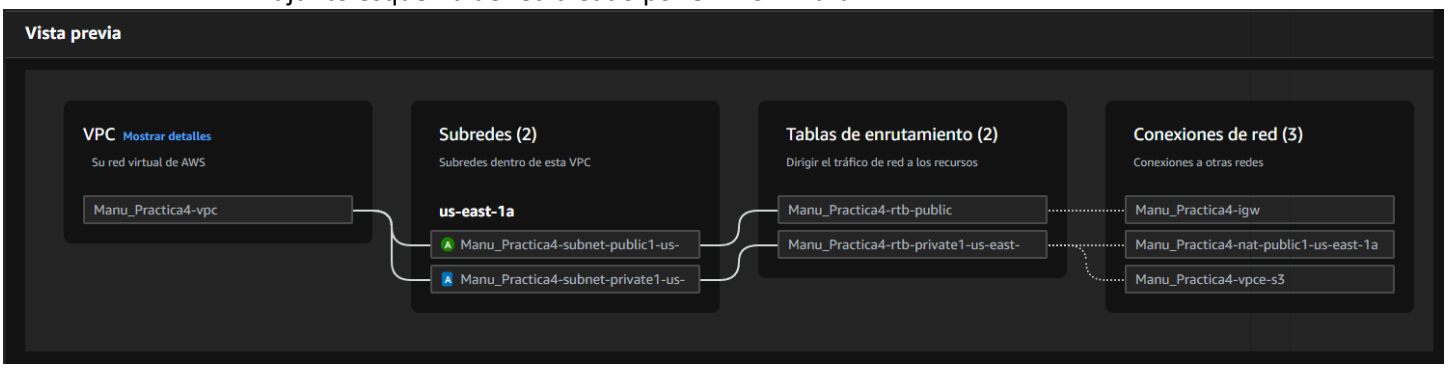
UD.1 Practica 4. VPC y servidor Libre

1. Crear VPC

- Creamos VPC con nuestro nombre. En us-east-1 Usamos CIDR IPv4, no el IPv6. Rango de direcciones privadas: 10.10.0.0/16.
- Creamos 2 subredes (1 privada y 1 públicas) en una única AZ, con el asistente. Nombre_privada1 ; Nombre_publica1. Debemos tener 1 NAT gateway en la AZ



Adjunto esquema de red creado por el VPC Wizard



Creo la VPC con rango 10.10.0.0/16. Activo la opción de DNS en acciones, que me vendrá bien para luego configurar los servicios.

Compruebo los rangos y asigno la mitad de las direcciones a cada sibred

Name	ID de subred	Estado	VPC	CIDR IPv4
Manu_Practica4-subnet-public1-us-east...	subnet-0238075e2fbc623db	Available.	vpc-0a7e50bd1b378b118 | Ma...	10.10.0.0/20
Manu_Practica4-subnet-private1-us-eas...	subnet-0894d6ddfb670ce0f	Available.	vpc-0a7e50bd1b378b118 | Ma...	10.10.128.0/20

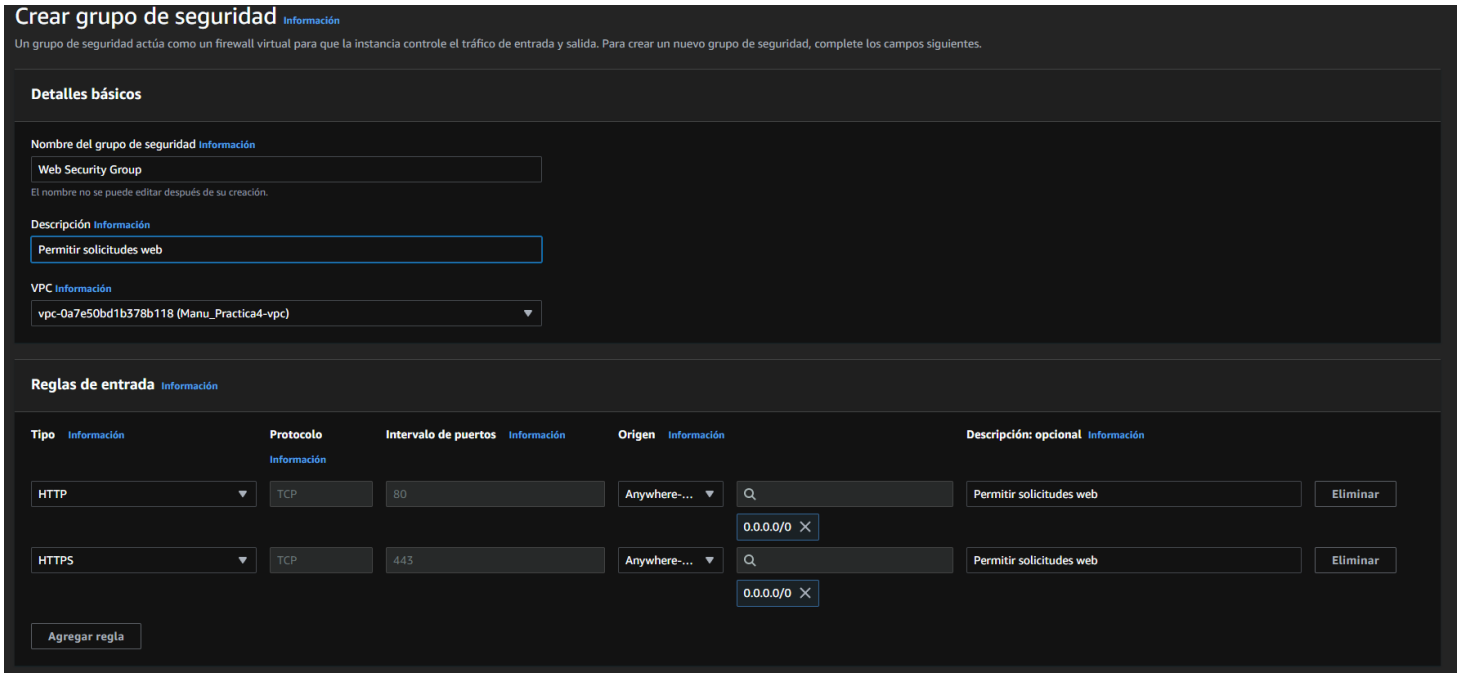
- De momento no creo un internet Gateway NAT pero lo crearle manualmente después)

  Doy a crear la VPC

2. Crear el grupo de seguridad para internet

- Creamos un grupo de seguridad "Web Security Group" que de a acceso HTTP y HTTPS

  Reglas de entrada:



Source (Origen): Anywhere (Cualquiera)

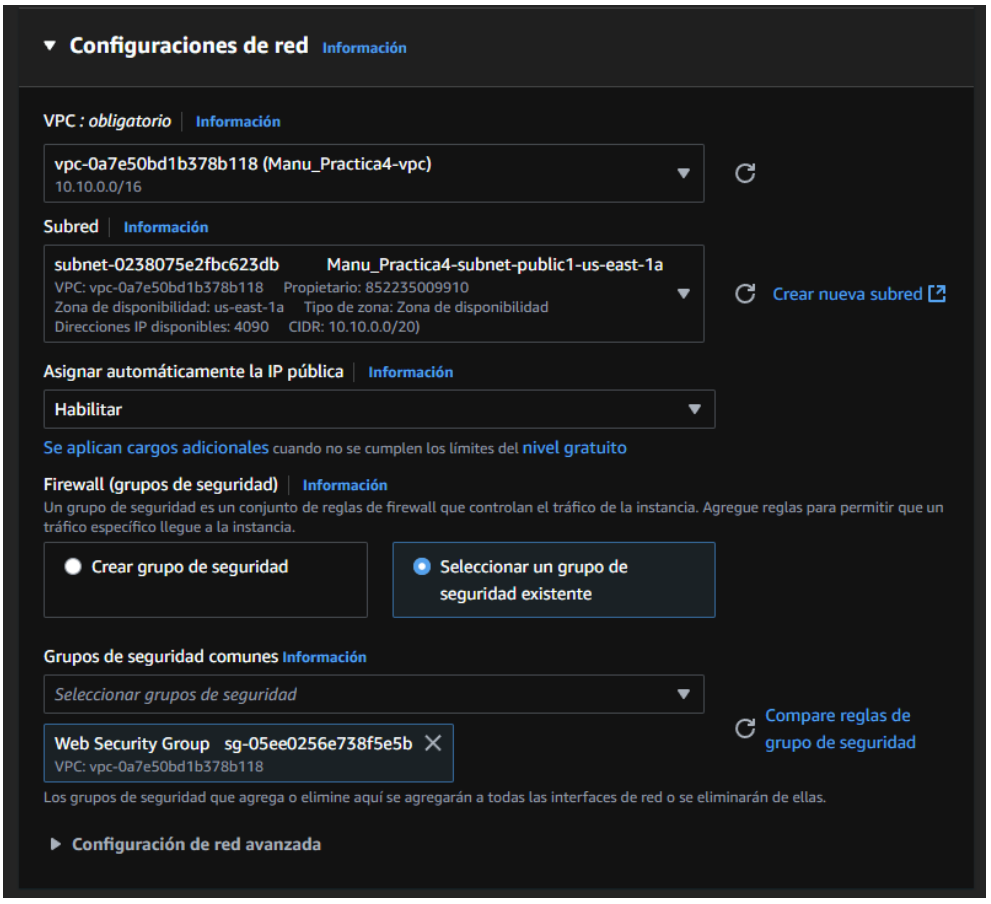
Description (Descripción): Permitir solicitudes web

3. Creamos instancia EC2 en la subred pública. Lanzada desde la consola

- Nombre: Tunombre_Server_I_P4

  Ej: Ruth_Server_I_P4

- tipo t2.micro
- AMI: mira en el catálogo de la comunidad, y en modelo de precios, filtra por “free”
- Para esta ocasión finalmente vamos a lanzar un Ubuntu Server
- Red y Subred: asegúrate de que está en la red/sub red adecuadas
- Auto-assign Public IP (Asignar automáticamente IP pública): Enable
- Selecccciona el grupo de Web Security Group para lanzarlo
- Emplea las claves de vockey para acceder al servidor



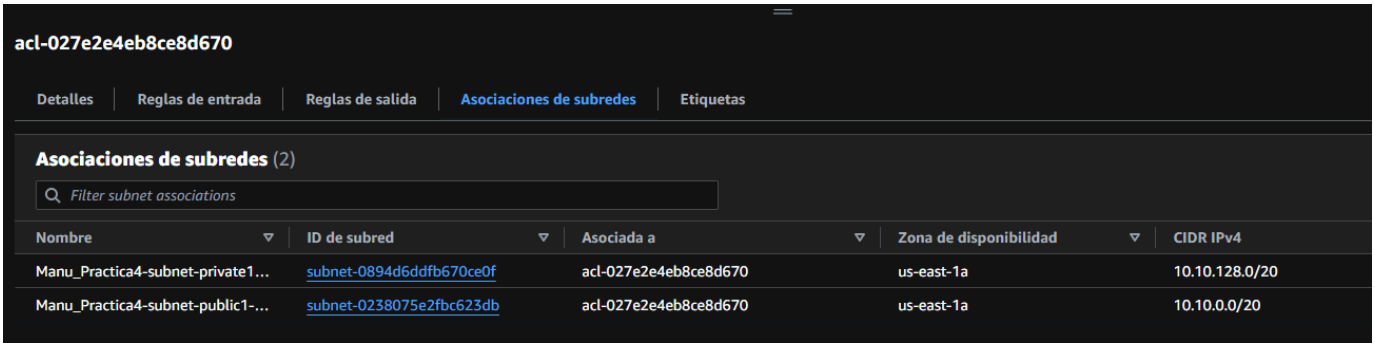
- En detalles avanzados pegamos lo siguiente en los datos de Usuario. (Yo he cogido Script para instalar Nginx):

```
#!/bin/bash
yum update -y
yum install -y nginx
systemctl start nginx
```

```
systemctl enable nginx
echo "<h1>¡Bienvenido a mi servidor Nginx!</h1>" > /usr/share/nginx/html/index.html
```

4. Probar que el servidor funciona, probar ACLs y conectividades

Por defecto al crear la red se ha creado una ACL asociada a nuestras dos subredes. Pero nota: una ACL puede reutilizarse para muchas subredes.



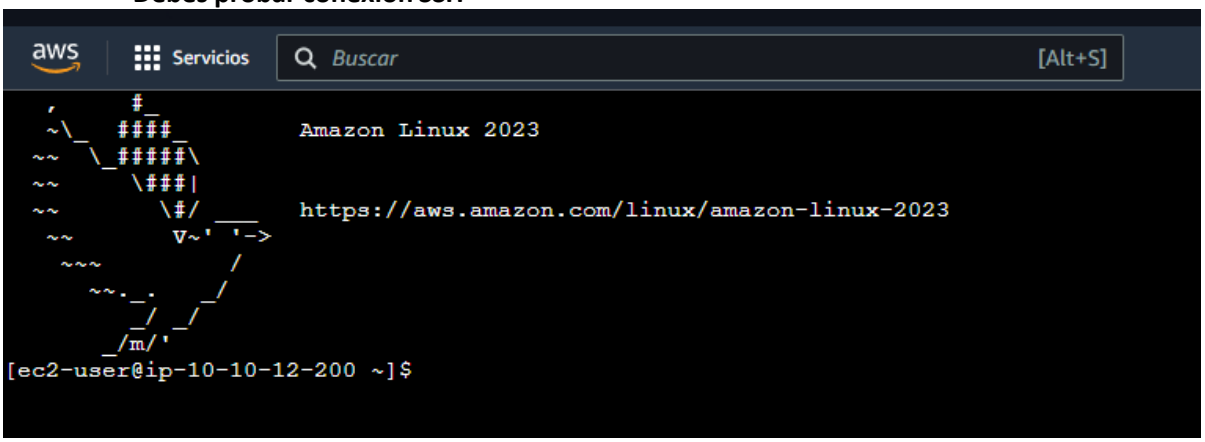
Ahora que hemos lanzado la instancia nos aplica:

- ACL de la red
- Grupo de seguridad que hemos elegido

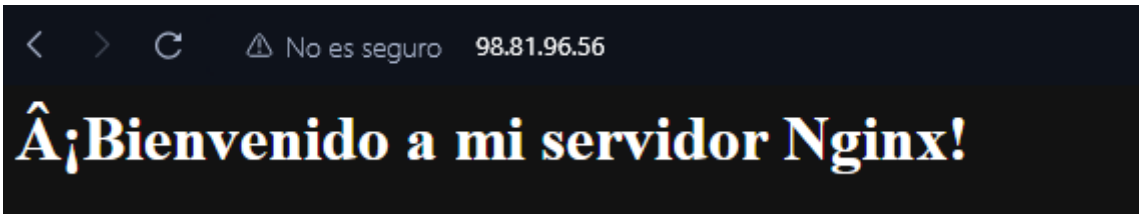
Probablemente tengas algún problema … averigua cual es. Comprueba que la ACL y el web security group no entran en conflicto.

¿Qué problema ha sido y como lo has arreglado (si alguno). Se debía a que la ACL tiene permitido todos los tipos de conexiones, y la Web Security Group no.

Debes probar conexión SSH



También debes probar conexión HTTP



Consulta los metadatos de la instancia

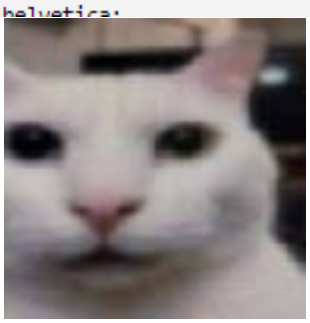
Los metadatos de la instancia son datos sobre la instancia. Mientras esté conectado a la instancia los puedes ver así:

En un navegador: **http:// 44.223.66.67/latest/meta-data/**

En una ventana de terminal: **curl http://44.223.66.67/latest/meta-data/**

```
eee_w_3595259@runweb138769:~$
eee_w_3595259@runweb138769:~$ curl http://98.81.96.56/latest/meta-data/
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
    <head>
        <title>The page is not found</title>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
        <style type="text/css">
            /*<![CDATA[*/
            body {
                background-color: #fff;
                color: #000;
                font-size: 0.9em;
                font-family: sans-serif,helvetica;
```

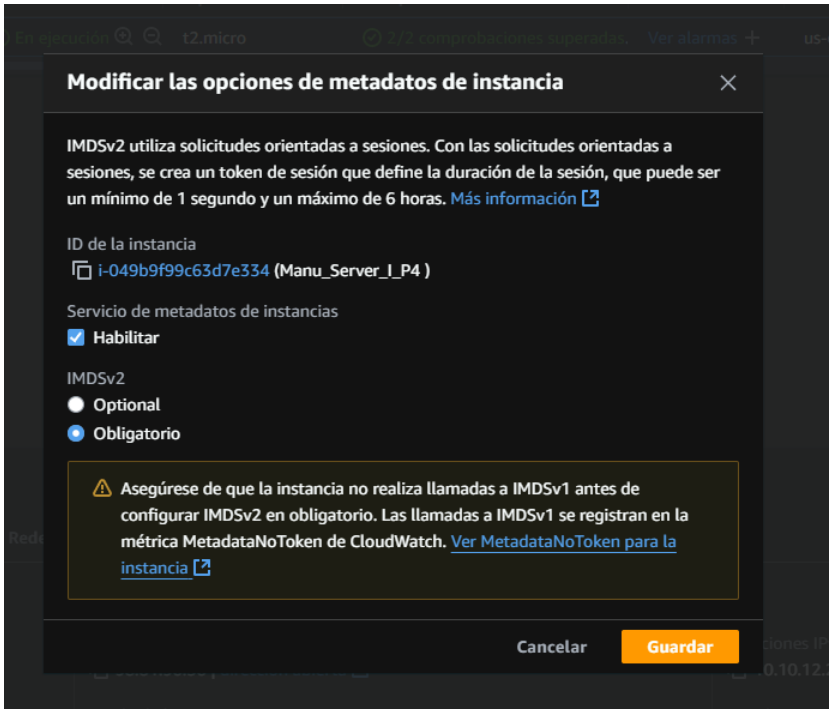


Eh?

Revisa las opciones metadatos de una instancia existente mediante la consola

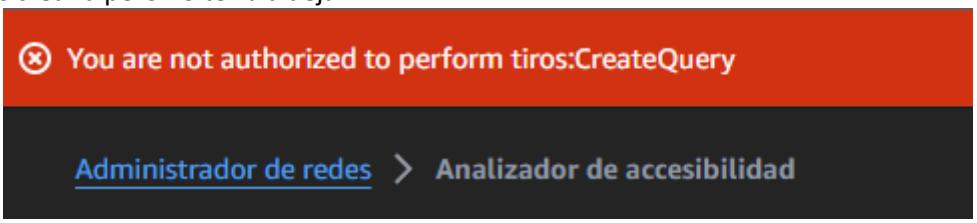
En la consola de Amazon EC2, selecciona la instancia-> Elija **Acciones**, **Configuración de la instancia** y **Modificar opciones de metadatos de instancia**.

Revise las opciones de metadatos de la instancia actuales en el cuadro de diálogo **Modificar las opciones de metadatos de la instancia**.



5. Intenta usar la herramienta Reachabily analyzer para ver que todo está bien

Puedes crearla pero no te va a dejar…

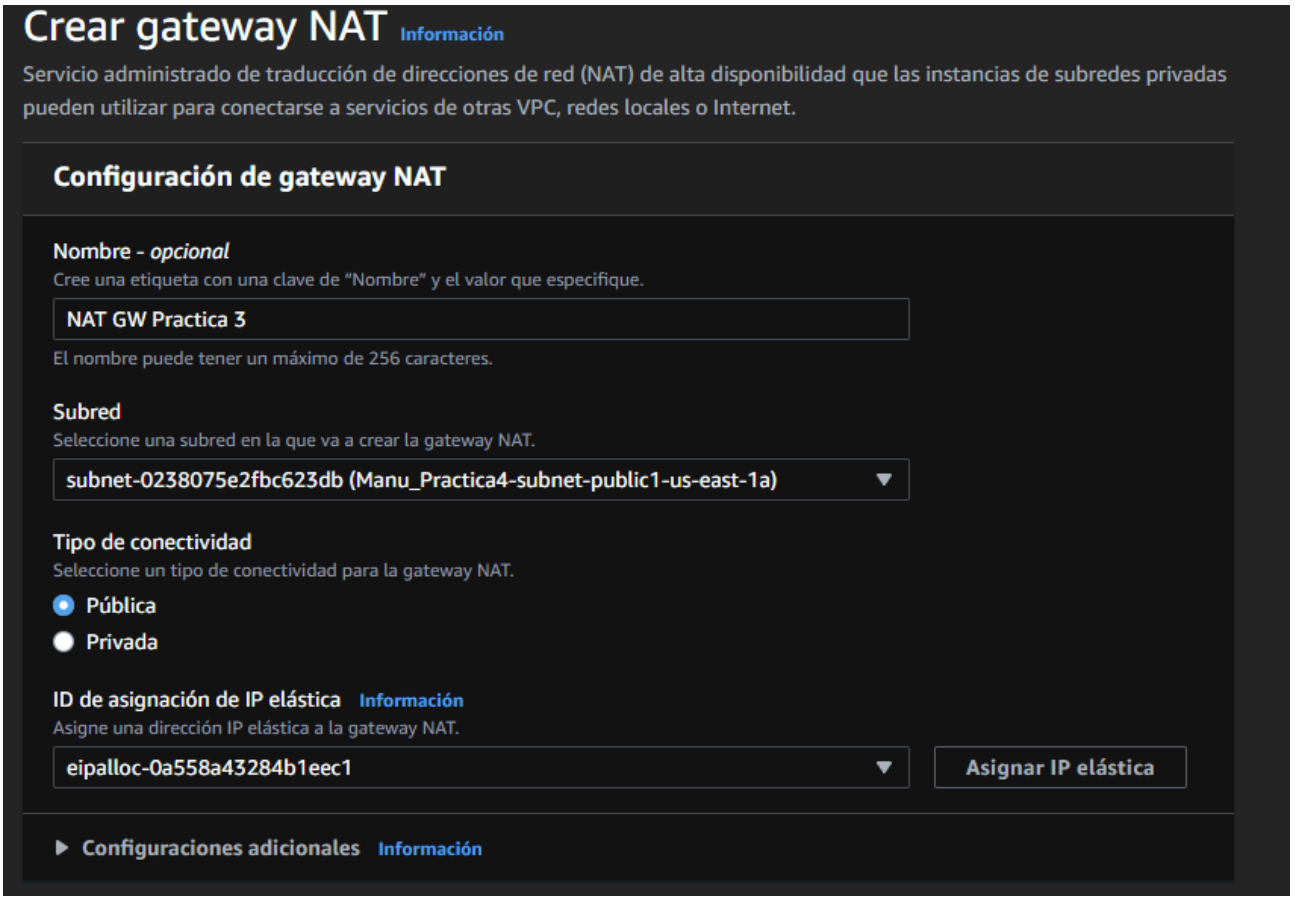


6. Gateway NAT

Si lo que queremos es salida a internet pero no entrada… necesitamos un NAT Gateway. Así que lo creamos. Al crearlo nos pone las subredes disponibles a las que podemos asociarlo.

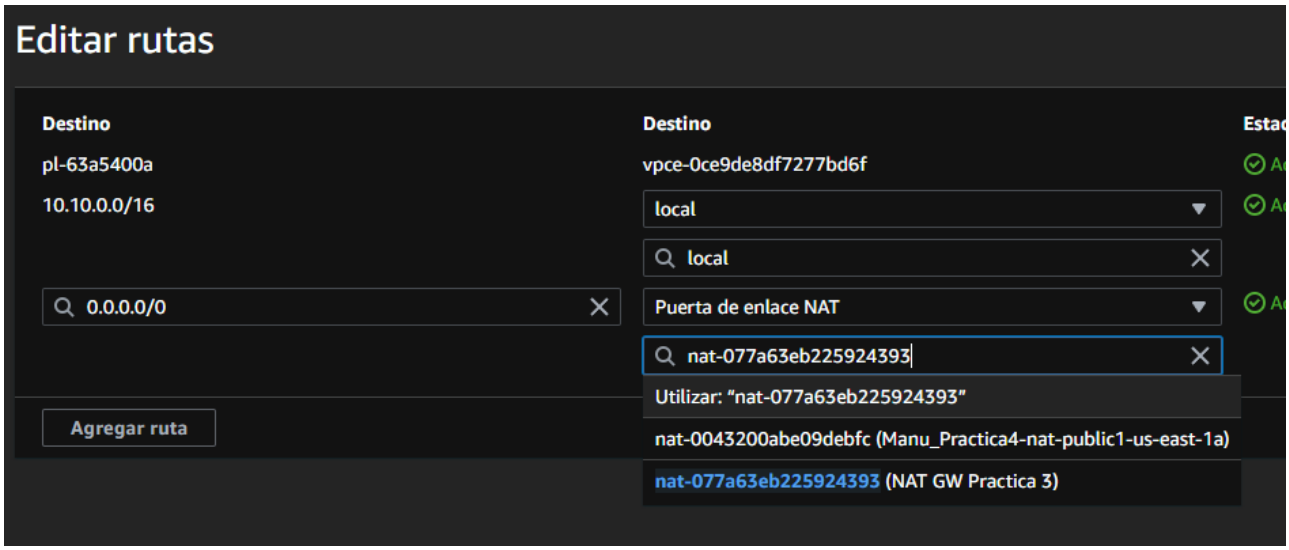
Muy importante: UN NAT Gateway debe asociarse siempre a una subred publica. Un NAT Gateway necesita salir a internet desde una subred publica, porque solo la subred publica está conectada a internet.

Lo asociamos a la subred publica. En conectividad ponemos publica y luego le asociamos una Elastic IP. Si no tenemos una, la generamos. Una Elastic IP es una IP que nos da AWS para conectarnos a internet, sin esto no tenemos conexión a internet.



Ya tenemos nuestro NAT Gateway disponible. Ahora tenemos que hacer que nuestra red privada acceda a internet, pero que desde internet no se pueda acceder a ella. Editamos la ruta de la tabla de enrutamiento privado. Tenemos que añadir la ruta para el caso de internet:

Todo lo que vaya a internet tiene que ira al NAT Gateway. Seleccionamos el NAT Gateway que acabamos de crear (Y que además se encuentra en la subred publica 1):



Ahora, de la privada si es de la VPC permanece interno, y si es de internet, va al NAT Gateway que se encuentra en una subred publica. Desde internet no pueden ser accedidos. Prueba el servicio con la IP elástica. Para ello, sal del laboratorio y vuelve a entrar, verás que la IP es la misma.

7. Deshabilitar Gateway NAT

Ojo! Los NAT Gateway cuestan 0,048 por hora=1,15 dolares al dia … unos 30 euros al mes. Asegurate de que no tienes ninguna IP elástica contratada y de que no hay ningún Gatway NAT