PRACTICA 4

COMPUTACIÓN
EN LA NUBE

Contenido

INTRODUCCIÓN

A lo largo de esta práctica se va a tener una primera toma de contacto con el sistema Docker, para la creación y administración de contenedores, se trabajará con el sistema de repositorios y a su vez se utilizarán los clústeres EC2 y Fargate de AWS.

ACTIVIDADES

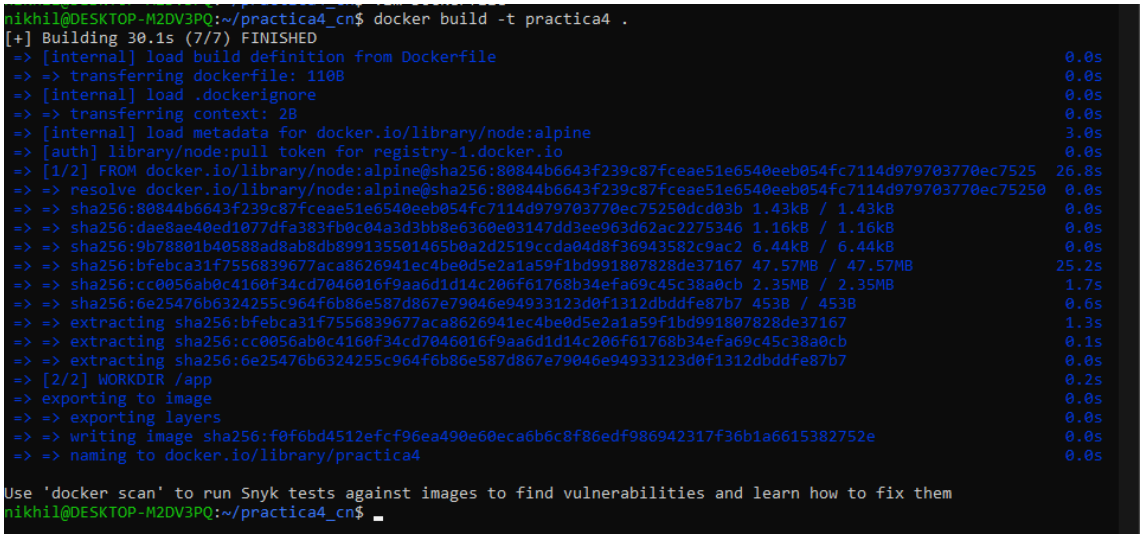
1. Crear un contenedor con docker que tenga una aplicación que permita comprobar su funcionamiento (e.g. una pagina web)

En primer lugar se crean ficheros en local para poder lanzar un servidor web donde aparecerá el contenido “Practica 4 CN”.

Se debe configurar el dockerfile de la siguiente manera:

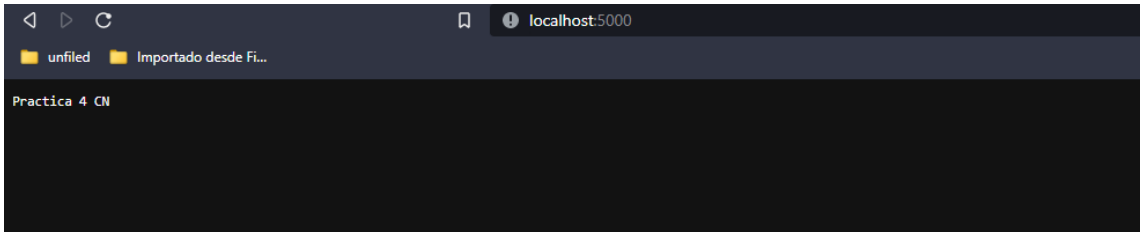
Para crear la imagen que se va a utilizar para el contenedor se debe ejecutar el siguiente comando:

docker build -t practica4



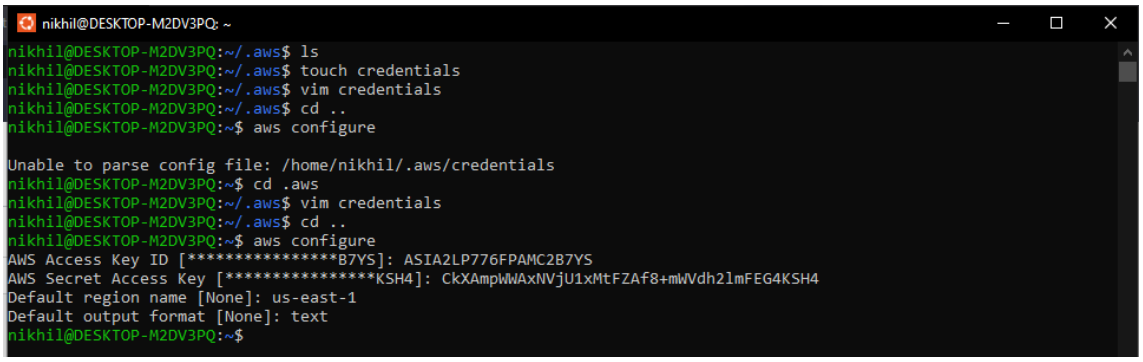
```
nikhil@DESKTOP-M2DV3PQ:~$ vim index.js
nikhil@DESKTOP-M2DV3PQ:~$ vim Dockerfile
nikhil@DESKTOP-M2DV3PQ:~$ docker build -t practica4 .
[+] Building 1.2s (7/7) FINISHED
 => [internal] load build definition from Dockerfile                                                0.0s
 => => transferring dockerfile: 37B                                                                 0.0s
 => [internal] load .dockerignore                                                                   0.0s
 => => transferring context: 2B                                                                     0.0s
 => [internal] load metadata for docker.io/library/node:alpine                                      1.0s
 => [internal] load build context                                                                   0.0s
 => => transferring context: 346B                                                                   0.0s
 => CACHED [1/2] FROM docker.io/library/node:alpine@sha256:80844b6643f239c87fceae51e6540eeb054fc7114d979703770  0.0s
 => [2/2] COPY index.js /index.js                                                                   0.0s
 => exporting to image                                                                              0.0s
 => => exporting layers                                                                             0.0s
 => => writing image sha256:dba89e2bf0e966905dcdc2f1fb75848bbbe9e2ada7bf57cb2836716e90bdbb5e        0.0s
 => => naming to docker.io/library/practica4                                                        0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
nikhil@DESKTOP-M2DV3PQ:~$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED         SIZE
practica4    latest    dba89e2bf0e9   9 seconds ago   171MB
nikhil@DESKTOP-M2DV3PQ:~$ docker run -p 5000:5000 dba89e2bf0e9
El servidor se esta ejecutando
```

Una vez creada la imagen, se debe ejecutar **docker run -p 5000:5000 número_id_imagen** para poder crear el contenedor. Si se ha creado y ejecutado correctamente sale el mensaje programado “El servidor se está ejecutando”

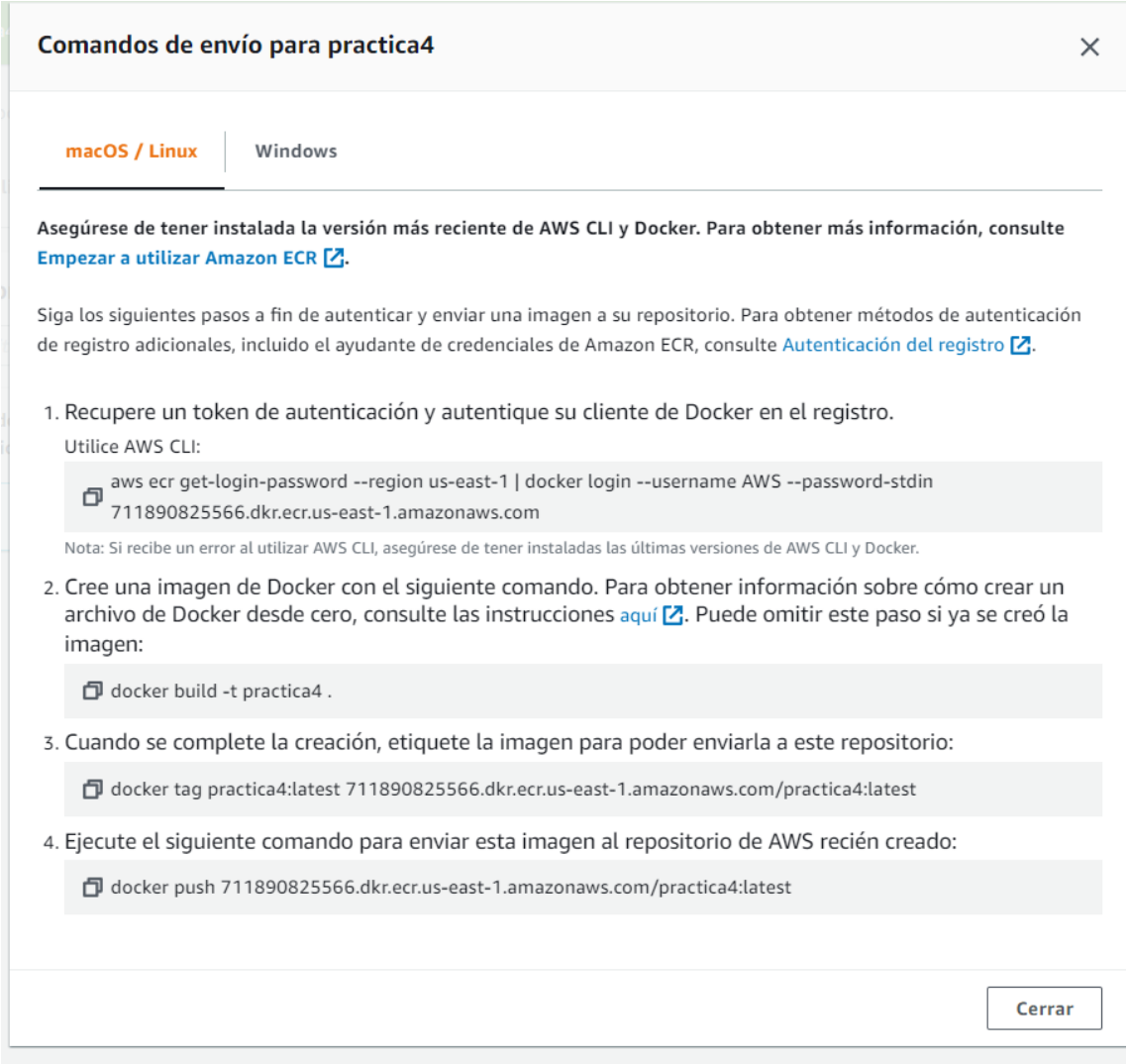


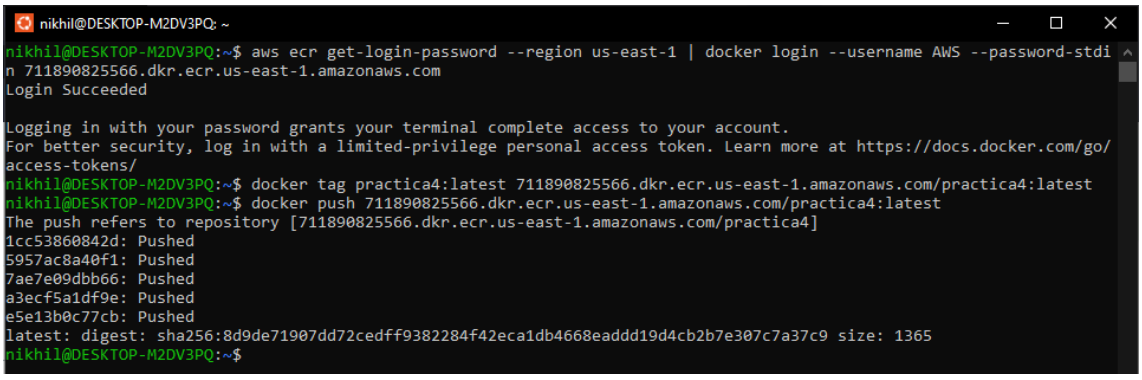
2. Crear un repositorio en ECR y subir el contenedor creado en el paso 1

Se introducen las credenciales que se han copiado desde el laboratorio en un fichero con nombre *credentials* en la carpeta oculta .aws.



A continuación se crea el repositorio en la plataforma AWS.

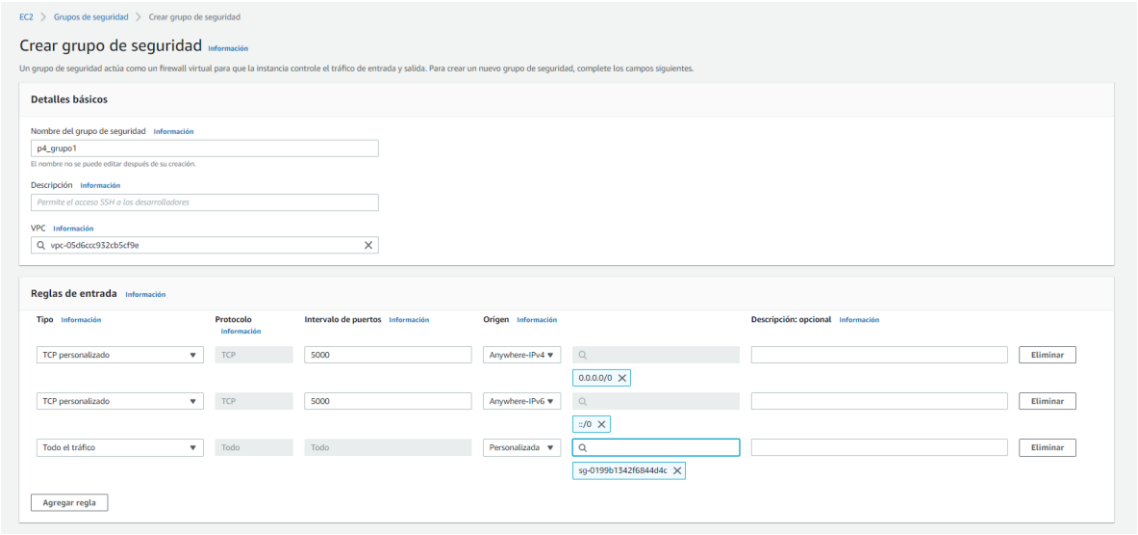




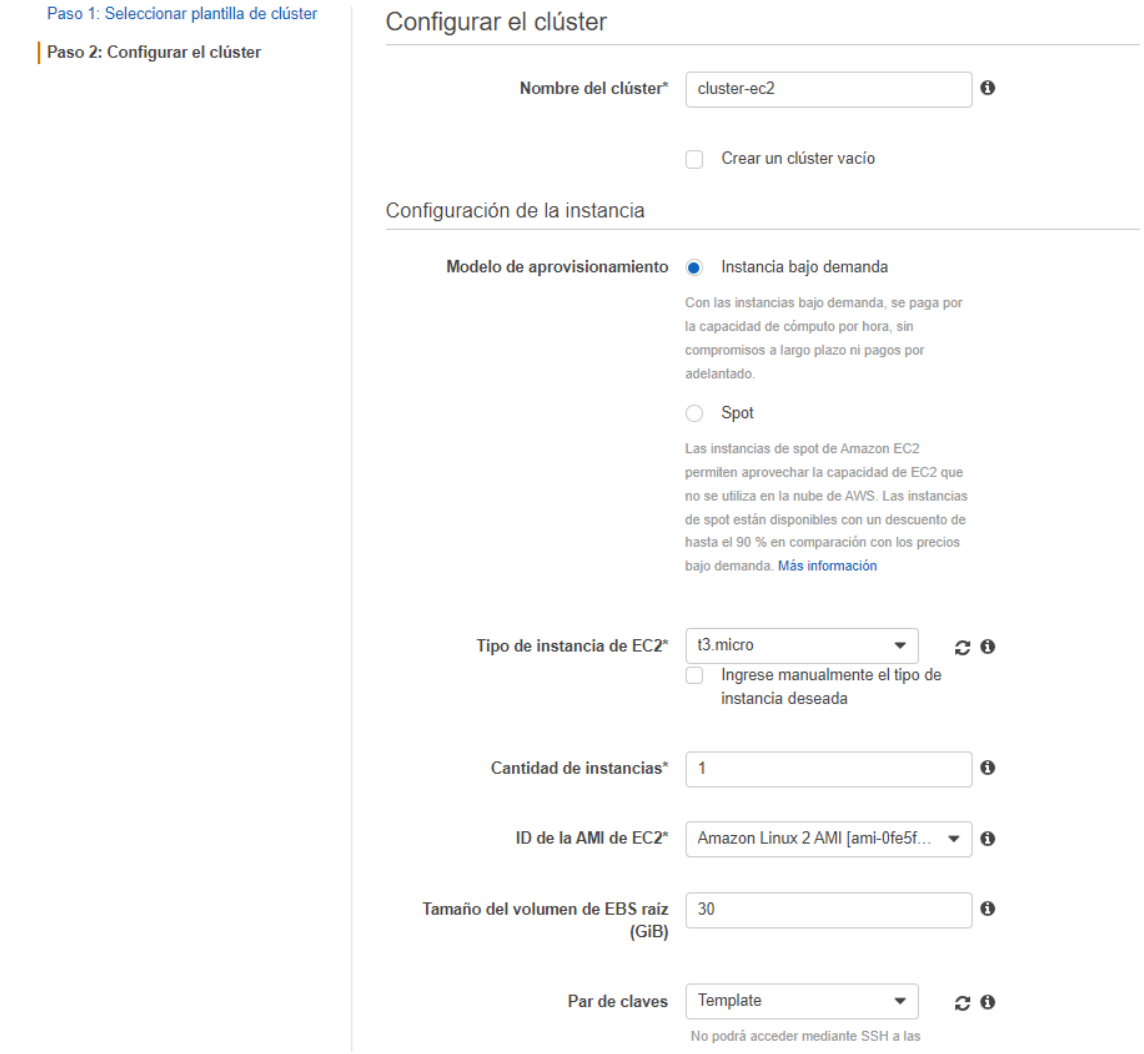
Una vez que se ha creado el repositorio se ejecutan las respectivas instrucciones para poder realizar un push del contenido.

3. Desplegar el contenedor usando ECS

Para desplegar un contenedor ECS se van a crear 2 recursos, el primero es un clúster EC2 y lo segundo es su respectiva tarea.



Se crea el grupo de seguridad que se utilizará posteriormente.


Paso 1: Seleccionar plantilla de clúster
Paso 2: Configurar el clúster
Configurar el clúster
Nombre del clúster*
cluster-ec2
Crear un clúster vacío
Configuración de la instancia
Modelo de aprovisionamiento
Instancia bajo demanda
Con las instancias bajo demanda, se paga por la capacidad de cómputo por hora, sin compromisos a largo plazo ni pagos por adelantado.
Spot
Las instancias de spot de Amazon EC2 permiten aprovechar la capacidad de EC2 que no se utiliza en la nube de AWS. Las instancias de spot están disponibles con un descuento de hasta el 90 % en comparación con los precios bajo demanda. Más información
Tipo de instancia de EC2*
t3.micro
Ingrese manualmente el tipo de instancia deseada
Cantidad de instancias*
1
ID de la AMI de EC2*
Amazon Linux 2 AMI [ami-0fe5f...
Tamaño del volumen de EBS raíz (GiB)
30
Par de claves
Template
No podrá acceder mediante SSH a las

Se utiliza el grupo de seguridad que se creó anteriormente.

Instancia de contenedor Rol de IAM

El agente de contenedores de Amazon ECS realiza llamadas a las acciones de la API de Amazon ECS en su nombre, por lo que las instancias de contenedor que ejecutan el agente requieren la política y el rol de IAM ecsInstanceRole para que el servicio sepa que el agente le pertenece a usted. Si aún no tiene el ecsInstanceRole, podemos crear uno para usted.

Rol de IAM de instancia de contenedor LabRole

Para que las instancias de contenedor reciban el nuevo formato de ARN e ID de recurso, el usuario raíz debe aceptar el rol de IAM de la instancia de contenedor. Acepte e inténtelo de nuevo.

Etiquetas

Clave Valor

Agregar clave Agregar valor

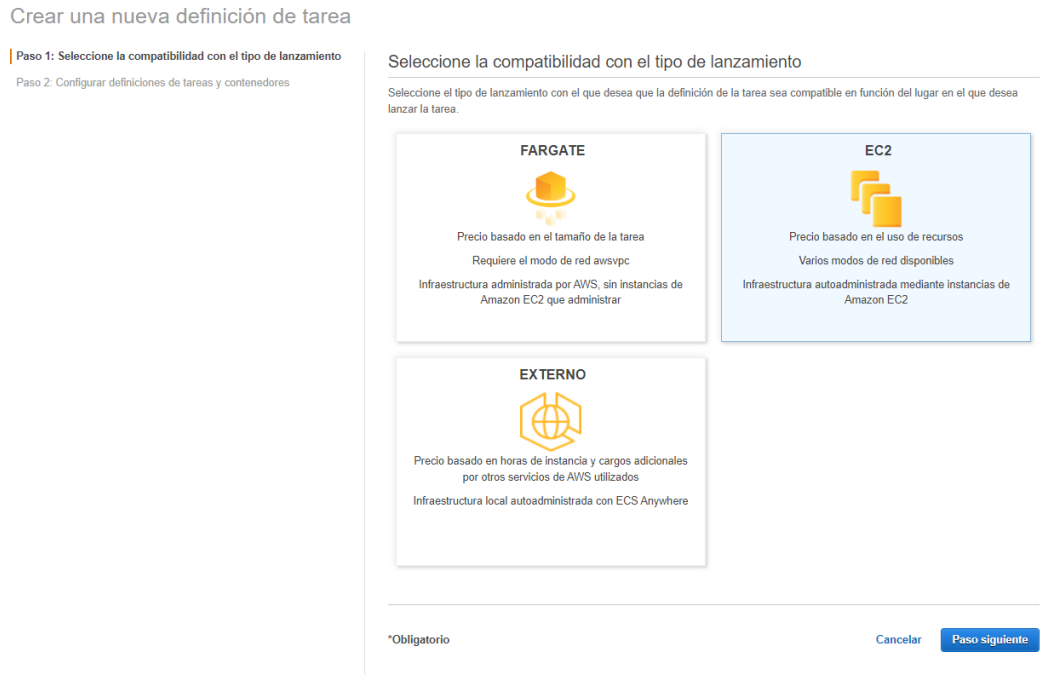
Información de contenedores de CloudWatch

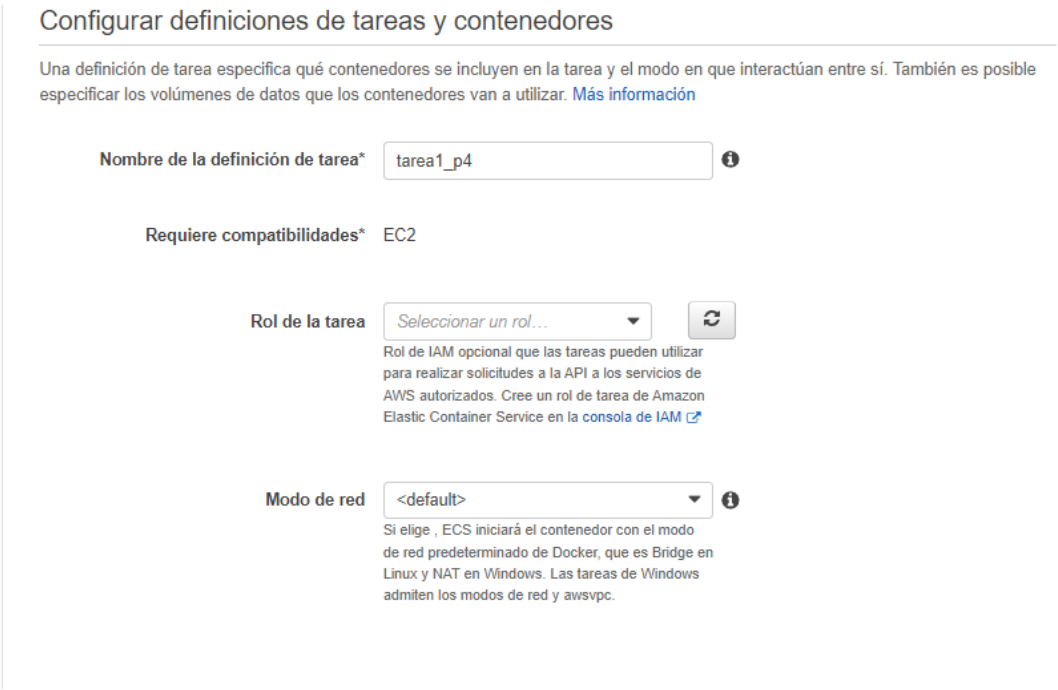
Información sobre contenedores de CloudWatch es una solución de supervisión y resolución de problemas para aplicaciones y microservicios en contenedores. Recopila, agrega y resume la utilización de los recursos de computación, como la CPU, la memoria, el disco y la red, así como información de diagnóstico, como los errores de reinicio de los contenedores, a fin de ayudar a aislar los problemas de los clústeres y resolverlos rápidamente. Más información

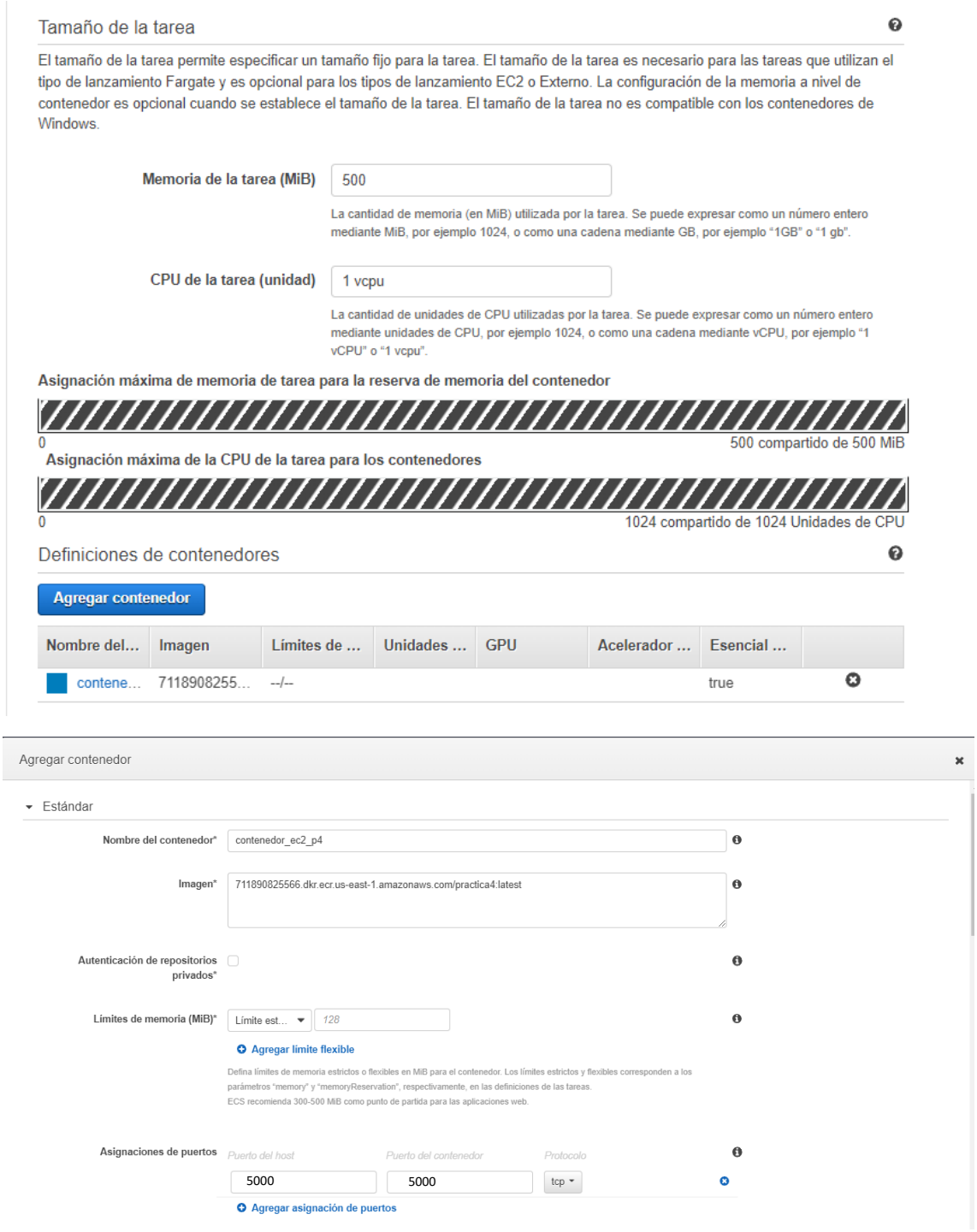
Información de contenedores de CloudWatch Habilitar Información sobre contenedores

*Obligatorio Cancelar Anterior Crear

A partir de aquí se lleva a cabo la definición de la tarea, como el clúster es de EC2, la tarea tiene que seguir el mismo formato.



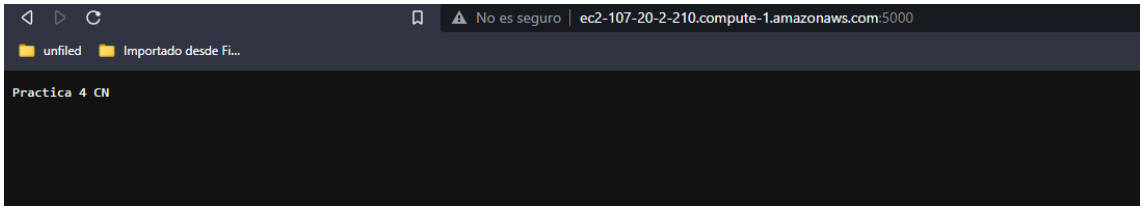




Cabe destacar la asignación de puertos, por ello se les asigna el puerto 5000 como se hizo anteriormente.

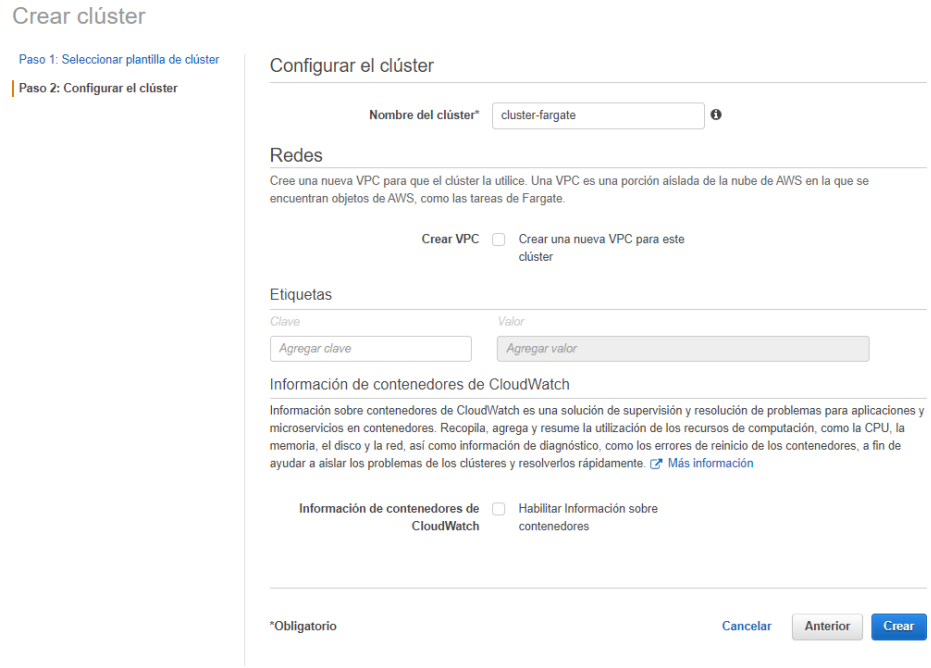
Una vez terminada la definición de la tarea, se procede a realizar su ejecución, que no es más que cargar la tarea creada y ejecutarla.

Se puede observar el funcionamiento del contenedor ECS.

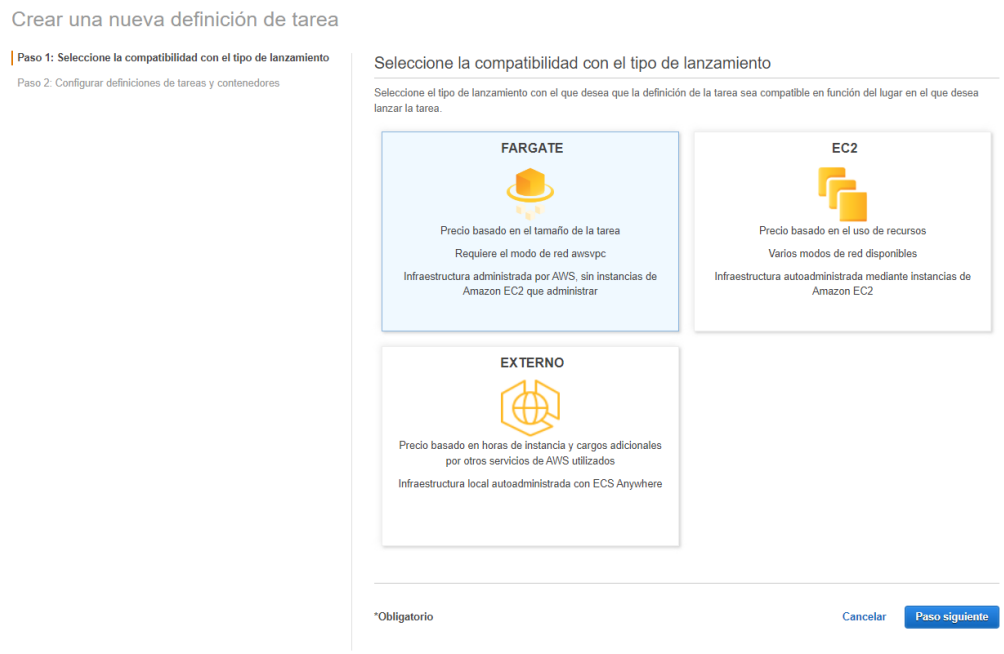


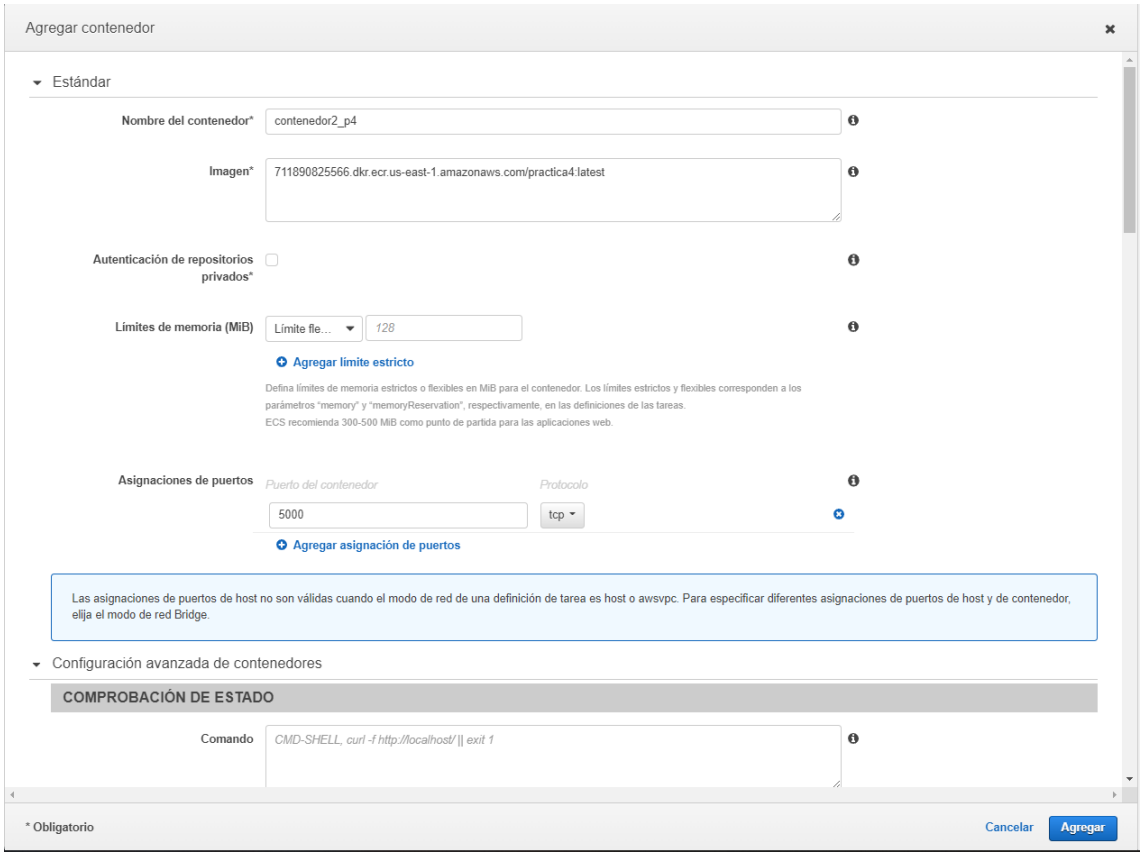
4. Desplegar el contenedor usando Fargate y comparar la experiencia

Para crear el contenedor usando Fargate es prácticamente lo mismo que el caso anterior solo que a la hora de crear el clúster se selecciona el de Solo Redes.



A continuación se define la nueva tarea y en esta ocasión se escoge FARGATE.





Crear una nueva definición de tarea

Paso 1: Seleccione la compatibilidad con el tipo de lanzamiento

Paso 2: Configurar definiciones de tareas y contenedores

Configurar definiciones de tareas y contenedores

Una definición de tarea especifica qué contenedores se incluyen en la tarea y el modo en que interactúan entre sí. También es posible especificar los volúmenes de datos que los contenedores van a utilizar. Más información

Nombre de la definición de tarea* ec2_p4_tarea2

Requiere compatibilidades* FARGATE

Rol de la tarea LabRole

Rol de IAM opcional que las tareas pueden utilizar para realizar solicitudes a la API a los servicios de AWS autorizados. Cree un rol de tarea de Amazon Elastic Container Service en la consola de IAM

Modo de red awsvpc

Si elige , ECS iniciará el contenedor con el modo de red predeterminado de Docker, que es Bridge en Linux y NAT en Windows. Las tareas de Windows admiten los modos de red y awsvpc.

Familia de sistemas operativos Linux

Ejecución de tareas Rol de IAM

Este rol es requerido por las tareas para extraer imágenes de contenedores y publicar registros de contenedores en Amazon CloudWatch en su nombre. Si aún no tiene el rol ecsTaskExecutionRole, podemos crear uno para usted.

Rol de ejecución de tareas LabRole

Tamaño de la tarea

El tamaño de la tarea permite especificar un tamaño fijo para la tarea. El tamaño de la tarea es necesario para las tareas que utilizan el tipo de lanzamiento Fargate y es opcional para los tipos de lanzamiento EC2 o Externo. La configuración de la memoria a nivel de contenedor es opcional cuando se establece el tamaño de la tarea. El tamaño de la tarea no es compatible con los contenedores de Windows.

Memoria de la tarea (GB) 0,5GB

La cantidad de memoria (en MiB) utilizada por la tarea. Se puede expresar como un número entero mediante MiB, por ejemplo 1024, o como una cadena mediante GB, por ejemplo "1GB" o "1 gb".

CPU de la tarea (vCPU) 0,25 vCPU

La cantidad de unidades de CPU utilizadas por la tarea. Se puede expresar como un número entero mediante unidades de CPU, por ejemplo 1024, o como una cadena mediante vCPU, por ejemplo "1 vCPU" o "1 vcpu".

Asignación máxima de memoria de tarea para la reserva de memoria del contenedor

0 — 512 compartido de 512 MiB

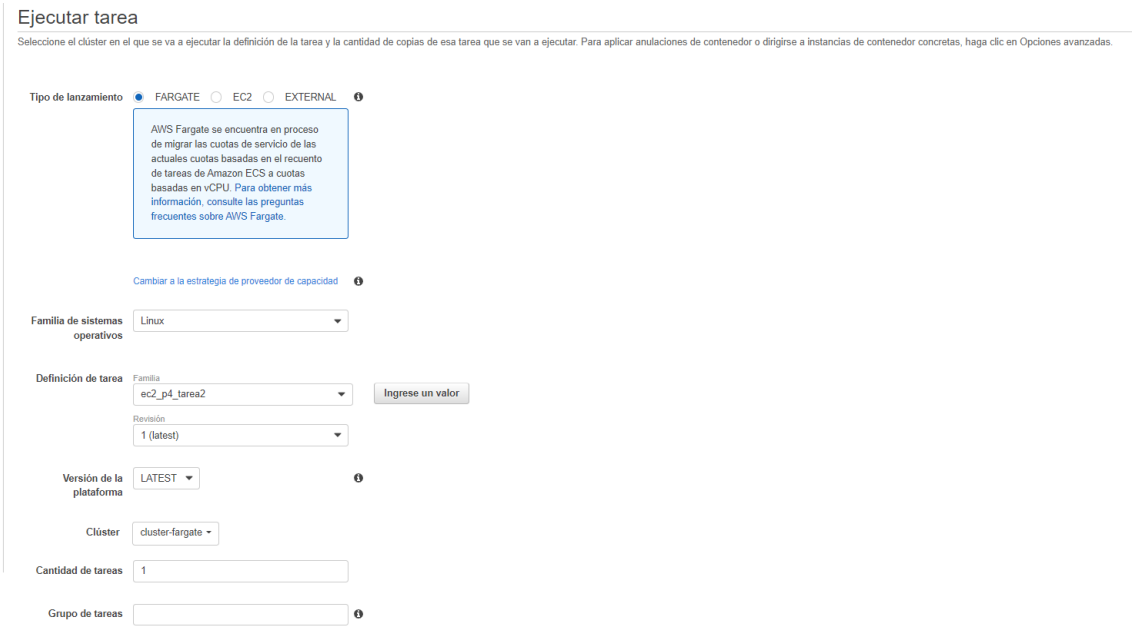
Asignación máxima de la CPU de la tarea para los contenedores

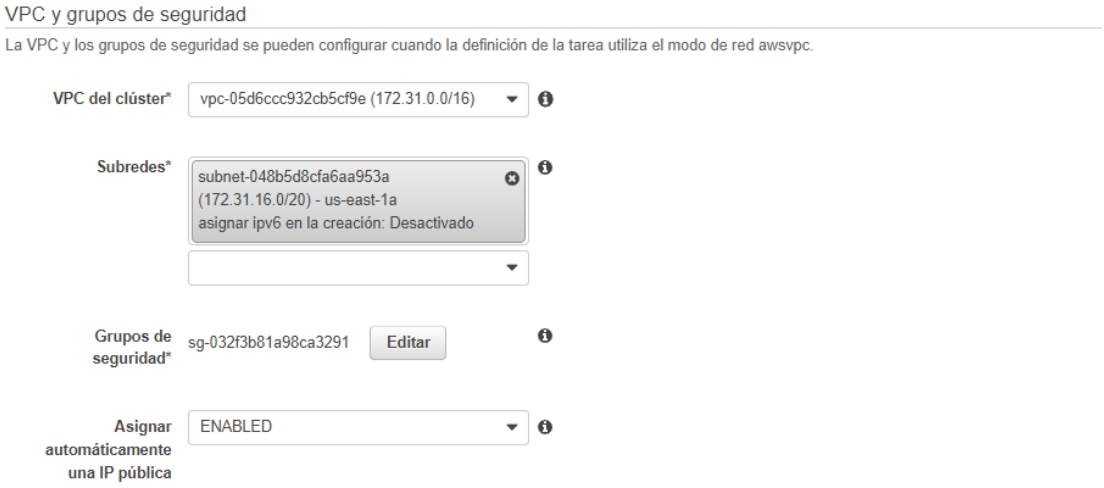
0 — 256 compartido de 256 Unidades de CPU

Definiciones de contenedores

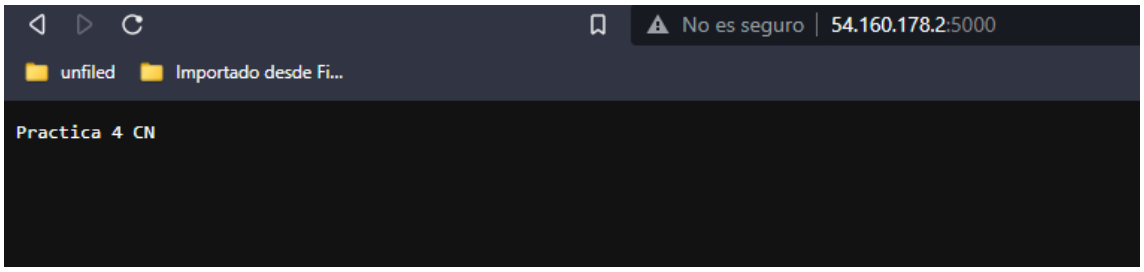
Agregar contenedor

Nombre del...	Imagen	Límites de ...	Unidades ...	GPU	Acelerador ...	Esencial ...	
contene...	7118908255...	--/--				true	





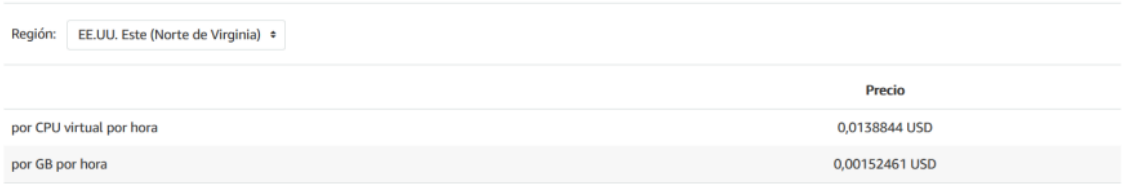
Se puede observar que el clúster FARGATE funciona correctamente.



ESTIMACIÓN DE GASTOS

Cuando se despliega el contenedor ECS se utiliza una instancia de tipo t3.micro. Esta a comparación con las otras prácticas es un poco más cara. El coste por cada hora constante de uso es de 0.0104$/h. Esto juntado al amazon EBS supone un coste mensual de 11.27$. Lo cual se traduce en 135.24$ anuales.

Mientras que el despliegue del contenedor usando FARGATE tiene los siguientes gastos:



Esto según la configuración que se ha utilizado se traduce en 3.05$ mensuales o que es lo mismo 36,6$ anuales.