

**TRABAJO DE GRADO
Opción Seminario-Diplomado.**

Informe técnico de Aprendiendo a navegar en los servicios de AWS

Corporación Universitaria Remington.

Nombre de la facultad: Ingenierías

Nombre del programa académico: Ing. de Sistemas

Presentado por:

MARLON ENRIQUE CALDERON BUELVAS

EDWIN LEONARDO CRISTANCHO RUIZ

JUAN MANUEL CHONA ROMERO

Tutor:

JUAN PABLO BERRIO LOPEZ.

Seminario (AWS) AMAZON WEB SERVICES.

2024.

Agradecimientos

Este trabajo está dedicado a todos aquellos que creemos en el poder del aprendizaje, colaboración y trabajo en equipo, agradecemos a nuestro tutor de este curso quien nos estuvo apoyando durante el desarrollo de este proceso, especialmente a nuestras familias y amigos quienes nos brindaron su apoyo y motivación.

Este trabajo es el resultado del esfuerzo y dedicación de cada uno de nosotros por querer superarnos como futuros profesionales y que mejor manera en aprendizajes de AWS.

Finalmente, a Dios por permitirnos conocer a grandes personas durante el desarrollo de este trabajo.

Tabla de Contenidos.

Resumen.....	6
Palabras clave.....	6
Marco conceptual y contextual	6
1. Decisión Estratégica de Infraestructura IT para TechSolutions S.A.	7
Análisis de la situación.	8
Estado actual	8
Evaluación de Riesgos y Seguridad.....	8
Riesgos en el data-center	8
Medidas para mitigar estos riesgos.	9
Riesgos en la nube.	10
Medidas para mitigar estos riesgos.....	10
Escalabilidad.	11
A. Data-center.....	11
B. Nube.....	11
Personal y operacional.	12
A. Data-center.....	12
B. Nube.....	12
Recomendación final.	12
Desarrollo e implementación del aprendizaje.....	13
2. Creación de VPC e Instancia EC2	14
2.1 Sitio estático en S3.....	46
3. Arquitectura de AWS para aplicación con dos instancias con balanceador de carga de contenedores en su interior.	61
Conclusiones.....	71
Bibliografía	72

Tabla de imágenes.

IMAGEN 1	14
IMAGEN 2	15
IMAGEN 3	15
IMAGEN 4	16
IMAGEN 5	16
IMAGEN 6	17
IMAGEN 7	17
IMAGEN 8	18
IMAGEN 9	18
IMAGEN 10	19
IMAGEN 11	19
IMAGEN 12	20
IMAGEN 13	20

IMAGEN 14	21
IMAGEN 15	21
IMAGEN 16	22
IMAGEN 17	22
IMAGEN 18	23
IMAGEN 19	23
IMAGEN 20	24
IMAGEN 21	25
IMAGEN 22	26
IMAGEN 23	26
IMAGEN 24	27
IMAGEN 25	28
IMAGEN 26	28
IMAGEN 27	29
IMAGEN 28	29
IMAGEN 29	30
IMAGEN 30	30
IMAGEN 31	31
IMAGEN 32	31
IMAGEN 33	32
IMAGEN 34	32
IMAGEN 35	33
IMAGEN 36	33
IMAGEN 37	34
IMAGEN 38	34
IMAGEN 39	35
IMAGEN 40	35
IMAGEN 41	36
IMAGEN 42	36
IMAGEN 43	37
IMAGEN 44	37
IMAGEN 45	38
IMAGEN 46	38
IMAGEN 47	39
IMAGEN 48	39
IMAGEN 49	40
IMAGEN 50	40
IMAGEN 51	41
IMAGEN 52	41
IMAGEN 53	42
IMAGEN 54	42
IMAGEN 55	43
IMAGEN 56	43
IMAGEN 57	44
IMAGEN 58	44
IMAGEN 59	45
IMAGEN 60	45
IMAGEN 61	46
IMAGEN 62	47
IMAGEN 63	47

IMAGEN 64	48
IMAGEN 65	48
IMAGEN 66	49
IMAGEN 67	49
IMAGEN 68	50
IMAGEN 69	50
IMAGEN 70	51
IMAGEN 71	51
IMAGEN 72	52
IMAGEN 73	52
IMAGEN 74	53
IMAGEN 75	53
IMAGEN 76	54
IMAGEN 77	54
IMAGEN 78	55
IMAGEN 79	55
IMAGEN 80	56
IMAGEN 81	56
IMAGEN 82	57
IMAGEN 83	57
IMAGEN 84	58
IMAGEN 85	58
IMAGEN 86	59
IMAGEN 87	59
IMAGEN 88	60
IMAGEN 89	60
IMAGEN 90	62
IMAGEN 91	62
IMAGEN 92	63
IMAGEN 93	63
IMAGEN 94	64
IMAGEN 95	64
IMAGEN 96	65
IMAGEN 97	65
IMAGEN 98	66
IMAGEN 99	66
IMAGEN 100	67
IMAGEN 101	67
IMAGEN 102	68
IMAGEN 103	68
IMAGEN 104	69
IMAGEN 105	69
IMAGEN 106	70
IMAGEN 107	70

Resumen

Hay un problema recurrente para algunas empresas cuando quieren estar en internet o tan solo para tener servidores que les permita mejorar su funcionamiento, en este informe técnico miraremos un caso de una empresa que está en la dificultad de crear un nuevo data center o estar en la nube con AWS. [1]

Miraremos como se puede implementar instancia en la nube y diferentes formas de publicar y balancear una página o aplicación, y esto permitirá dar un vistazo a servicios en la nube con AWS.[2]

Palabras clave

Servicios de la nube: son servidores que prestan servicios de almacenamiento, bases de datos, software y otras adecuaciones muy útiles.

Data-center: son instalación que contienen infraestructura informática para almacenar, ejecutar, entregar y crear aplicaciones y servicios,

Big data: son para procesar datos en gran cantidad donde se busca encontrar patrones que beneficien.

Inteligencia artificial: es una simulación computacional donde se busca la resolución de problemas como lo haría un ser humano.

Blockchain: un método de seguridad compartido e inmutable que realiza un registro en una red específica.

Marco conceptual y contextual

Servicios de la nube: son servidores que prestan servicios de almacenamiento, bases de datos, software y otras adecuaciones muy útiles.

Data-center: son instalación que contienen infraestructura informática para almacenar, ejecutar, entregar y crear aplicaciones y servicios.

Instancia: recurso para implementar un servidor en la nube.

máquina virtual

contenedor: encapsular unidades estándar que incluya lo que se necesita para la ejecución de un programa.

Nube: espacio en la red.

IP: una dirección única.

AWS: plataforma que presta servicios en la nube.

Docker: aplicación para crear contenedores.

interfaz gráfica: conjunto de imágenes en programas informáticos para fácil navegación.

protocolo SSH: administración remota.

Comando: ordenes en un programa.

Nginx: programa para balancear cargas.

S3: servicio de guardar archivos estáticos de AWS.

VPC: permite administrar recursos en la nube para AWS

EC2: capacidad informática modificable y segura en AWS. [3]

1. Decisión Estratégica de Infraestructura IT para TechSolutions S.A.

Análisis de la situación.

Estado actual

- Empresa mediana
- Sistema IT básico
- Dedicada al desarrollo de software
- Data-center local
- Rápido crecimiento
- 150 empleados
- 50 desarrolladores de software
- Proyectos sencillos y complejos
- Estado esperado
- Modernización infraestructura IT
- Crecer el data-center o migrar a la nube
- Un ambiente robusto y flexible
- Fiabilidad
- Innovación y adopción de tecnologías emergente

Evaluación de Riesgos y Seguridad

Riesgos en el data-center

- Desaprovechar los recursos
- Mantenimiento de la infraestructura
- Diseño de la infraestructura

- Tiempos e inversiones requeridos al momento del escalado
- Capacidad física limitada
- Fuga de información tanto física como lógica
- Daños y fallas de la infraestructura
- Incumplimiento del servicio
- Ataques de ciberseguridad (software malicioso)
- Accesos no autorizados
- Errores humanos
- Incumplimiento de normativas y políticas

Medidas para mitigar estos riesgos.

- Gestionar y planifica de nueva infraestructura
- Control y monitoreo constante de la infraestructura
- Formar y capacitar al personal
- Evaluar la complejidad al momento del escalado
- Crear un control de acceso y privilegios
- Encriptación de datos
- Revisión y actualización de los sistemas
- Redundancia en la infraestructura
- Protección a los puntos de acceso a la red
- Uso de programas para evitar software malicioso
- Crear políticas, normativas y protocolos

Riesgos en la nube.

- Cargos adicionales no contemplados
- Responsabilidades del proveedor de servicio
- Desaprovechó de recursos
- Fuga de información lógica
- Incumplimiento del servicio
- Ataques de ciberseguridad (software malicioso)
- Facturación costos variables
- Ubicación de la información (tiempo)
- Accesos no autorizados
- Latencias de la red
- Errores humanos
- Incumplimiento de normativas y políticas

Medidas para mitigar estos riesgos.

- Crear políticas, normativas y protocolos.
- Establecer límites de uso de los recursos en la nube en forma continua
- Evaluar las condiciones del proveedor de servicios en la nube
- Diversificación del proveedor
- Configuración servicios de Escalados automáticos
- Crear un control de acceso y privilegios
- Encriptación de datos

- Revisión y actualización de los sistemas
- Protección a los puntos de acceso a la red
- Uso de programas para evitar software malicioso

Escalabilidad.

A. Data-center.

Al crear un data-center no permitiría un escalado rápido y tendrá que asumir costos cada vez que se adquiriera nuevos talentos e infraestructura física y lógica,

Al ser administrado por la empresa tendrá el control total de los componentes, esto conllevará que, al agregar recursos, se considere la mantenibilidad y posibles daños de sus partes.

Algo importante para tener en cuenta son los picos de demanda, para solventar estos se hace imprescindible tener más elementos que proporcionen la capacidad necesaria para tener los servicios funcionando y el personal que este pendiente en el flujo de uso de la infraestructura para mantener solo los recursos para un óptimo funcionamiento.

B. Nube.

Al utilizar servicios en la nube dejaremos que el proveedor tenga control de la infraestructura lo que nos permite analizar las nuevas oportunidades y pedir solo lo necesario al momento de escalar un recurso tanto vertical como horizontalmente donde al configurar escalados automáticos mejorara la capacidad de respuesta cuando se presenten picos de demanda,

Los proveedores de la nube nos ofrecerán gran gama de servicios que dependiendo de la región y el alcance de la empresa hay que validar los tiempos que llevaría al acceso de los aplicativos o servicios gestionados, aunque al ser muy flexibles este método cualquier cambio hecho se puede readaptar si no es lo que se desea.

Personal y operacional.**A. Data-center.**

Se necesita contratar personal para manejar las nuevas instalaciones aumentado la gestión al momento de evaluar el rendimiento tanto operativo como humano lo que provocara complejidades en el funcionamiento, adaptación, planificación y coordinación en el entorno laboral.

La ventaja es que los conocimientos ya existen en la empresa y estos se utilizaran para capacitar directamente desde personal actual evitando errores humanos adaptándolos a nuevos procedimientos para minimizar la interrupción de las operaciones actualmente llevadas.

B. Nube.

Una transición compleja de cultura en todo el equipo de IT donde vemos conceptos, habilidades y cambios de proceso que crearan flujos de trabajo no establecidos hasta el momento afectando la fluidez que actualmente tiene la empresa en sus labores diarias,

Planificando y detallando los documentos tanto de flujos como proceso permita una transición fácil teniendo en cuenta el costo de contratar personal y mantenimiento de la infraestructura actual mientras se crea y migra a la nube asumiendo la pérdida de información si ocurre algún error por la inexperiencia.

Recomendación final.

Teniendo en cuenta las características y requerimientos de la empresa es recomendable migrar a la nube en lugar de ampliar el data-center por las siguientes razones:

En términos de escalabilidad, la nube le permite mayor flexibilidad a la hora de adquirir nuevos recursos y es mucho más rápido que una ampliación a la data center.

Cuando hablamos de costos la situación no es muy diferente, a largo plazo la nube suele ser mucho más económica por que no es necesario adquirir o mantener la infraestructura física adicional.

Con respecto a la accesibilidad, al ser servicio en la nube el acceso se puede lograr desde cualquier lugar con conexión a internet, esto facilita el trabajo remoto y la colaboración de los diferentes equipos de trabajo de la empresa.

En cuanto a seguridad, los proveedores de servicios en la nube procuran siempre brindan medidas de seguridad avanzadas y actualizadas automáticamente, configuradas dependiendo la necesidad del cliente o usuario. En comparación con un data-center físico que requiere una gestión de actualización manual.

La disponibilidad es garantizada en un alto porcentaje, minimizando el riesgo de tiempo inactivo por fallas de hardware o por mantenimientos.

Y, por último, en el apartado de innovación y adaptación a nuevas tecnologías, la nube facilita acceder a tecnologías emergentes y herramientas avanzadas, como la Big Data, la Inteligencia Artificial, la tecnología blockchain y otras... Impulsando la innovación y la competitividad de la empresa.

Desarrollo e implementación del aprendizaje

En esta sección usted presentará los resultados obtenidos o generados del informe técnico. Deberá describir la ejecución de lo aprendido en el curso en un entorno o contexto. Podrá acompañar este ejercicio con gráficas, tablas, imágenes y referencias bibliográficas para comparar con otros ejercicios similares.

2. Creación de VPC e Instancia EC2.

Implementar una instancia EC2 con dos contenedores, estos contenedores deben tener un balanceador de carga.

Para el desarrollo de este punto se hará un paso a paso de cómo se hace para implementar una instancia EC2 en AWS usando imagines como ejemplos para saber cómo debe ir quedando nuestra instancia.

En esta página inicial debemos ingresar con nuestra cuenta y logearnos como usuario ya registrado e iniciar con nuestra creación de la máquina virtual, la página inicial debe ser tal cual nuestra la *imagen1*.

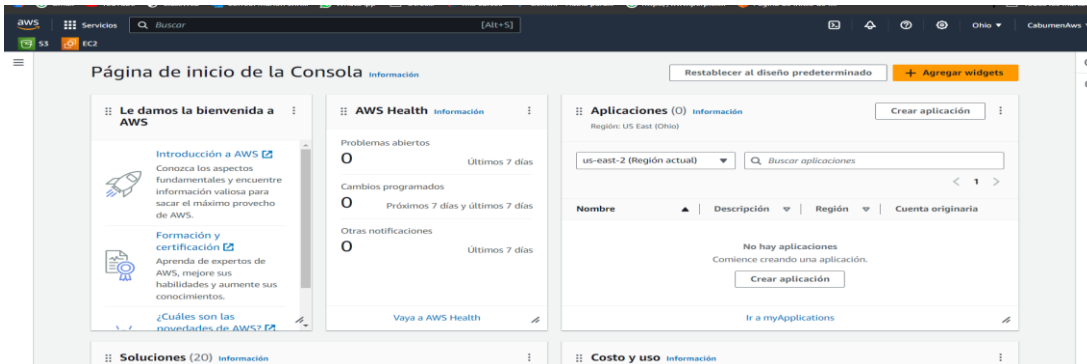


Imagen 1

Antes de lanzar instancias EC2 se debe crear un VPC que nos va a permitir crear y usar los recursos en la nube, donde podemos tener servidores, tarjetas de red, contenedores, routers y otros servicios, seleccionar como muestra la imagen 2.

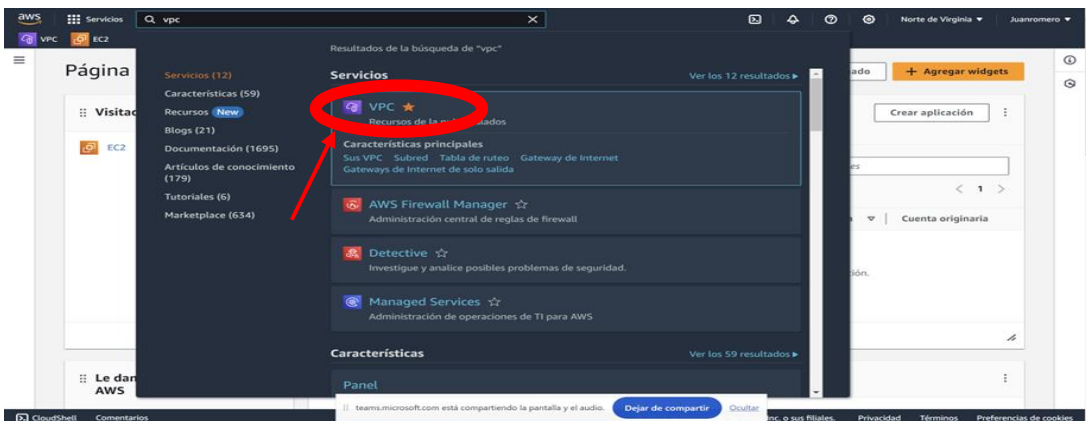


Imagen 2

Cuando hayamos seleccionado la opción de VPC nos debe aparecer tal cual nos muestra la imagen3, de esta manera poder crea ya nuestro VPC y ya estará asociado a más recurso.

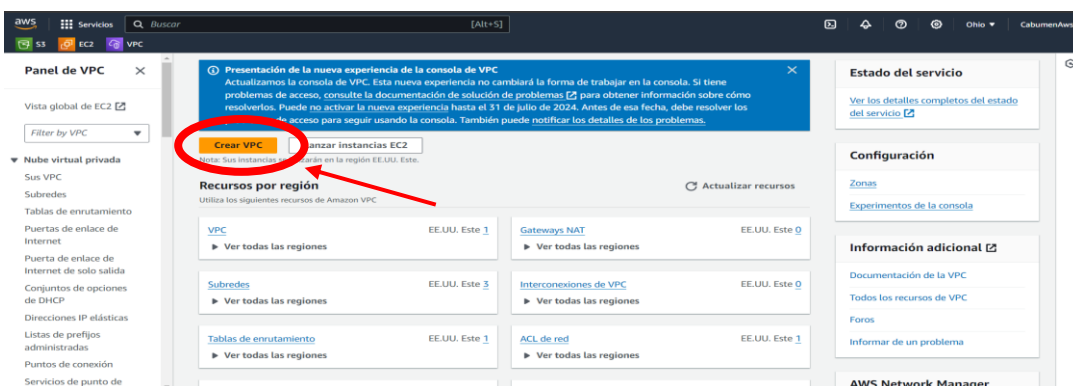


Imagen 3

Cuando se crea el VPC o la red privada una vez le hayamos dado crear VPC en la opción de la imagen 3, nos debe aparecer el siguiente cuadro como lo muestra la imagen 4, de esta manera se puede tener nuestro datacenter virtual con diferentes subredes.

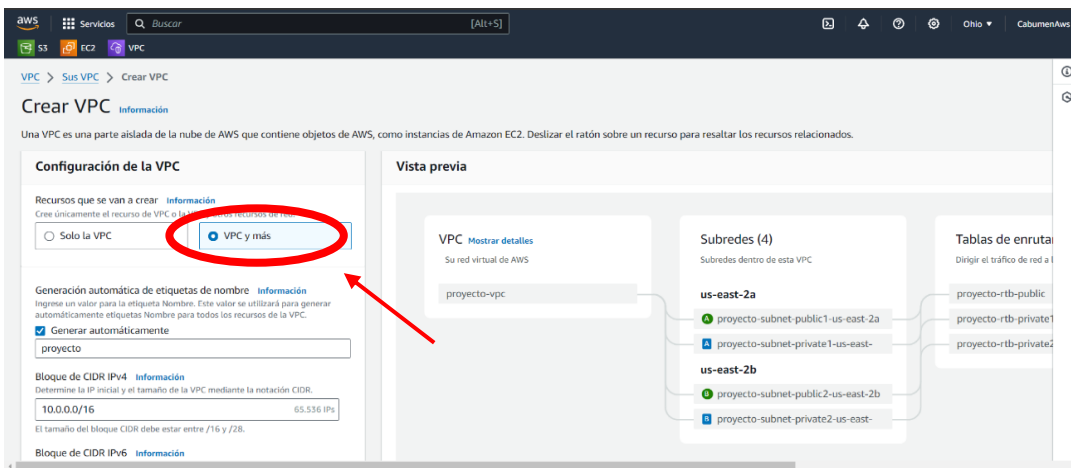


Imagen 4

Se le debe asignar un nombre o generar automáticamente en la parte que esta señalando la imagen 5 , por lo generar siempre se asigna manual para mas seguridad de nosotros mismos.

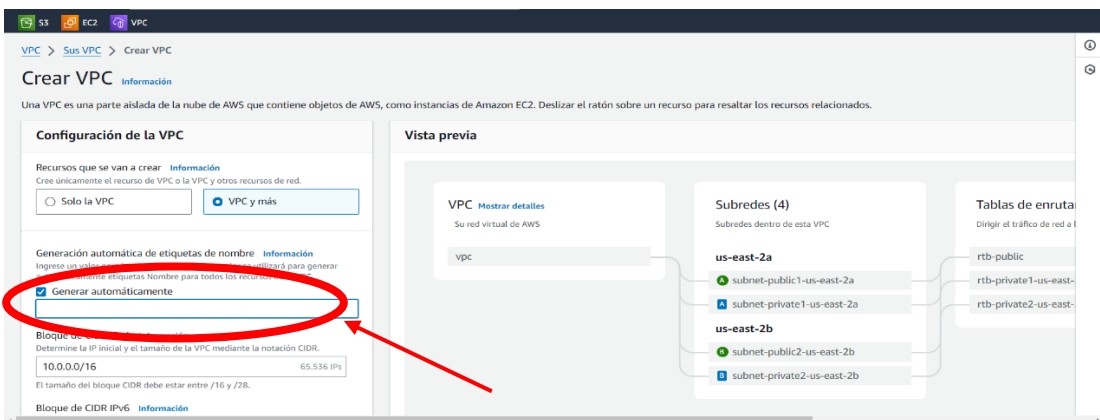


Imagen 5

Cuando se crea el nombre del VPC, este por defecto asocia a unas subredes de las cuales dos son públicas y dos privadas, una de enrutamiento que permiten decidir hacia donde van mis paquetes de datos, tal como lo muestra la imagen 6.

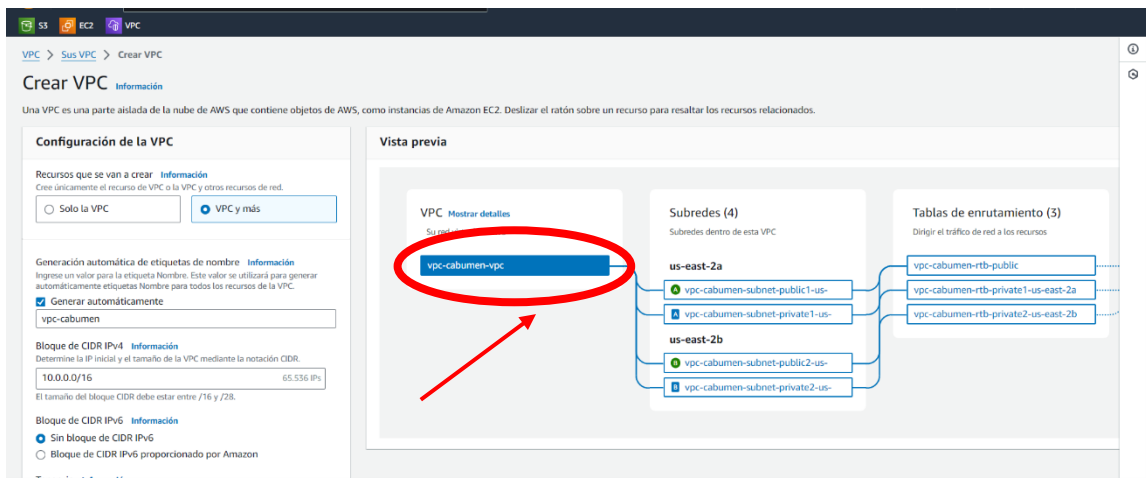


Imagen 6

Si yo quiero tener mas o menos zonas de disponibilidad debo seleccionar el numero que yo quiero o necesito como nuestra la *imagen7*. Eso no tiene ningun costo.

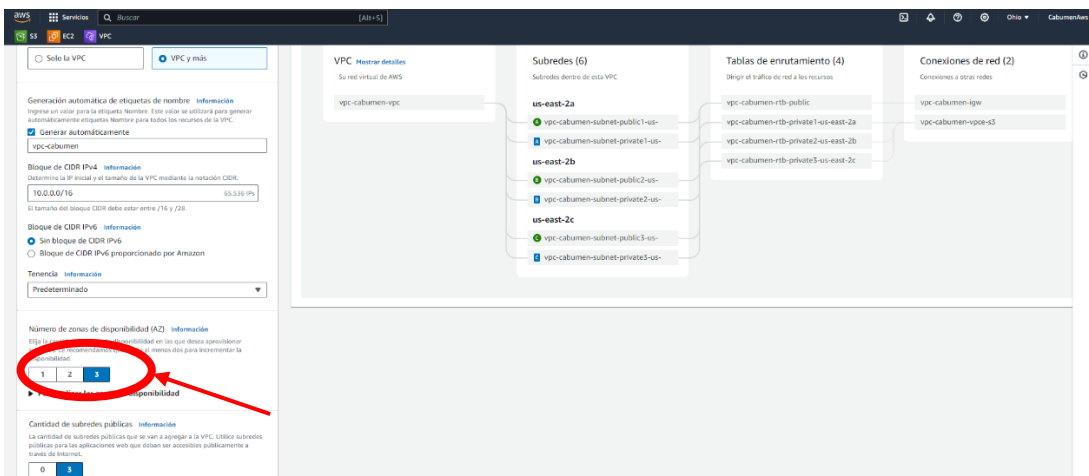


Imagen 7

Una vez ya configurado nuestro VPC le damos crear, como muestra la *imagen8*.

resultados ya creados. Toca tener seguridad de la región en la que se está trabajando para no tener inconvenientes y hacer cambios en la que no es.

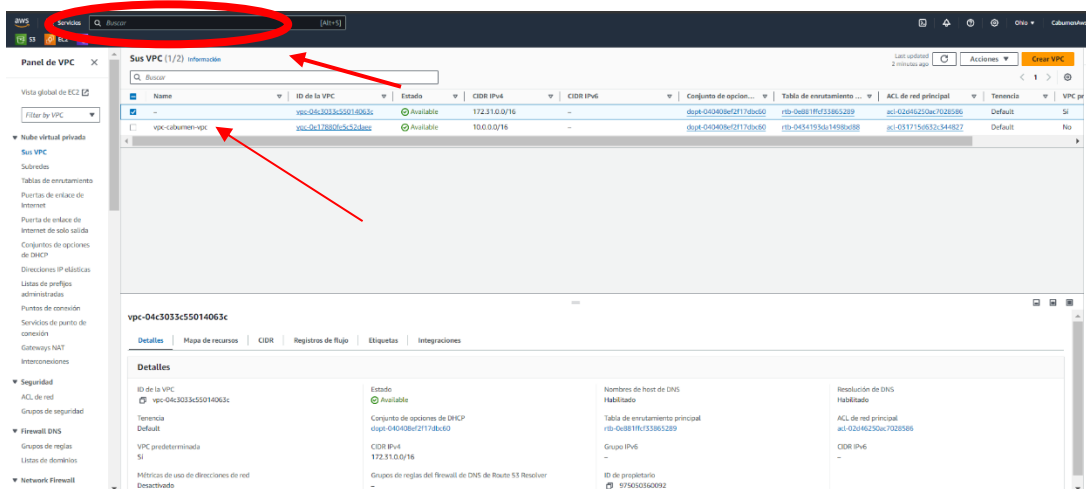


Imagen 10

Para comprobar que se hayan creado todas las subredes 3 públicas y 3 privadas, seleccionamos la opción subredes como muestra la imagen 11, hay que tener en cuenta siempre que se estén creando máquinas virtuales que digan público y privado.

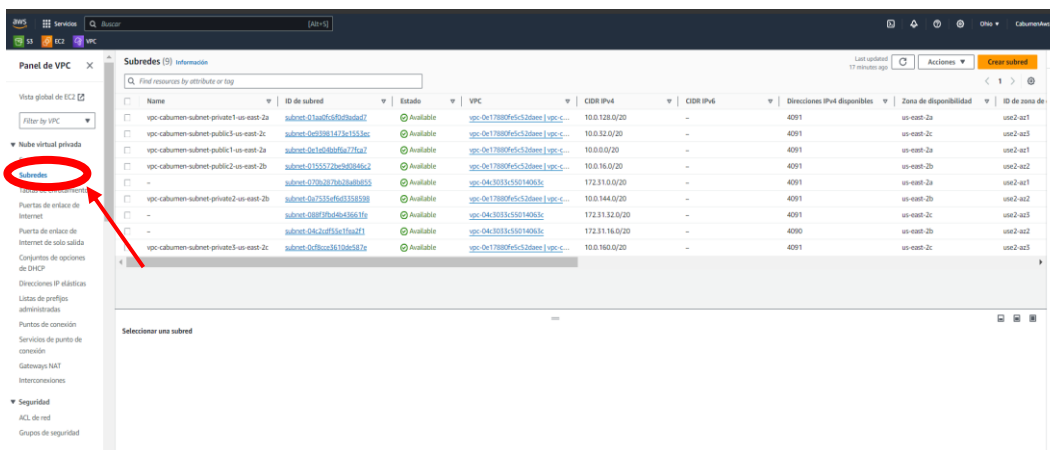


Imagen 11

De esta manera queda armado nuestro VPC lo que nos va a permitir que nuestros recursos tengan conectividad, podemos crear servidores con el direccionamiento pueden tener acceso a internet o

nosotros podemos conectarnos a ellos a través de internet, también se pueden comunicar internamente entre diferentes servidores.

Vamos a la creación de EC2, que es el servicio de maquinas virtuales que en este caso se llaman instancias, en el buscador de la consola de escribimos ec2 y nos va a aparecer el siguiente cuadro que muestra la *imagen 12*.

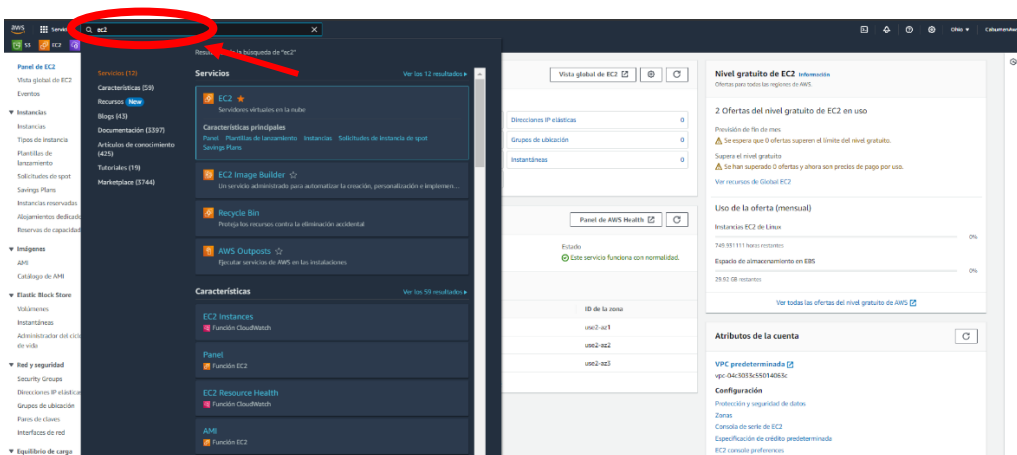


Imagen 12

De esta manera se selecciona para crear la instancia como lo muestra la imagen 13

Les debe mostrar el siguiente cuadro.

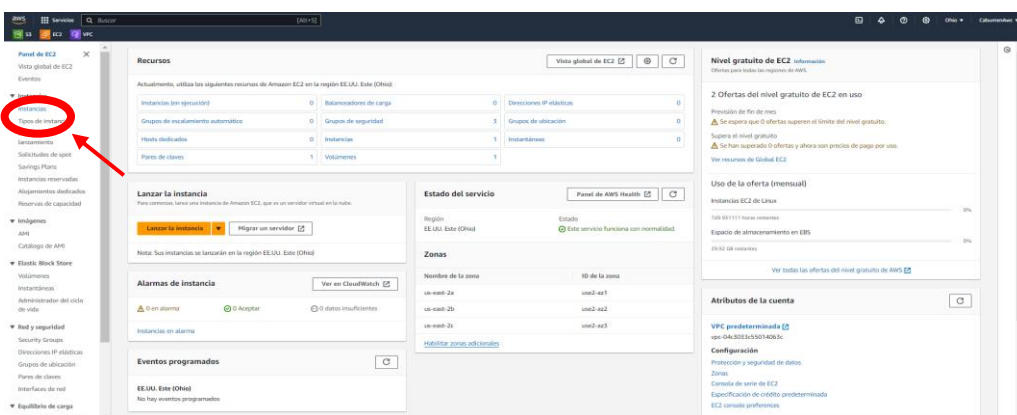


Imagen 13

Cuando se selecciona la instancia nos muestra el siguiente cuadro de la imagen 14 ya sea que este vacía o se tenga ya alguna instancia creada.

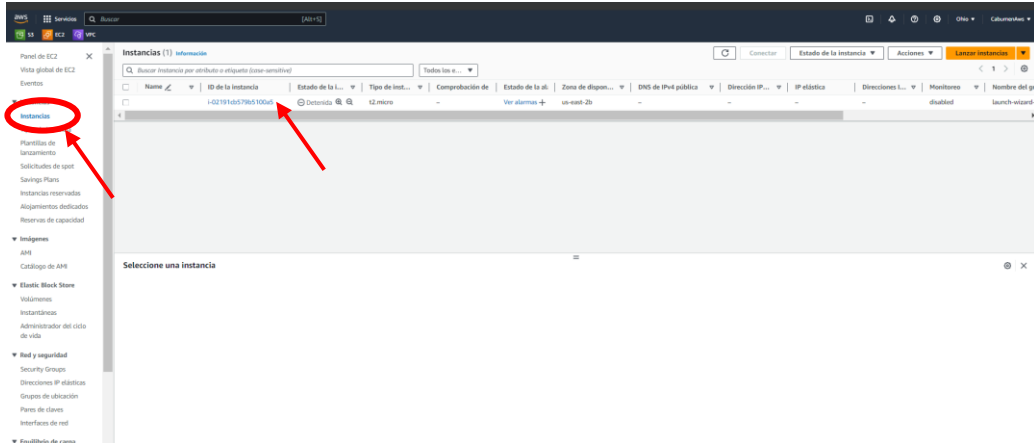


Imagen 14

Para crear una instancia le damos donde dice lanzar instancia en la parte superior derecha de la imagen 15.

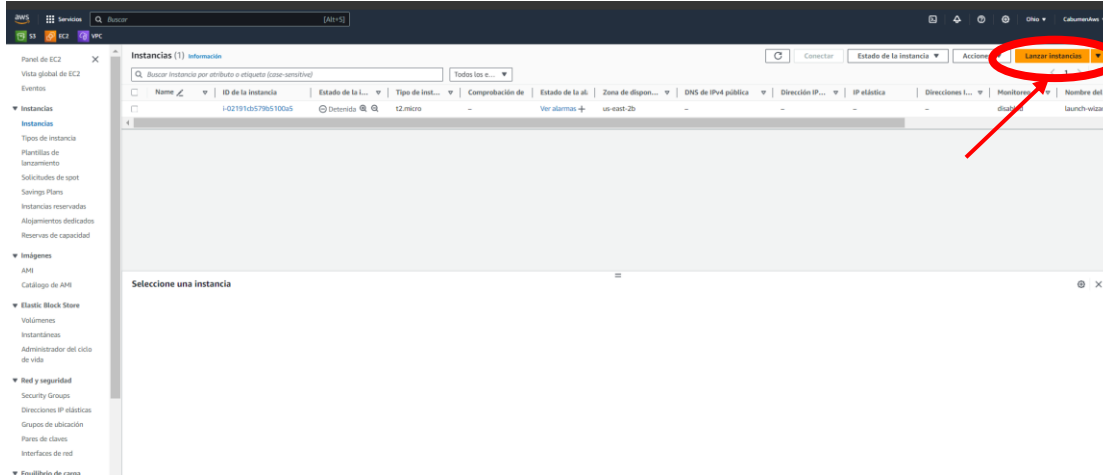


Imagen 15

Cuando hayamos dado en lanzar instancia nos debe aparecer el siguiente cuadro como muestra la imagen 16, se le debe dar un nombre para saber las características que va a tener esa instancia para el servidor que de estar corriendo una vez se lance la instancia.

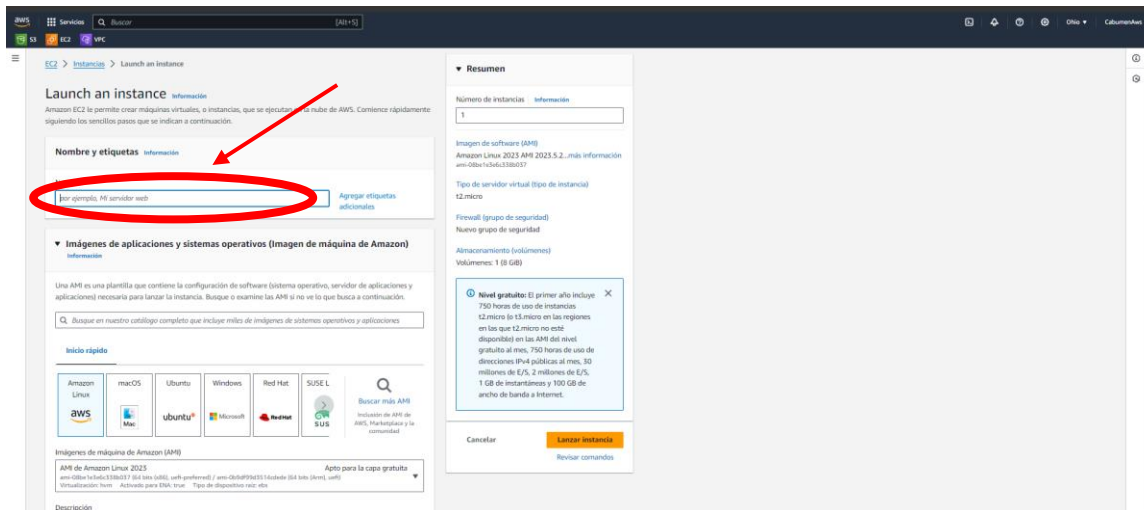


Imagen 16

Una vez se haya asignado nombre a nuestra instancia como se mostró en la *imagen 16*, se debe seleccionar un sistema operativo, el más recomendado es Amazon Linux siempre y cuando usted no tenga una distribución específica.

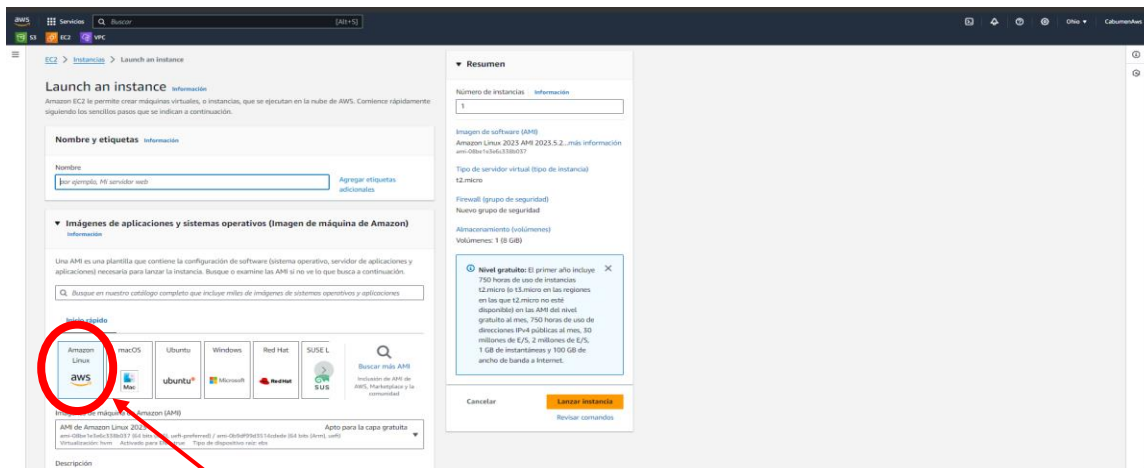


Imagen 17

Para este ejercicio vamos a usar sistema operativo de Windows, tiene una capa gratuita en tema de licencias, observe la *imagen 18*.

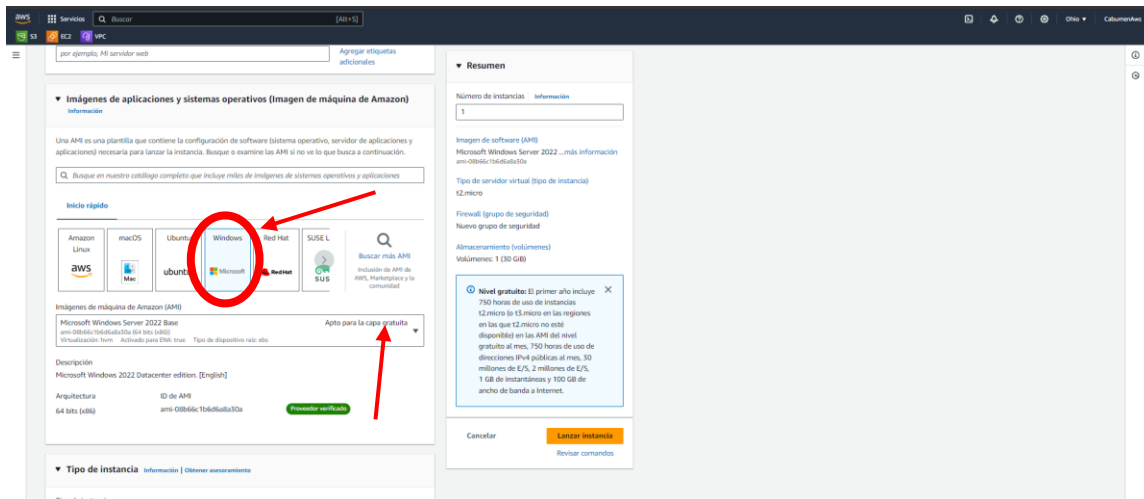


Imagen 18

Una vez se haya seleccionado el sistema operativo, nos vamos al tipo de instancia que requerimos para nuestro ejercicio, se recomienda usar t2.micro que es la capa gratuita y es de uso general para hacer pruebas. Observe la *imagen 19*.

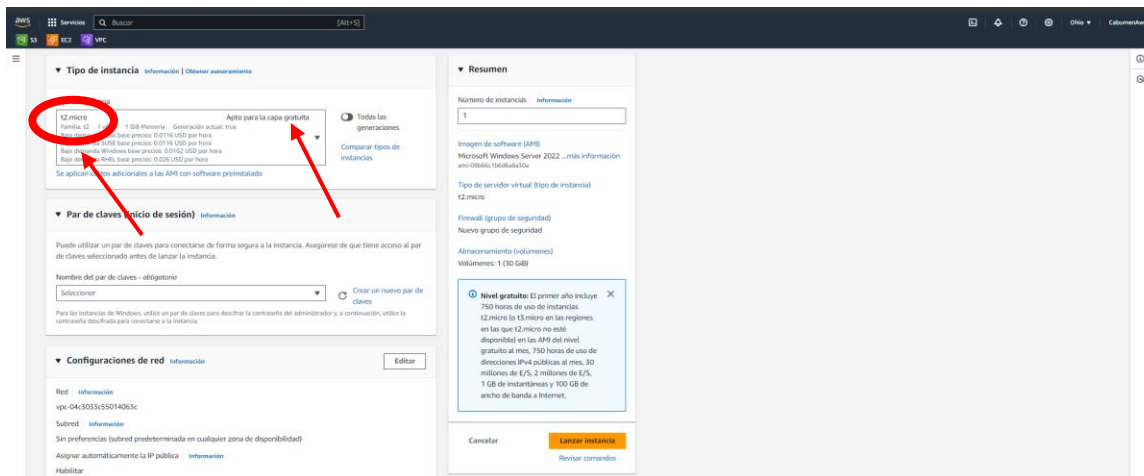


Imagen 19

En el siguiente paso nos va a pedir una autenticación, la forma de autenticarse es con un certificado con seguridad, no se va a ingresar contraseña, toca hacer un procedimiento para obtener esa contraseña y el certificado uno público y otro privado, nos van a brindar el

certificado público en un archivo de descarga y el privado se queda en el servidor, para poder logarse a la maquina debe ser con el certificado público. Ojo este certificado es único y si lo pierde no hay forma de recuperarlo y no va a poder entrar a esa instancia que usted ya había creado tampoco hay forma de volver a crearlo para esa instancia que ya tiene asociado ese archivo público que le fue entregado, le toca hacer el procedimiento desde cero y la información que tenía en esa instancia se pierde.

Observe la *imagen 20* donde debe ingresar para descargar el certificado.

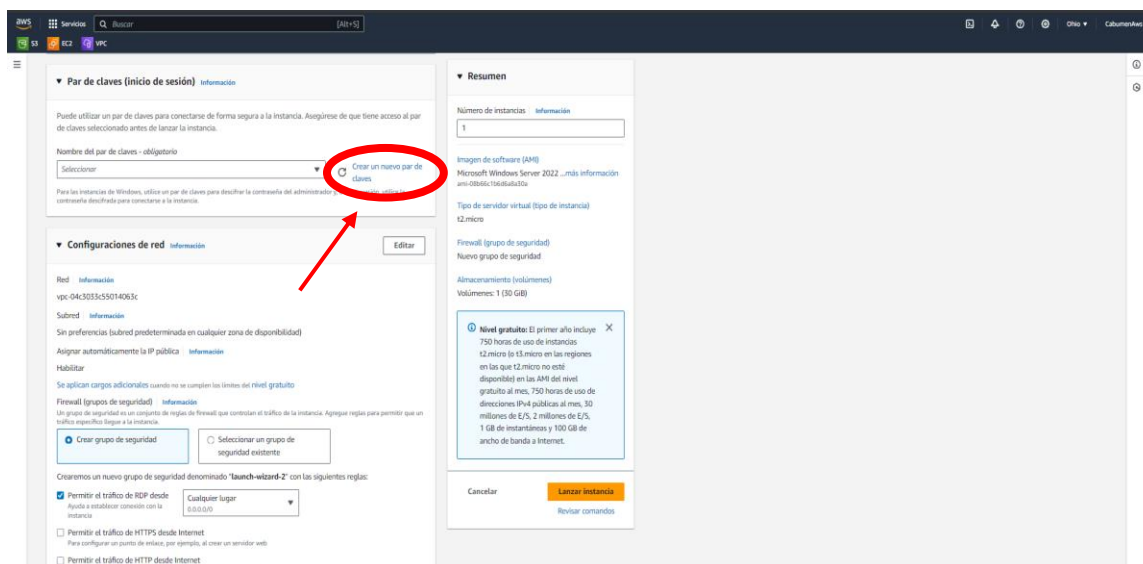


Imagen 20

Para la descarga del archivo nos muestra el siguiente cuadro, debemos colocar un nombre que va a ser la forma con la cual nos podemos autenticar como muestra la *imagen21*, Esto va a mandar la descarga del archivo que es la clave publica, se debe guardar bien para que no vaya a tener inconvenientes con su instancia como se describió en la *imagen20*.

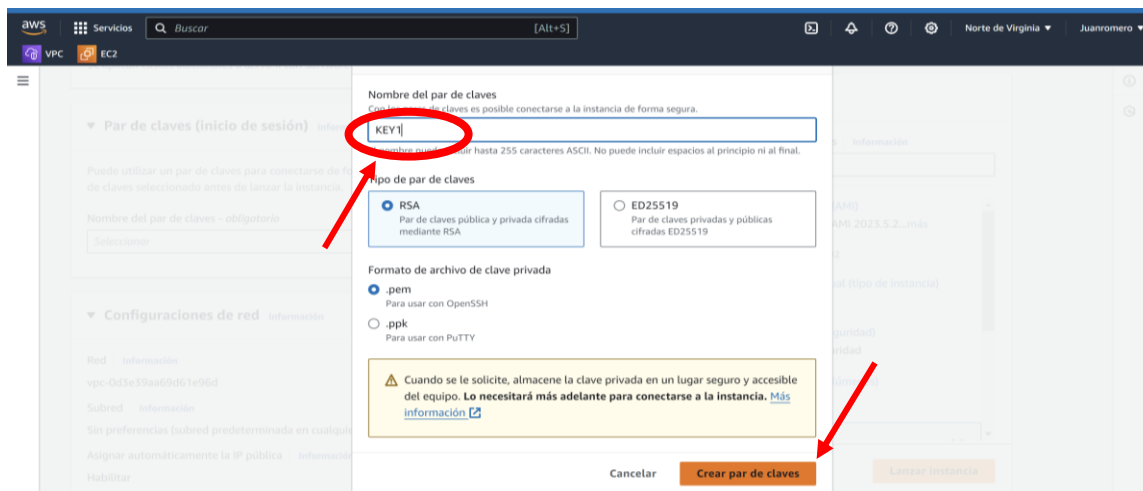


Imagen 21

Ahora viene la configuración de red, solo tenemos un VPC que se creó al inicio. Y si queremos que la maquina se pueda ver desde internet se debe cambiar a una IP pública como muestra la imagen 22. Las direcciones públicas son las que identifican a los dispositivos que están en internet. Se debe habilitar el autoasignado de la IP pública para garantizar que esa máquina va a tener una dirección IP pública y que además se podrá ver de internet, pero también necesita de una protección en este caso un firewall que en AWS se llama Security Group (grupo de seguridad).

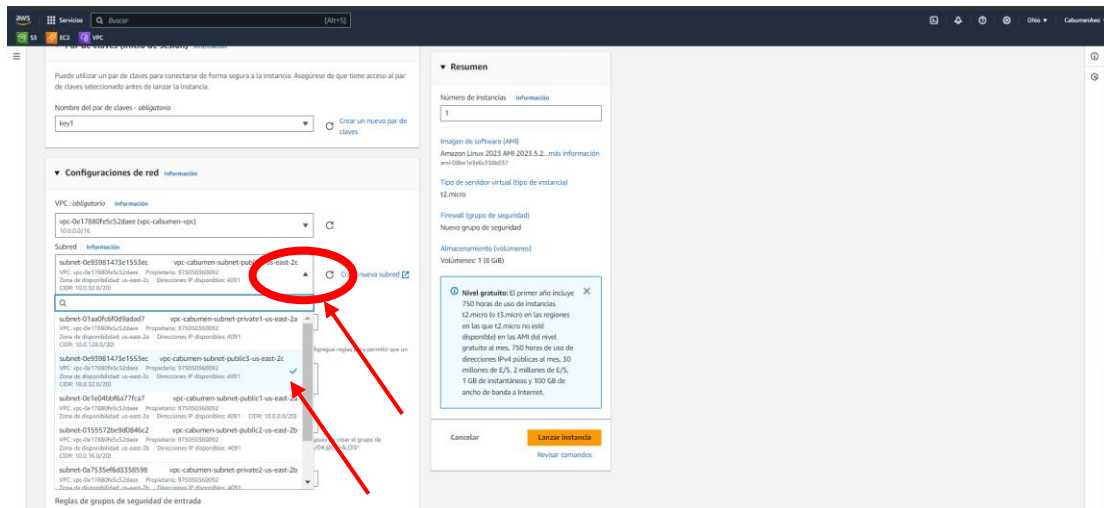


Imagen 22

De esta manera se crea un Firewall o Security Group, el nombre debe ser fácil de identificar, como se muestra en la *imagen23*.

Para este caso va a ser un servicio que permite el acceso de entrada o salida de peticiones hacia el servidor y de eso se encarga el firewall.

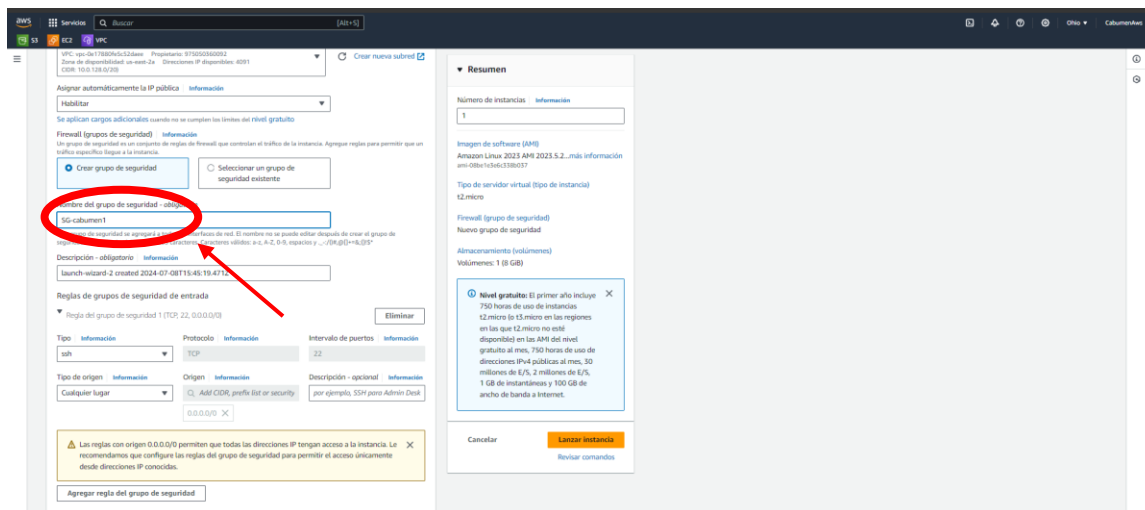


Imagen 23

El firewall va a tener unas reglas, aparecen inbound unas reglas de entrada, la conexión que puede entrarle a esa máquina virtual, como en este ejercicio tenemos un sistema operativo Windows se debe administrar por un protocolo rdp, como nos muestra la *imagen24* esto es una pantalla remota (interfaz gráfica) que se abre hacia el servidor.

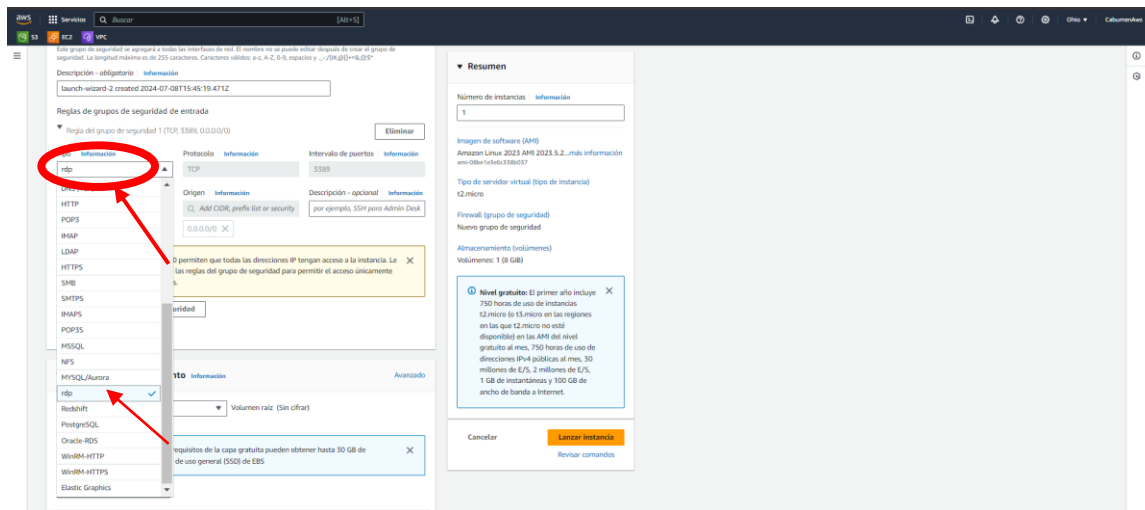


Imagen 24

Ahora viene la configuración del disco duro o configuración de almacenamiento. Que disco duro se la va a poner y de que tamaño, dependiendo de lo que se quiere hacer, hay un requerimiento mínimo y depende del sistema operativo porque el sistema operativo va a gastar una parte del disco. Por defecto se deja la capa gratuita que pueden obtener hasta 30 GB de almacenamiento magnético o de uso general (SSD) de EBS, tal como se muestra en la *imagen25*.

Cuando se haya terminado toda la configuración le damos lanzar instancia en la parte inferior derecha como señala la imagen.

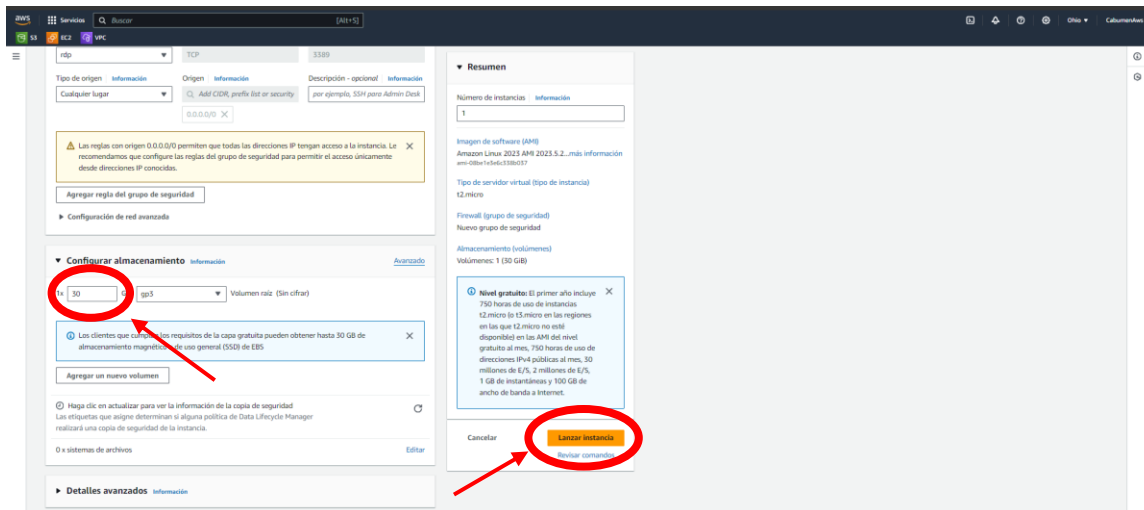


Imagen 25

Una vez se haya lanzado la instancia debe aparecer el siguiente cuadro tal cual muestra la imagen26.

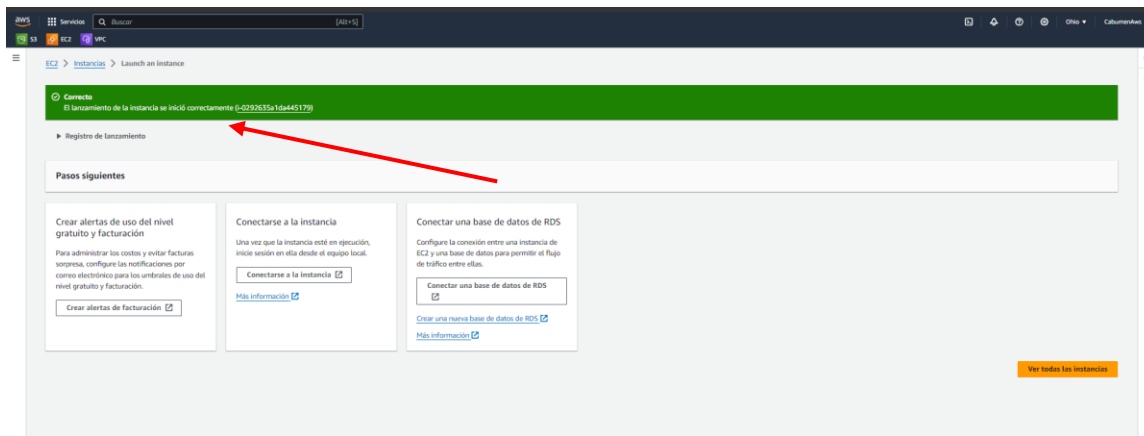


Imagen 26

Con esto podemos decir que nuestra instancia fue creada correctamente y está disponible un servidor virtual para utilizar con esas características.

Instalar un Docker es como una pequeña caja y esa caja es un proceso dentro del sistema operativo puede ejecutar muchos programas o un programa que al ejecutarse se llaman procesos.

Seleccionamos lanzar instancia parte superior derecha como muestra la *imagen27*

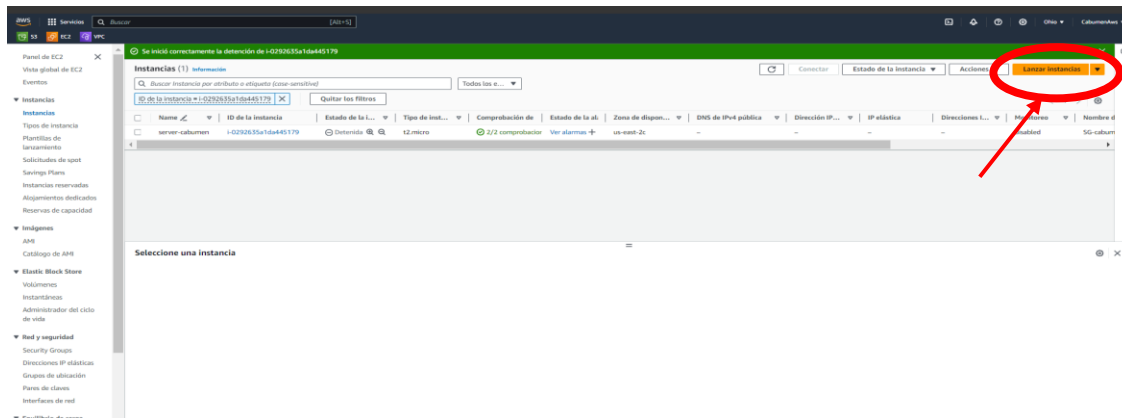


Imagen 27

Le colocamos un nombre y seleccionamos un sistema operativo, para este caso vamos a trabajar con Amazon Linux.

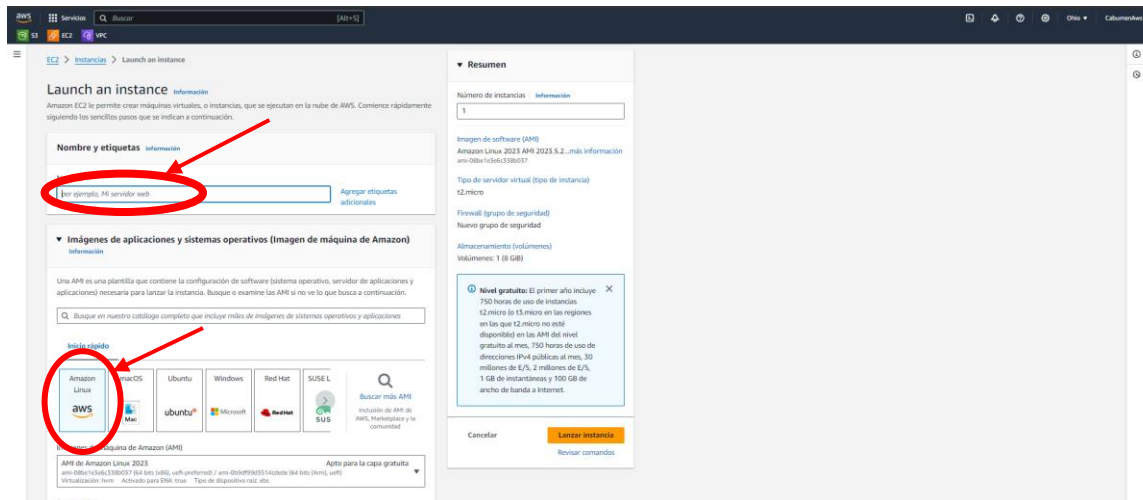


Imagen 28

Para el tipo de instancia seleccionamos t2.micro que es el apto para la capa gratuita, cuando se está creando la maquina por defecto sala el tipo de instancia, pero se recomienda revisar para más seguridad.

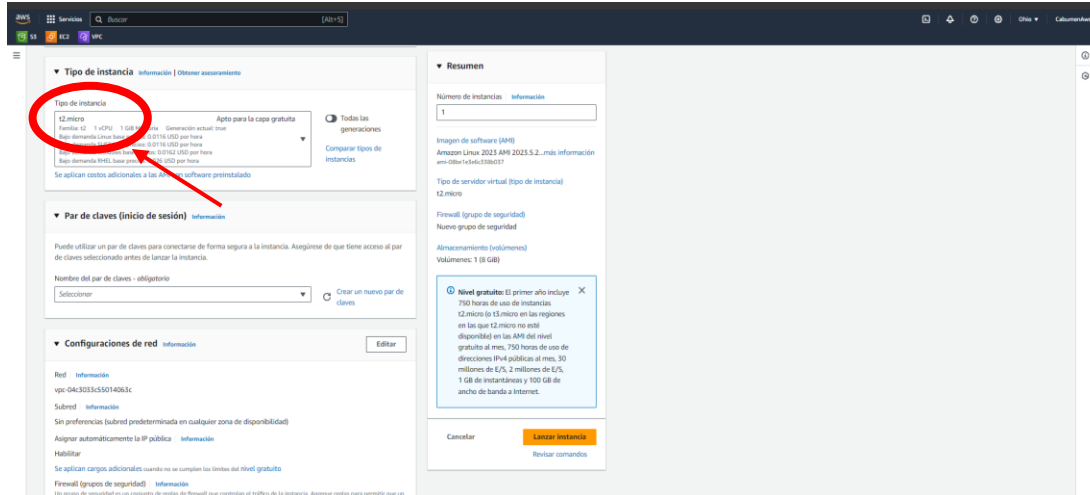


Imagen 29

Seleccionamos la clave que se creó anteriormente en el archivo de descarga.

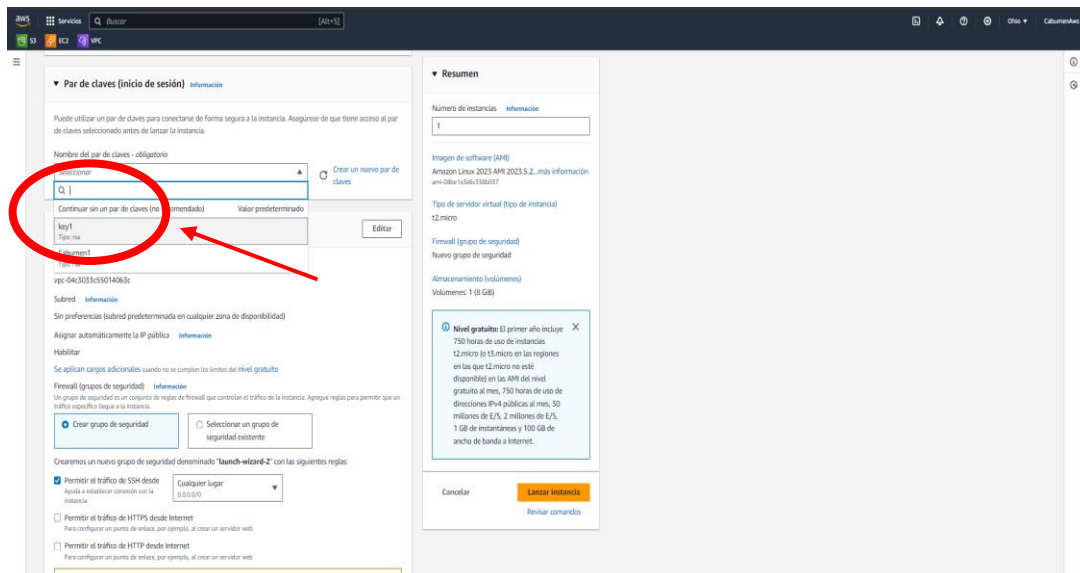


Imagen 30

Configuración de la red, se debe seleccionar la pública para que pueda acceder a través de internet.

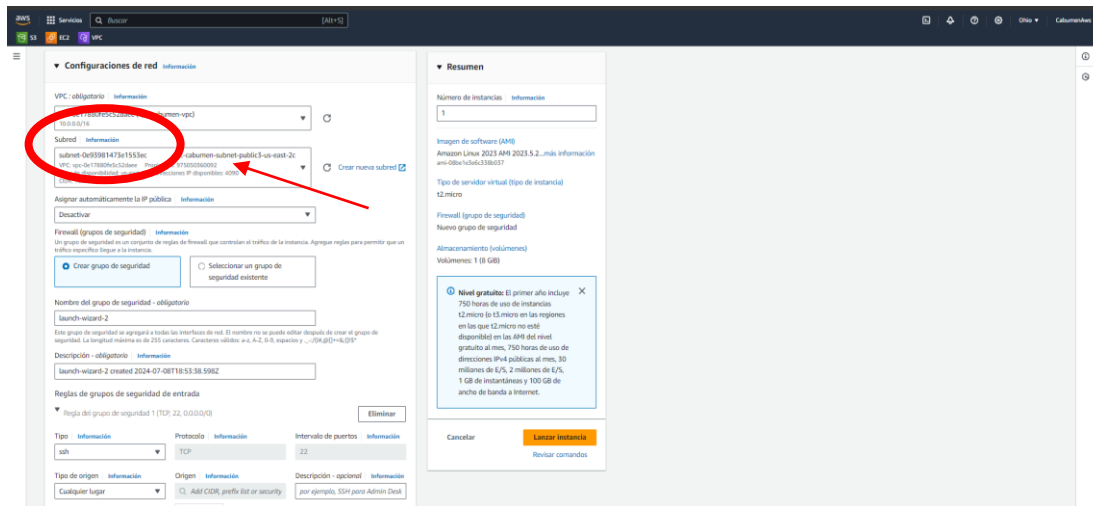


Imagen 31

Acá las reglas cambias con respecto al sistema operativo de Windows, Linux trae una regla que va a permitir ingresar es por el puerto 22. El tipo de ssh va a ser un tipo de consola de comando, ya no se va a usar interfaz gráfica como en Windows.

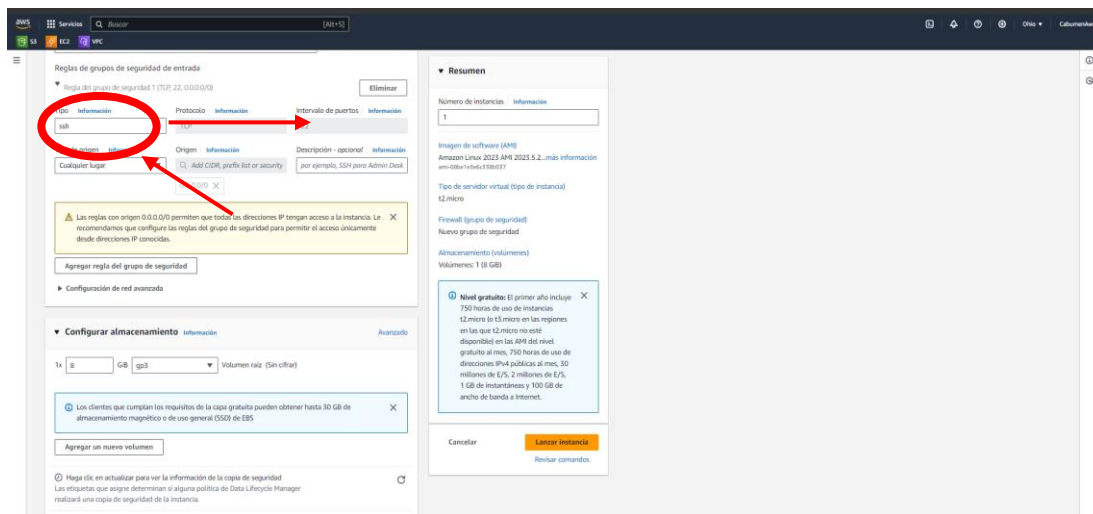


Imagen 32

La configuración de almacenamiento como es de Linux va a necesitar menos espacio se deja así por defecto como aparece en la imagen33, y le damos lanzar instancia en la parte inferior derecha.

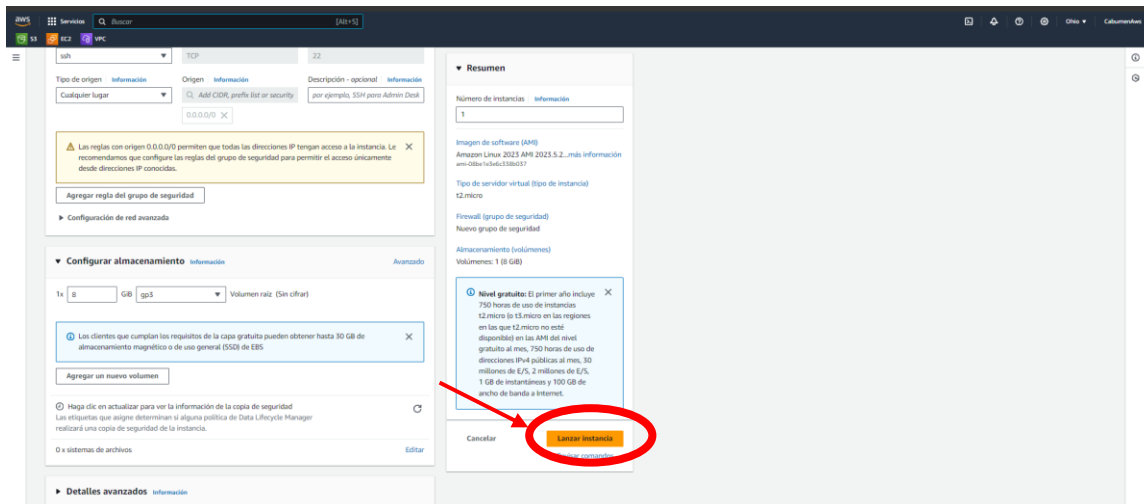


Imagen 33

Completado todos los pasos y verificada la configuración de nuestra instancia, nos debe aparecer el siguiente cuadro con el registro del lanzamiento correcto.

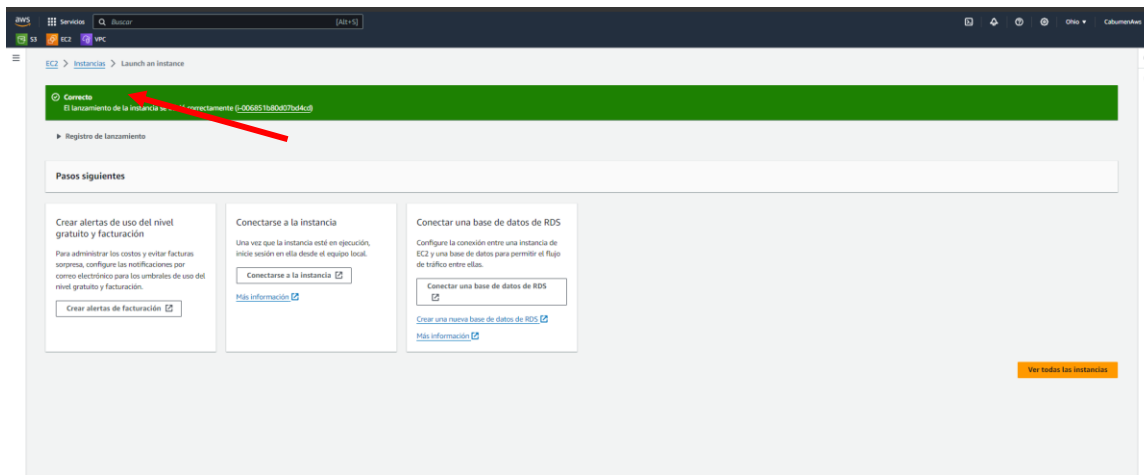


Imagen 34

Vamos a utilizar una aplicación para conectarnos a través del protocolo SSH, en Windows es una aplicación adicional que hay que instalar.

WSL que es una consola de Linux ejecutándose sobre Linux, en este caso es una consola con mandos de Linux, pero sobre Windows.

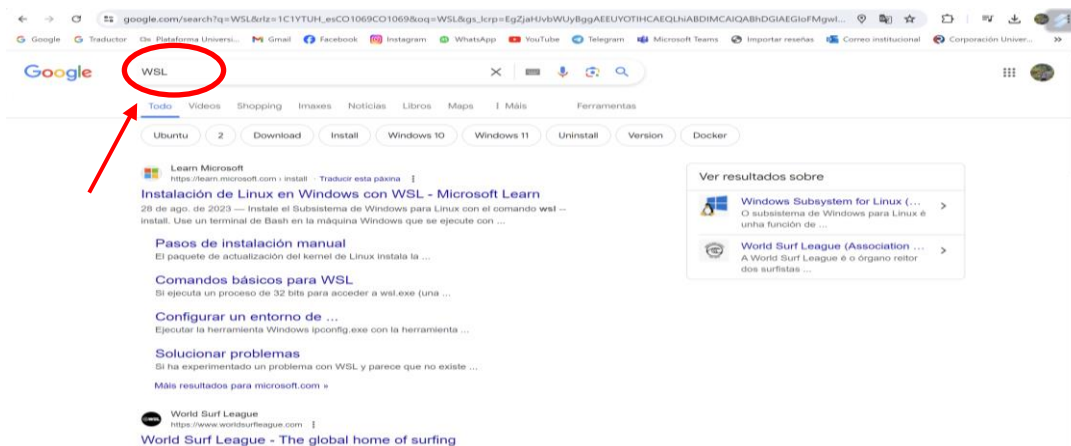
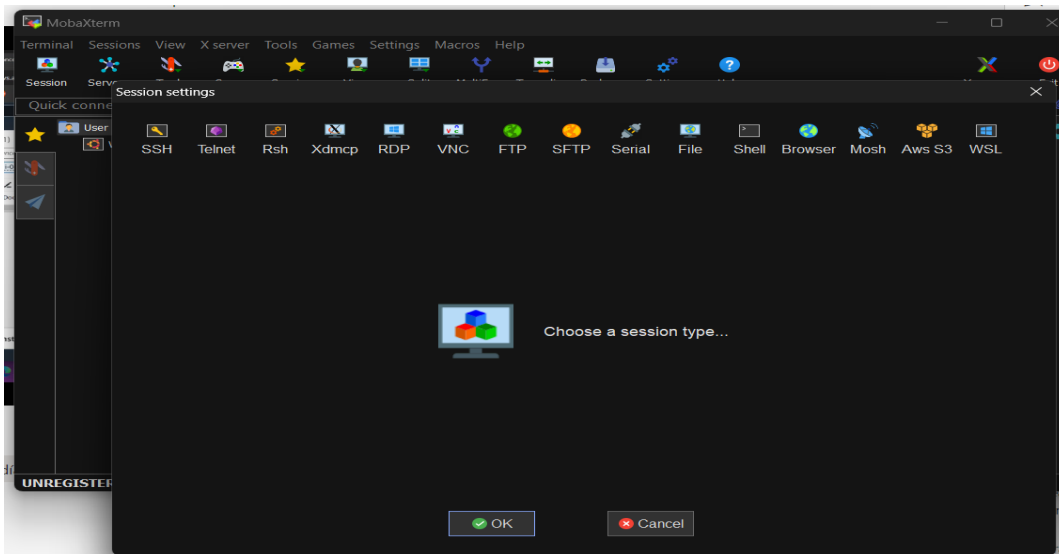


Imagen 35

Otra opción es MobaXterm es más rápida y con esta vamos a conectarnos a Linux por medio de



comandos.

Imagen 36

Para conectarse a la consola debe seleccionar donde dice cliente SSH y en la parte inferior muestra el comando que se debe utilizar para conectar a la consola SSH.

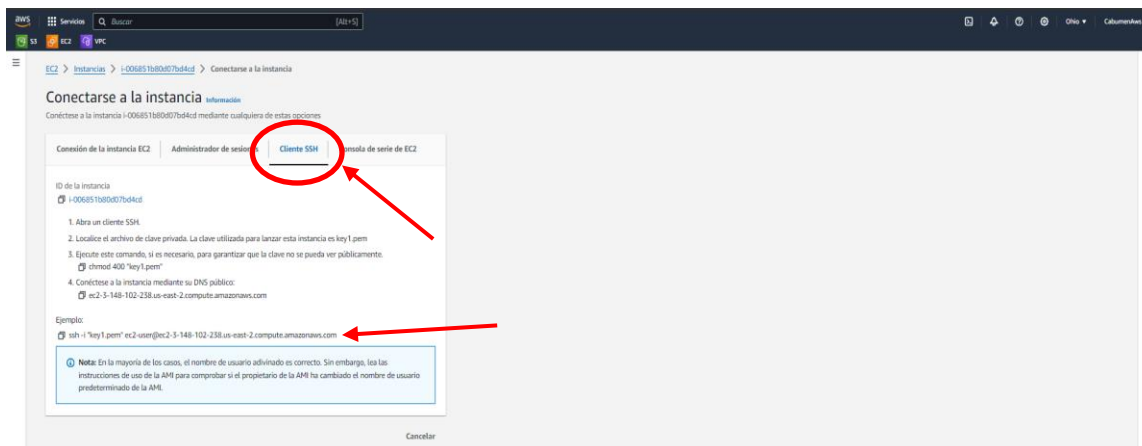


Imagen 37

Copiamos el nombre o la IP cual quiera de las dos, pegamos la copia en Remote host, como muestra la *imagen 38*, se especifica el nombre de usuario habilitando con un chulito donde dice Specify username. Para Amazon Linux siempre va a ser *ec2-user*.

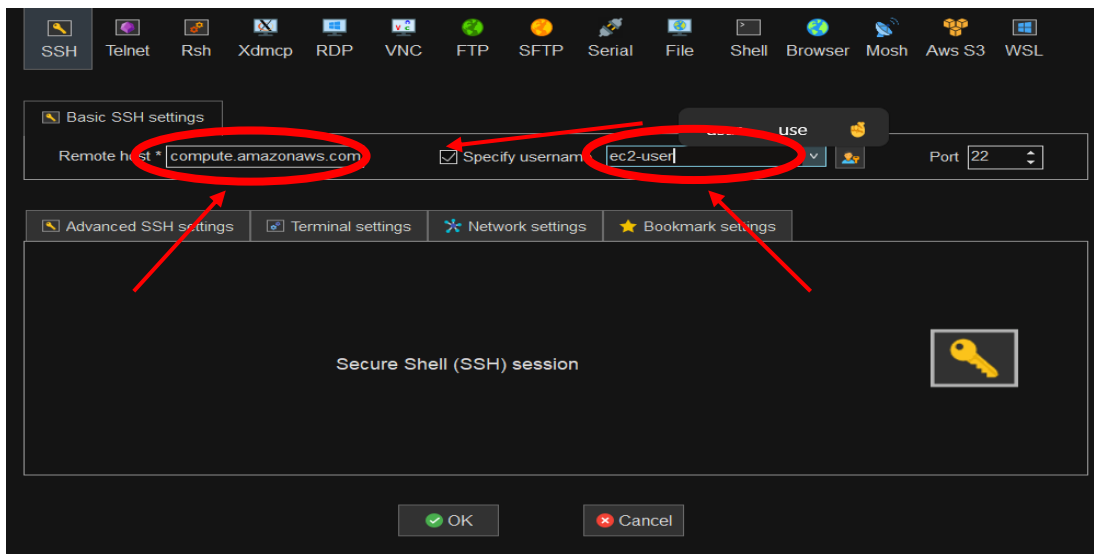


Imagen 38

Completado el paso anterior nos muestra esta nueva venta, se habilita la clave privada y se descarga el archivo de certificado de seguridad.

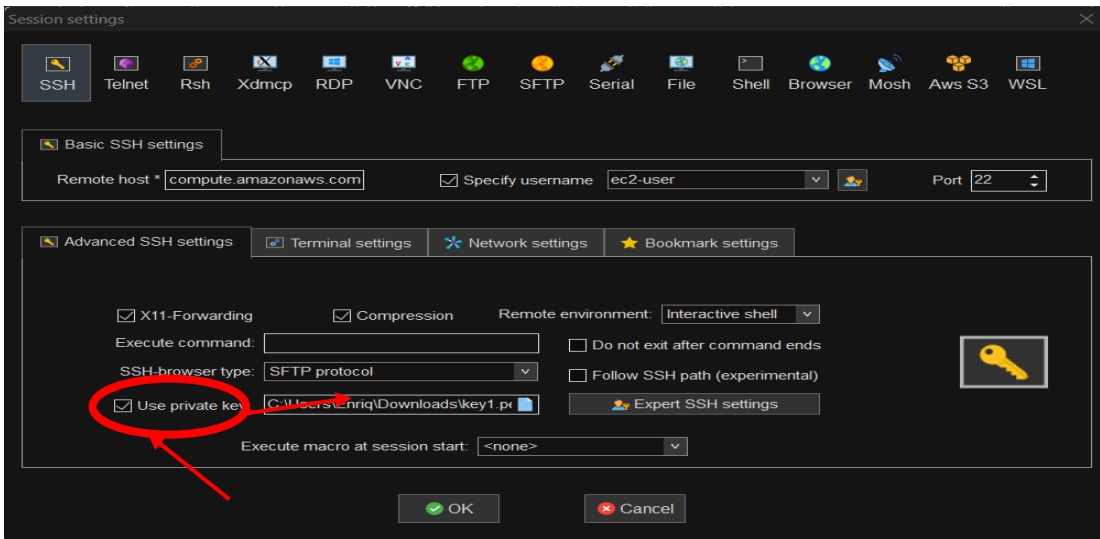


Imagen 39

Ya conectado debe aparecer el siguiente cuadro de la imagen40. 3 parámetros que se necesitan, nombre IP, el usuario y el certificado de seguridad.

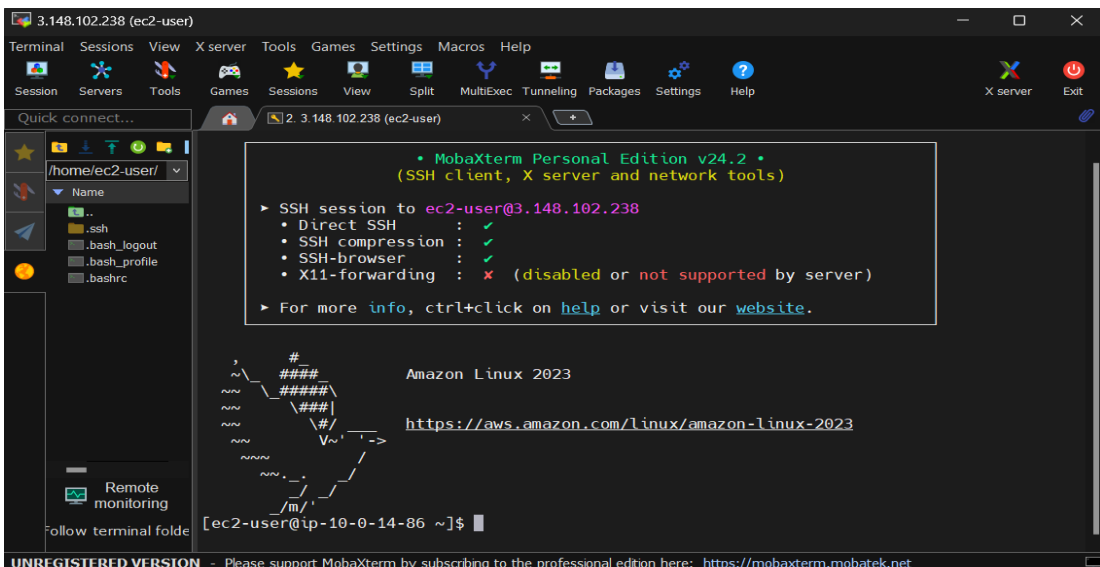


Imagen 40

Una vez ya estemos dentro de Linux, debemos logearnos como administrador usando el comando `sudo su`. Aquí ya estamos en modo root o administrador.

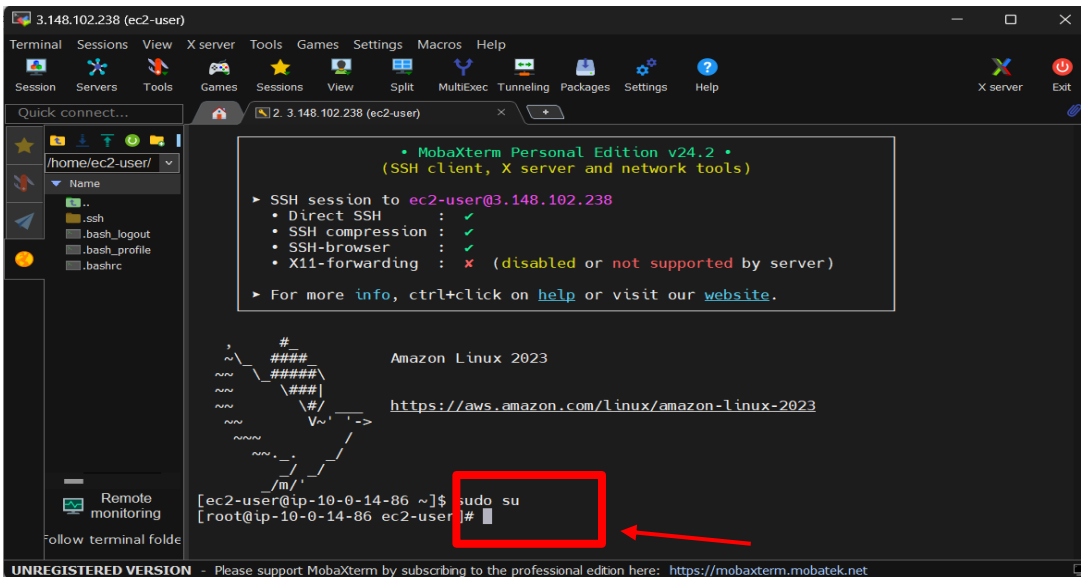


Imagen 41

Con el comando `yum install Docker`. Se instala toda la plataforma de Docker sobre Linux.

Debe colocar (Y) para que se instale completamente.

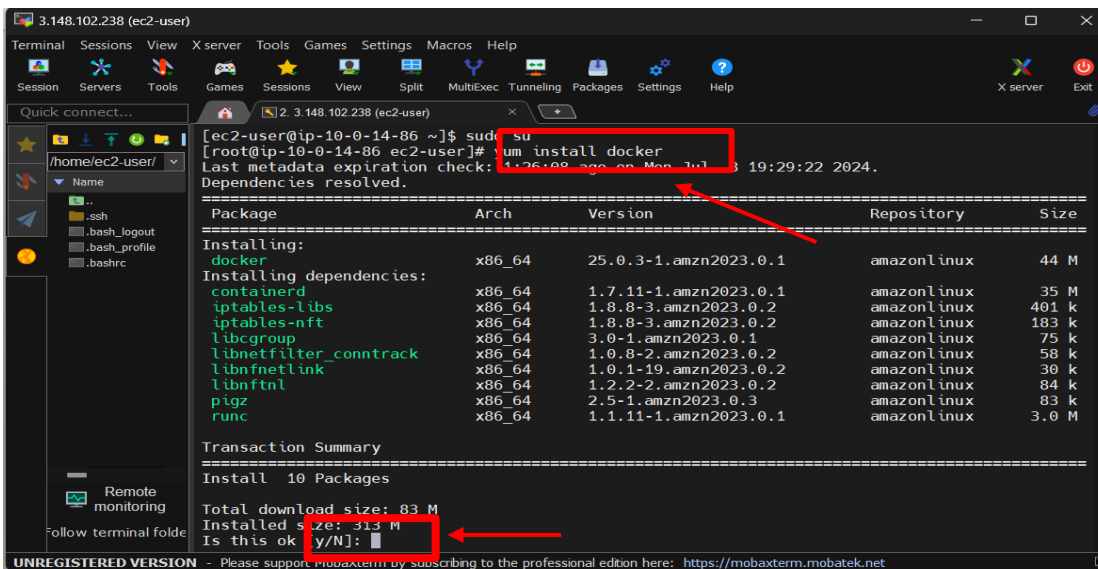


Imagen 42

Después de la instalación así debe mostrar la consola con el comando `Complete`.

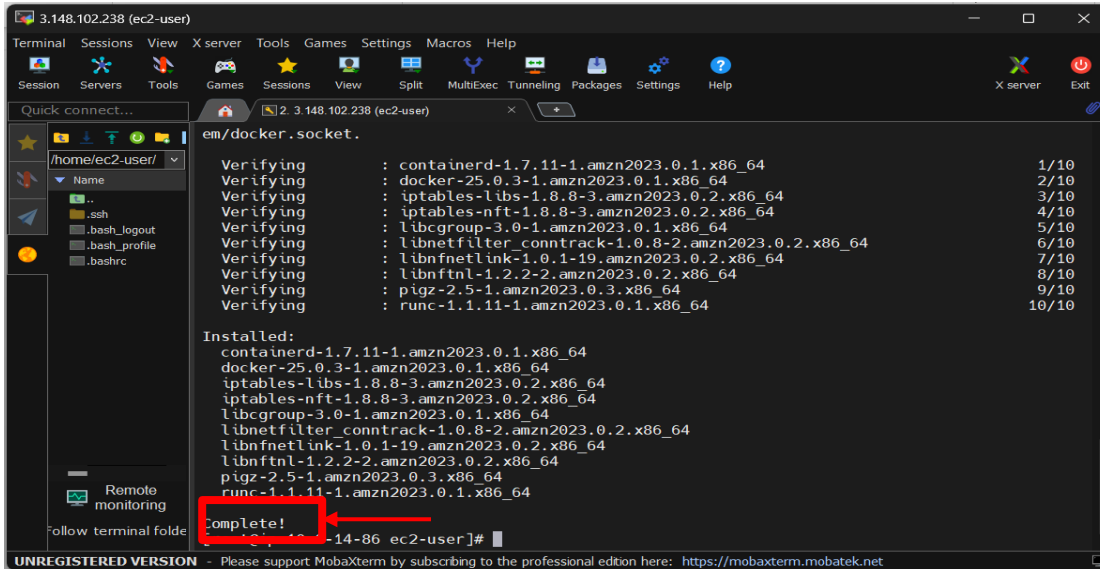


Imagen 43

Vamos a montar una aplicación básica, en este caso un servidor web con un apache, estas imágenes las encontramos en un repositorio que se llama Docker hub.

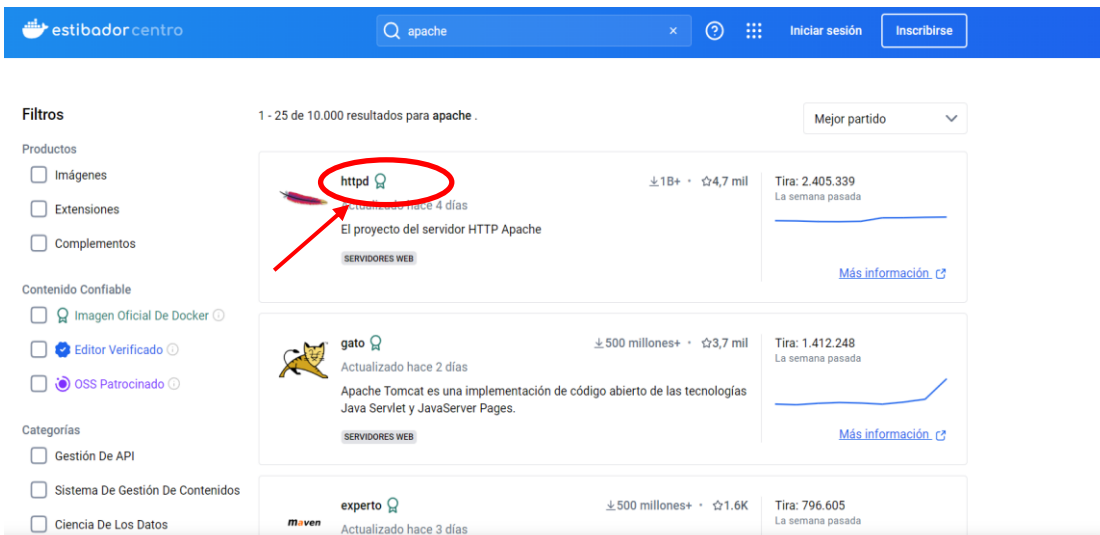


Imagen 44

Copiamos el código de la aplicación que muestra en la parte superior derecha y lo vamos a pegar en la consola de Linux.

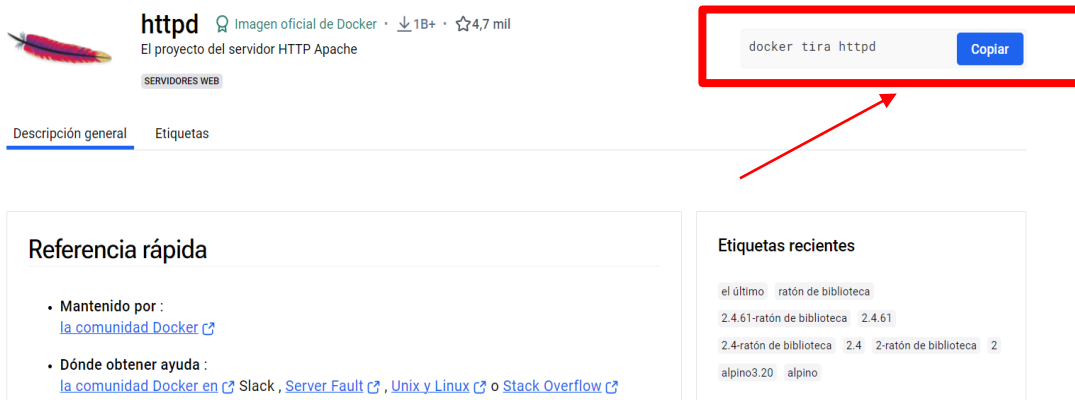


Imagen 45

En la consola vamos a ejecutar el comando Docker pull httpd para descargar nuestra imagen.

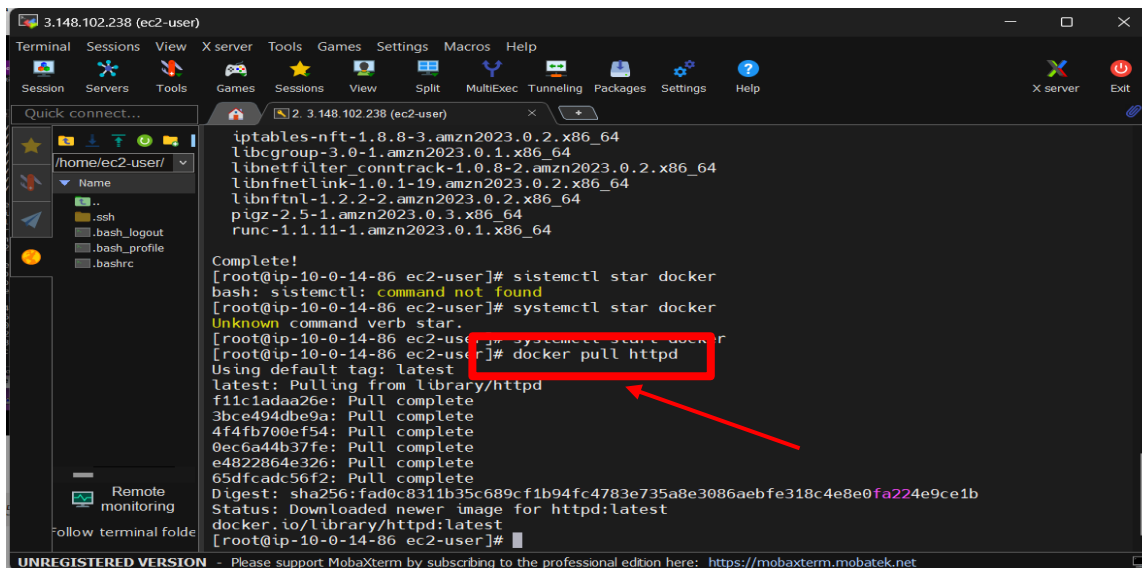


Imagen 46

Se crea una carpeta con los comandos `mkdir /tmp/sitio1` y el otro comando es `Nano /tmp/sitio1/index.html` para que el navegador no este vacío. Y debe mostrar esta ventana.

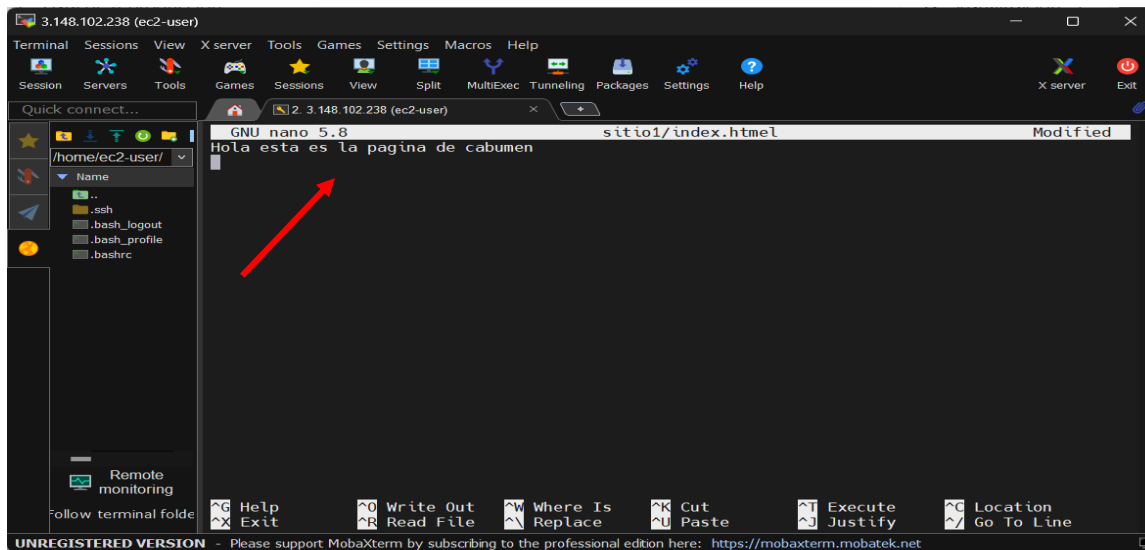


Imagen 47

Para crear el contenedor usaremos el comando `sudo Docker run -dit --name contenedor -p 8080:80 -v /tmp/sitio1/:/user/local/apache2/htdocs/ httpd`

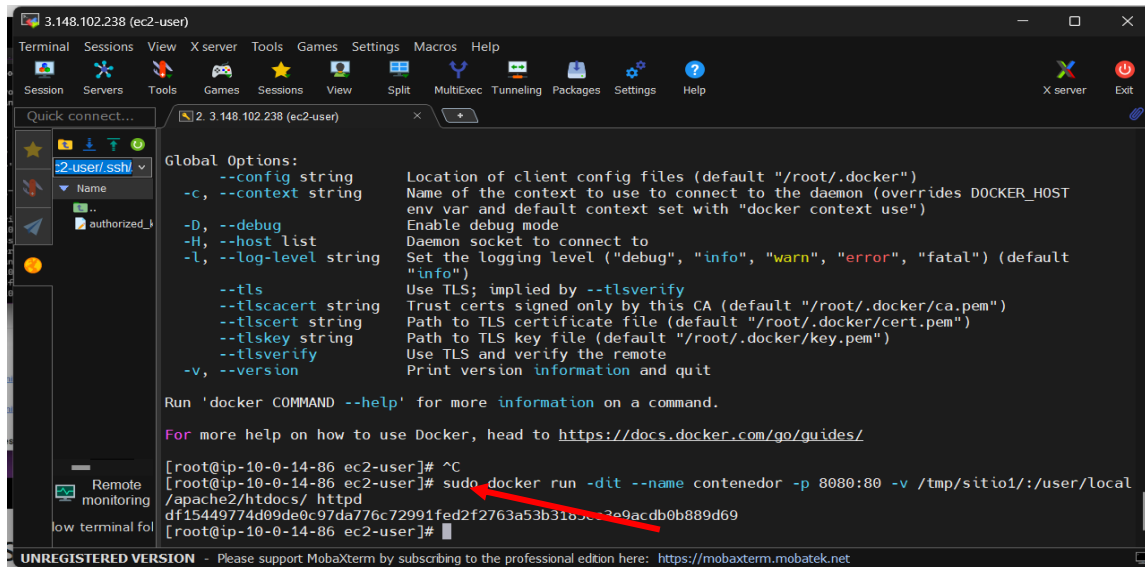


Imagen 48

Se debe agregar una regla nueva que permita conectar por el puerto 8080 entre cualquiera.

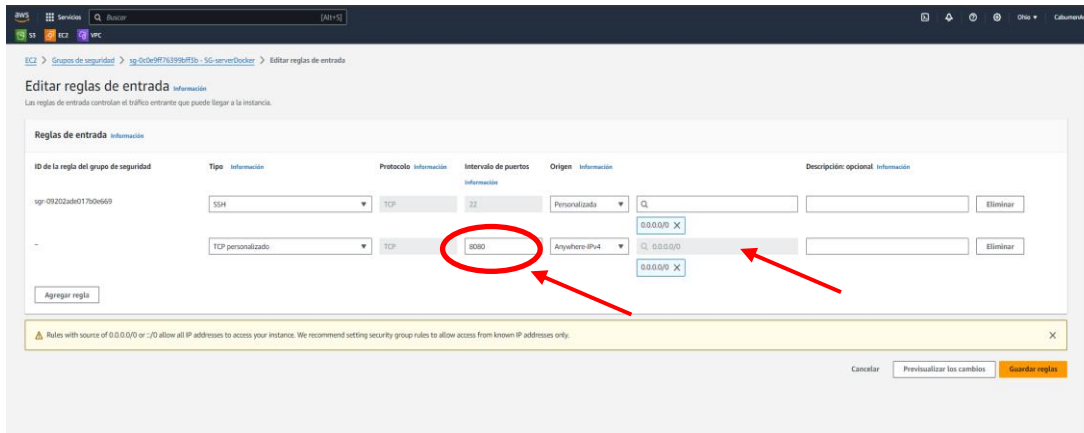
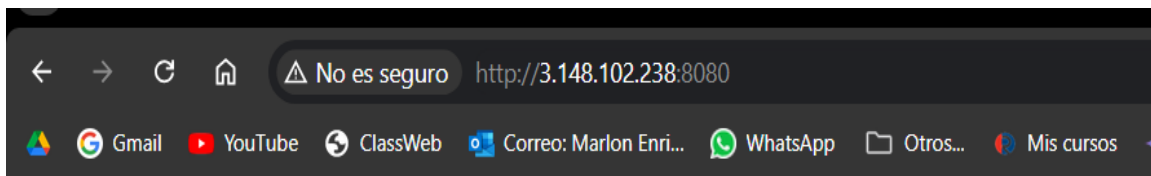


Imagen 49

Podemos ver que nuestro contenedor está corriendo, podemos ver que el puerto 8080 que se creó permite que funcione nuestro HTML.



It works!

Imagen 50

Como ya esta corriendo nuestro contenedeor, lo que hacemos es subir una palntilla de una aplicación para verificar que en verdad si esta cargardo, tal cual lo muestra la imagen con la nueva regla del puesto 8080.

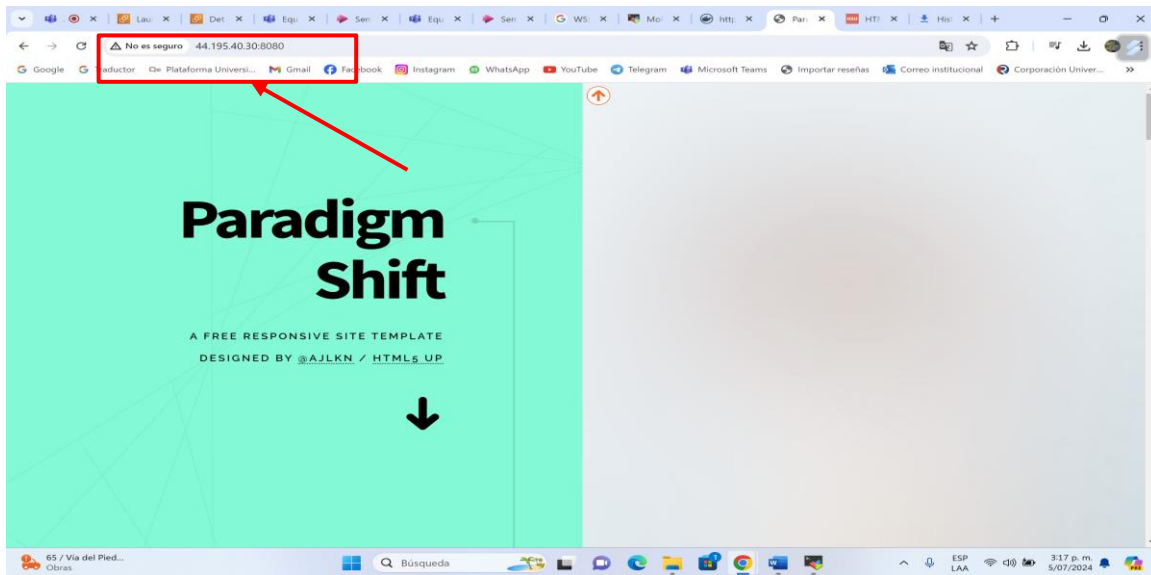


Imagen 51

Creamos el segundo contenedor con el comando `sudo Docker run -dit --name contenedor2 -p 8081:80 -v tmp/sitio2:/user/local/apache2/htdocs/ httpd`

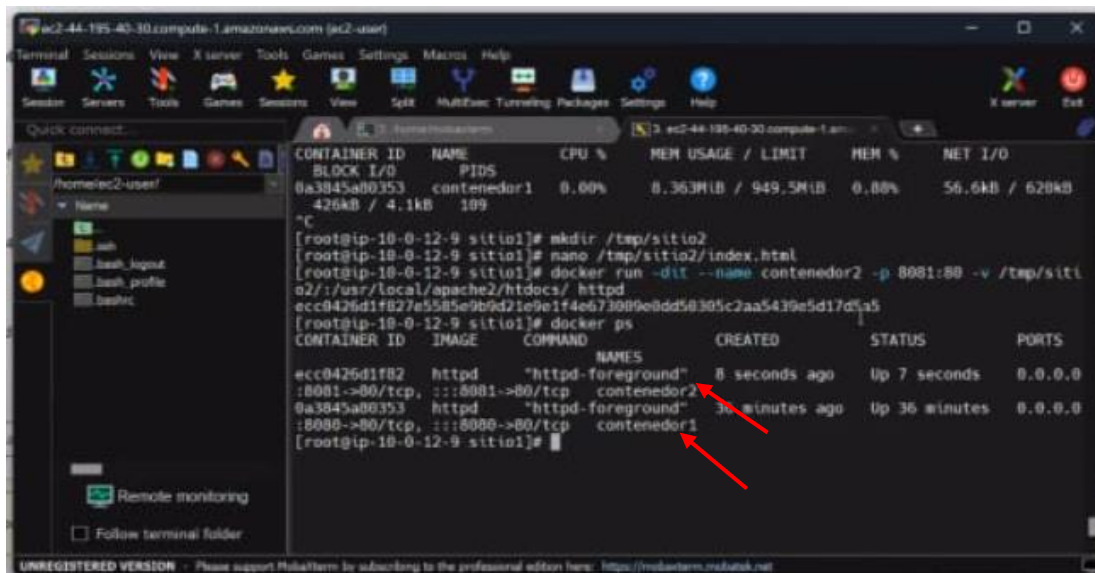


Imagen 52

Se debe agregar una regla nueva que permita conectar por el puerto 8081 entre cualquiera.

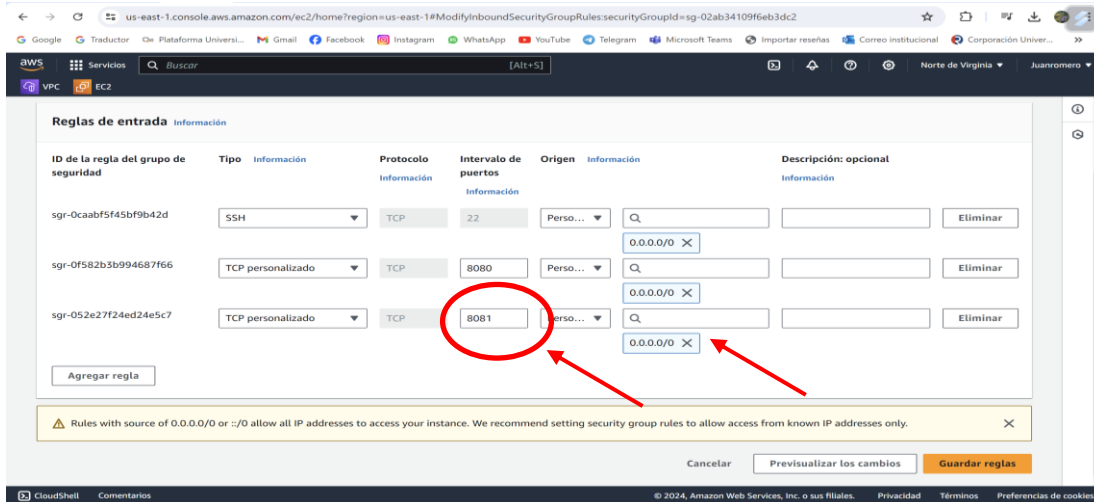


Imagen 53

Podemos ver que nuestro contenedor está corriendo, podemos ver que el puerto 8081 que se creó permite que funcione nuestro HTML.

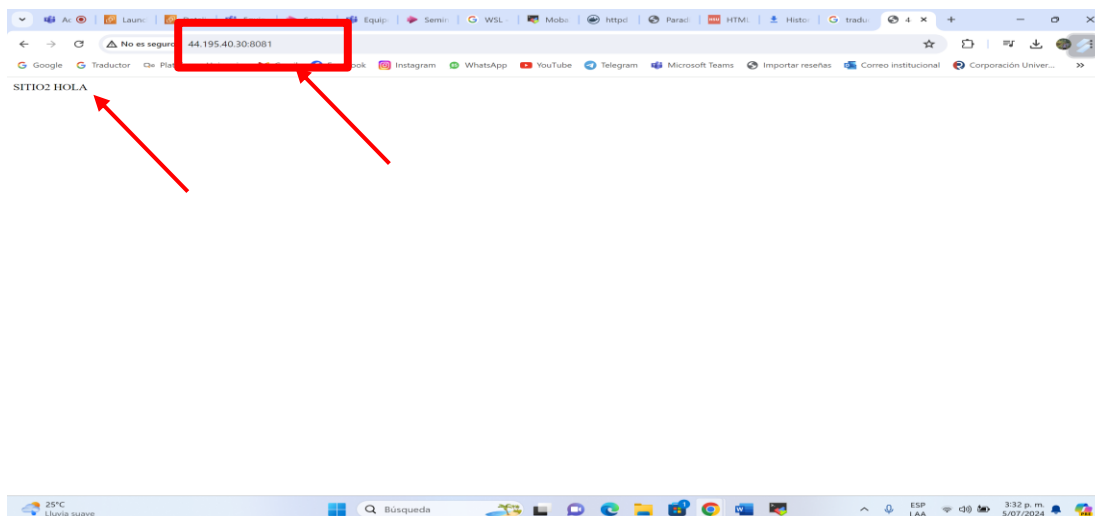


Imagen 54

Como ya esta corriendo nuestro contenedeor, lo que hacemos es subir una segunda palntilla de la misma aplicación para verificar que en verdad si esta cargardo, tal cual lo muestra la *imagen55* con la nueva regla del puesto 8081.

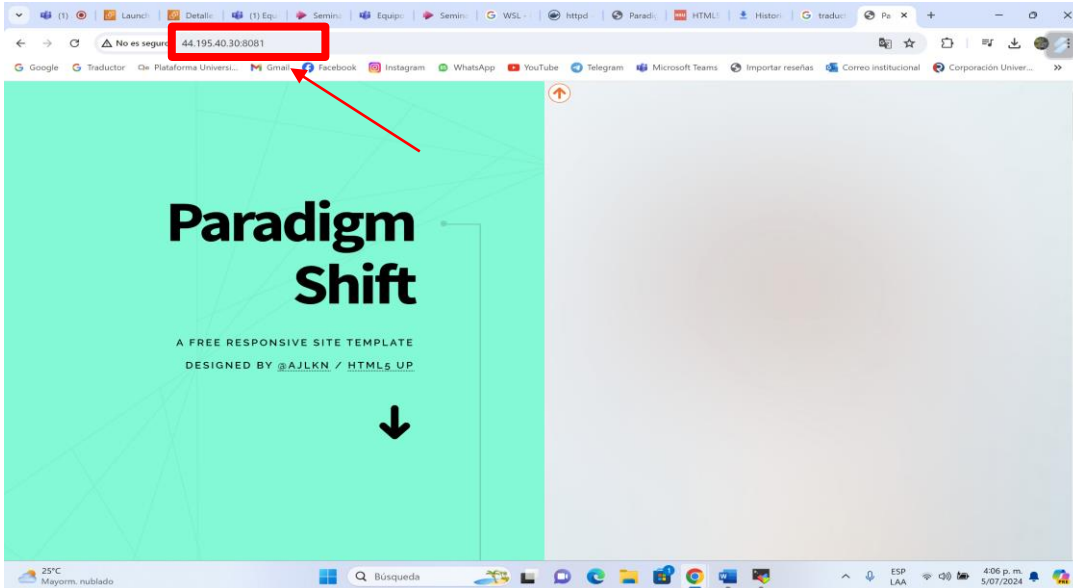


Imagen 55

Para montar un balanceador de carga, se debe instalar el servidor *nginx* dentro de la máquina virtual, usando el comando `dnf install nginx`.

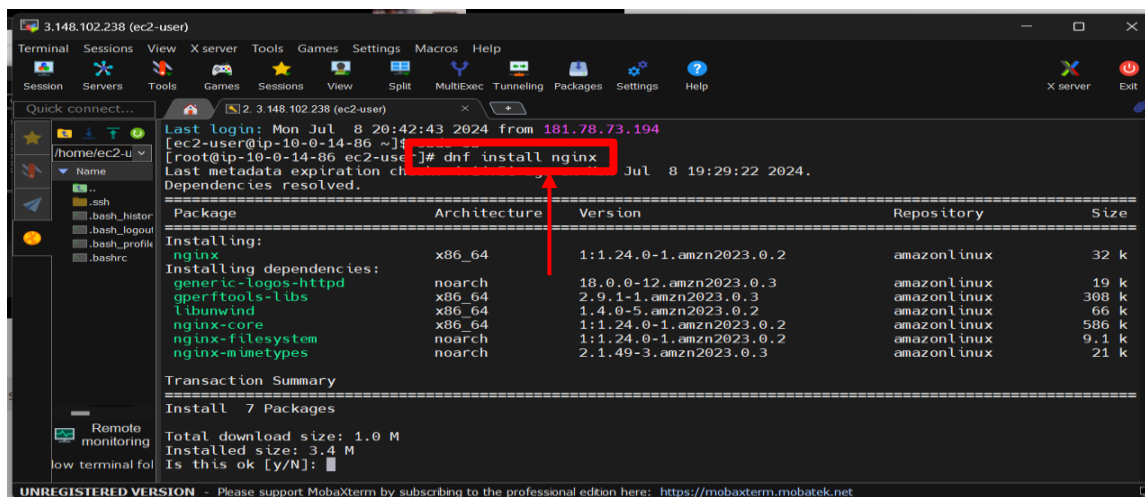


Imagen 56

Para saber si el `ngx` esta funcionando nos debe aparecer este cuadro que por defecto usa el puerto 4030 porque es un servidor web mas dentro de la maquina asi como el apache.

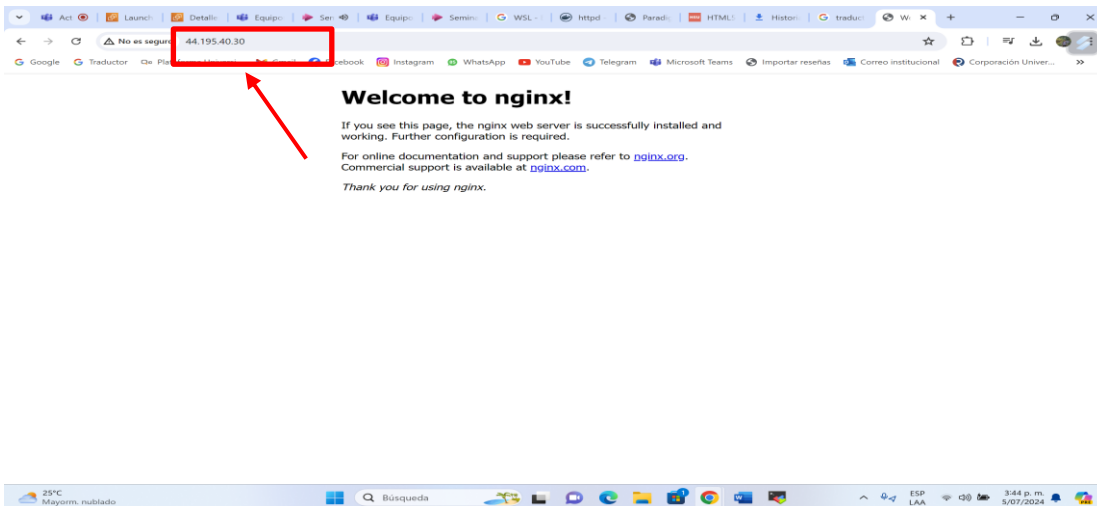


Imagen 57

Para balancear el tráfico se debe hacer una modificación se debe hacer una copia de seguridad por si se llega a dañar el archivo lo podemos recuperar.

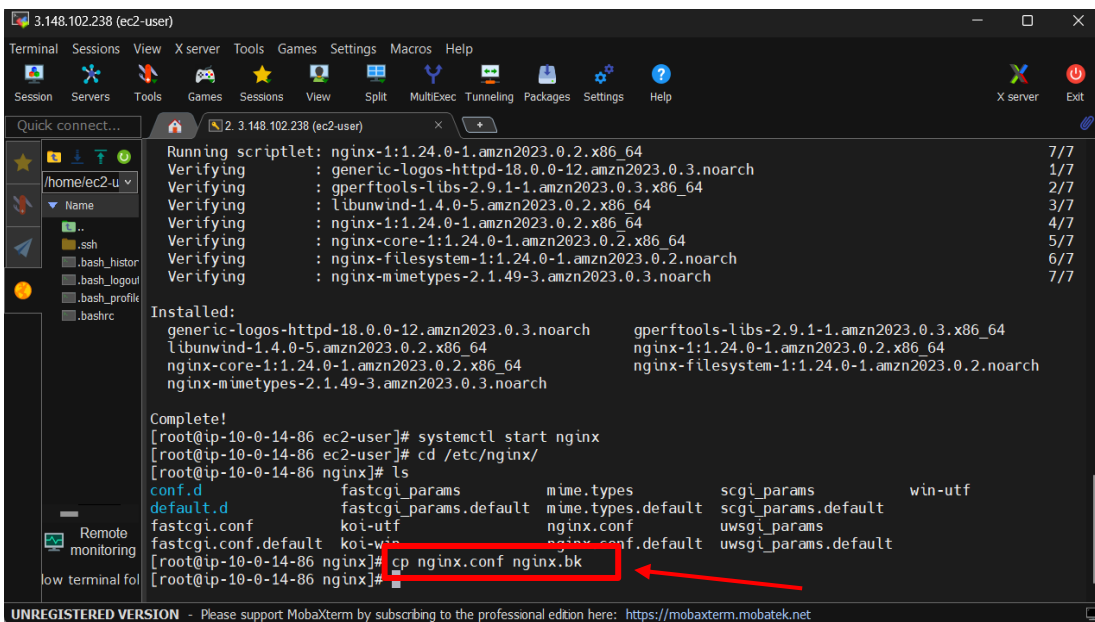


Imagen 58

Lo que muestra la imagen es el código para utilizar para que el nginx pueda funcionar y hacer el balanceo de cargas.

```

events {}

http {
    upstream backend {
        server localhost:8080;
        server localhost:8081;
    }

    server {
        listen 80;
        server_name nginx;
        location / {
            proxy_pass http://backend;
        }
    }
}

```

Imagen 59

Cuando se ingrese a la página debe cargar y mostrar lo que está en el contenedor1

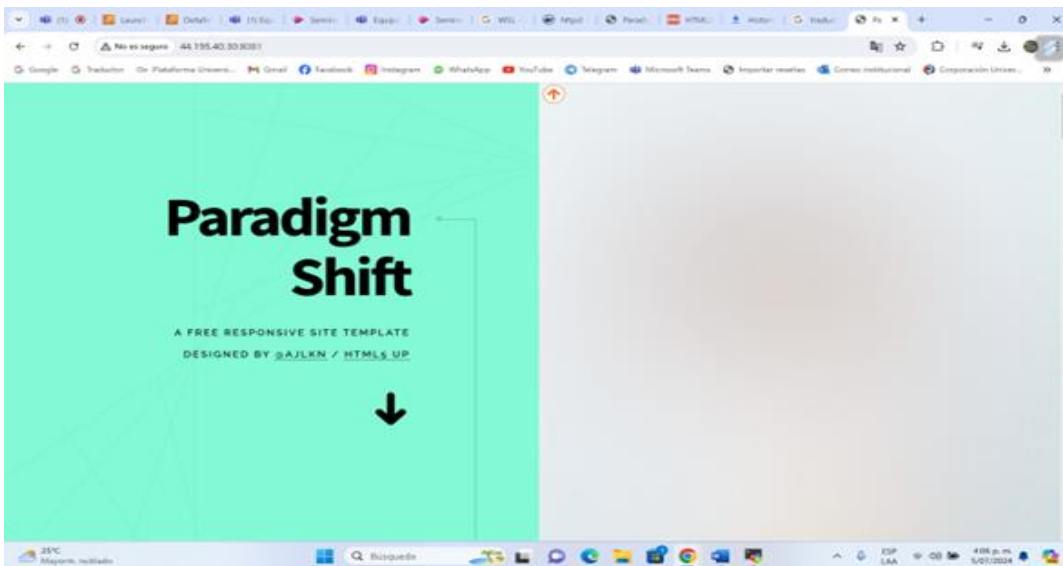


Imagen 60

Si le damos otra orden y actualizamos él debe mostrar lo que está en el cotenedor2.

De esta manera podemos diferenciar y comprobar que el balanceador de carga está funcionando de la mejor manera.

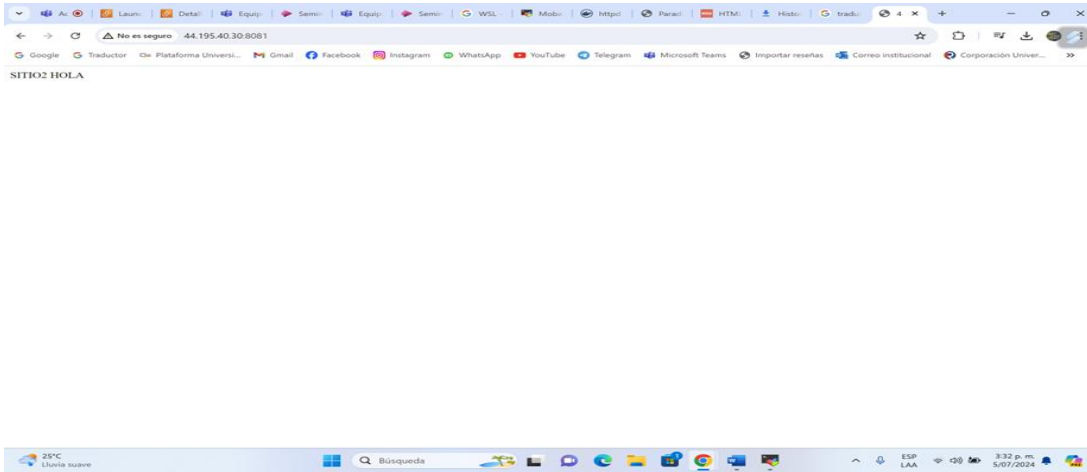


Imagen 61

De esta manera podemos comprobar que nuestra máquina virtual con 2 contenedores y un sistema de balanceo de cargas está funcionando correctamente, se presentaron algunos errores en los comandos, pero fue por mala digitación.

2.1 Sitio estático en S3.

Implementación de un sitio estático en S3, el mismo que usó en los contenedores.

Pantalla inicial de AWS para trabajar con S3, es un servicio muy sencillo que se utiliza para almacenar objetos (archivos, fotos, videos...)



Imagen 62

En el buscador de nuestra pantalla inicial vamos a buscar S3 y de esta forma se despliega una lista de opciones y seleccionamos la que vamos a usar en este caso S3 como lo muestra la imagen63

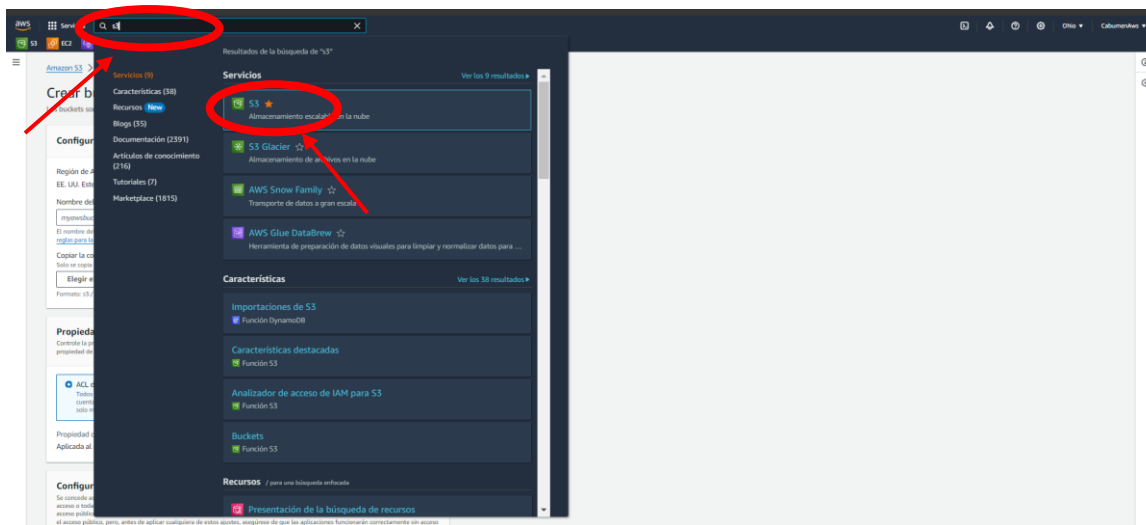


Imagen 63

Al momento de seleccionar S3, se va a desprender el siguiente cuadro que muestra la *imagen64*, y se va a la opción de crear Buckets.

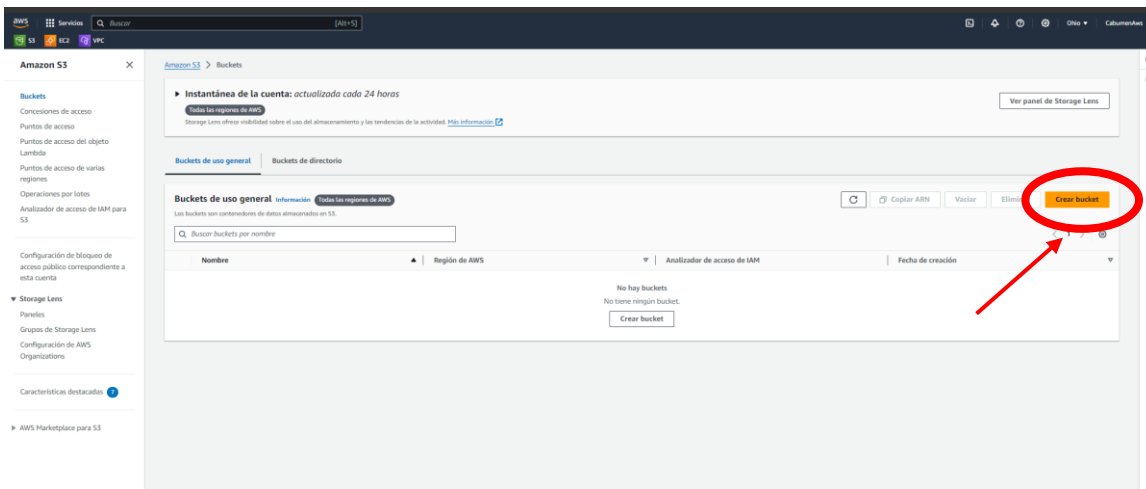


Imagen 64

El Buckets nos va a mostrar el siguiente cuadro, se le debe colocar un nombre, hay que tener en cuenta que este nombre debe ser único en ningún momento se debe repetir.

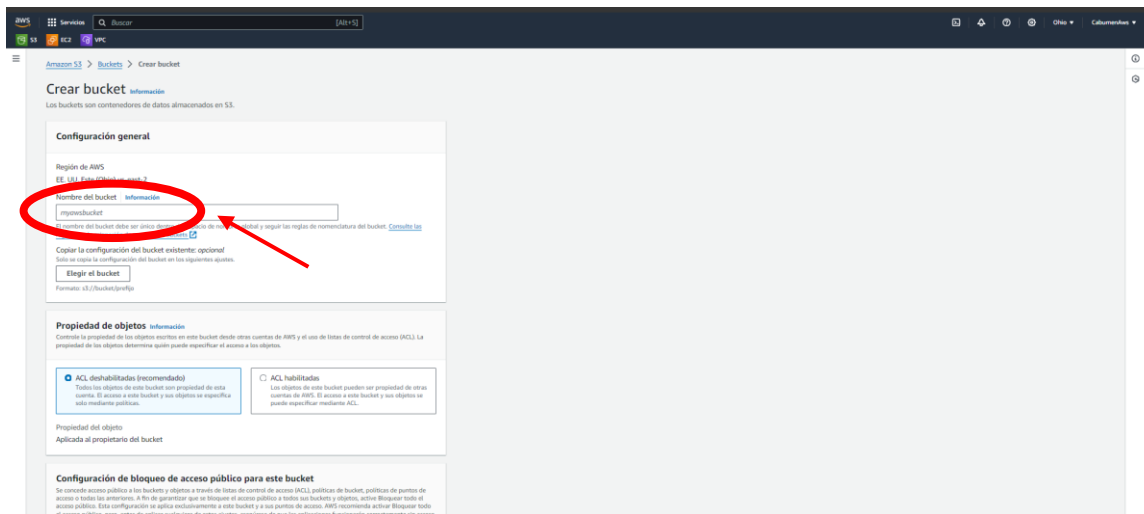


Imagen 65

Los Bockets tiene algo que se llama ACL (lista de control de acceso) que por defecto vienen deshabilitada para limitar quien pueda ver la información que hay en los Bockets y para este ejercicio toca habilitarlas.

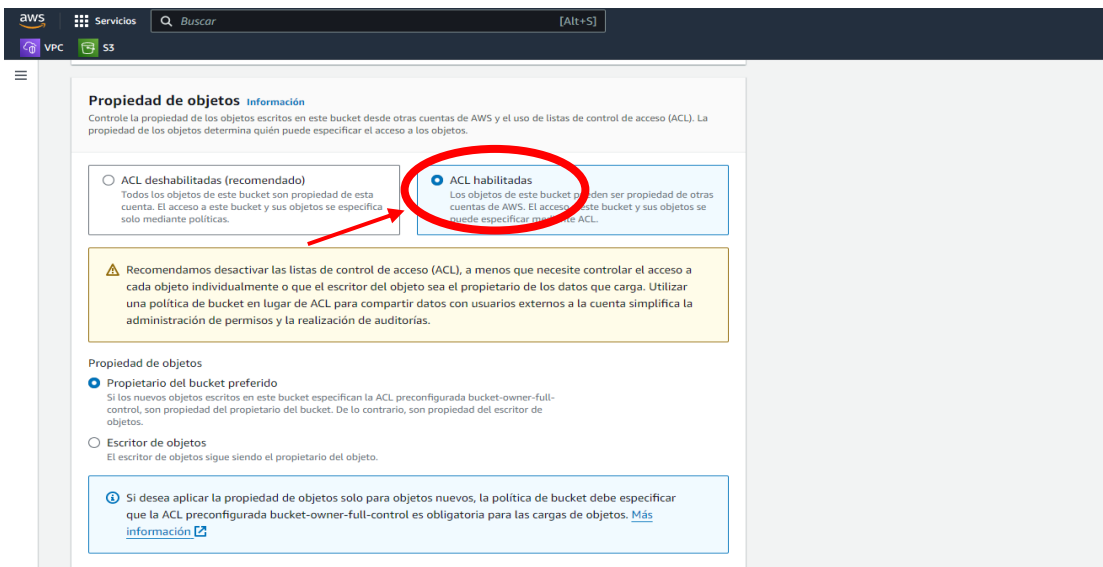


Imagen 66

La configuración de los Bockets por defecto es privada, se deja tal cual esta como muestra la imagen67.

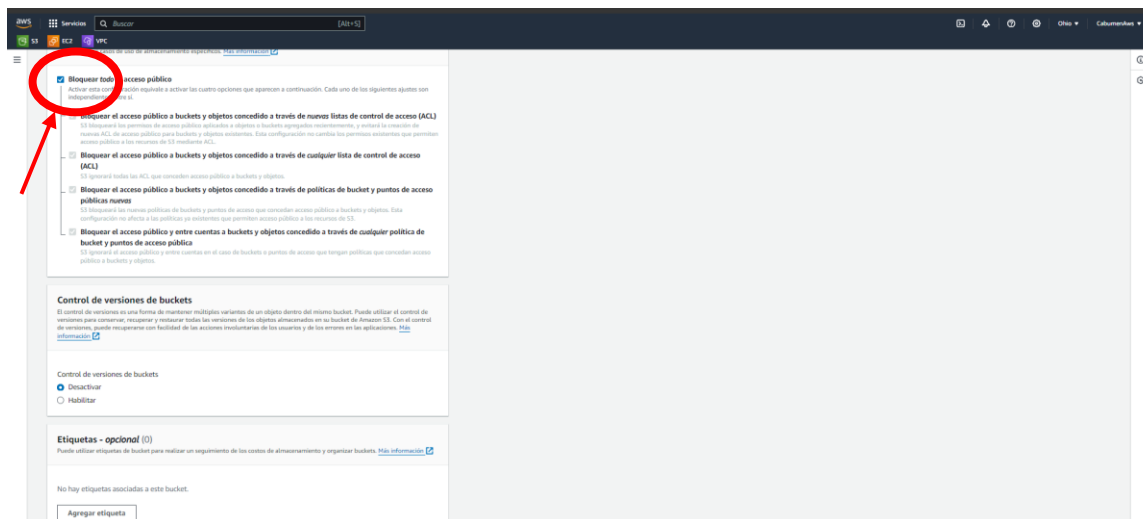


Imagen 67

Los Bockets pueden habilitar o deshabilitar la versión del archivo que en algún momento se le haya hecho una modificación, si está habilitada guarda el archivo con el mismo nombre y la versión más reciente.

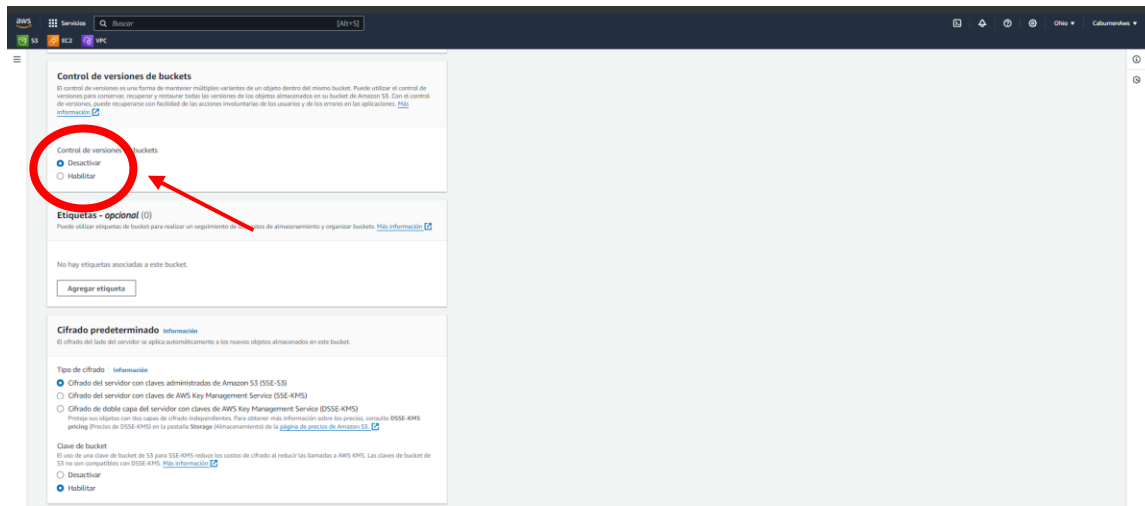


Imagen 68

Los objetos se cifran por defecto de AWS, todo lo que se guarde en los Bockets queda cifrado. Ya terminada la configuración del Bockets le damos crear.

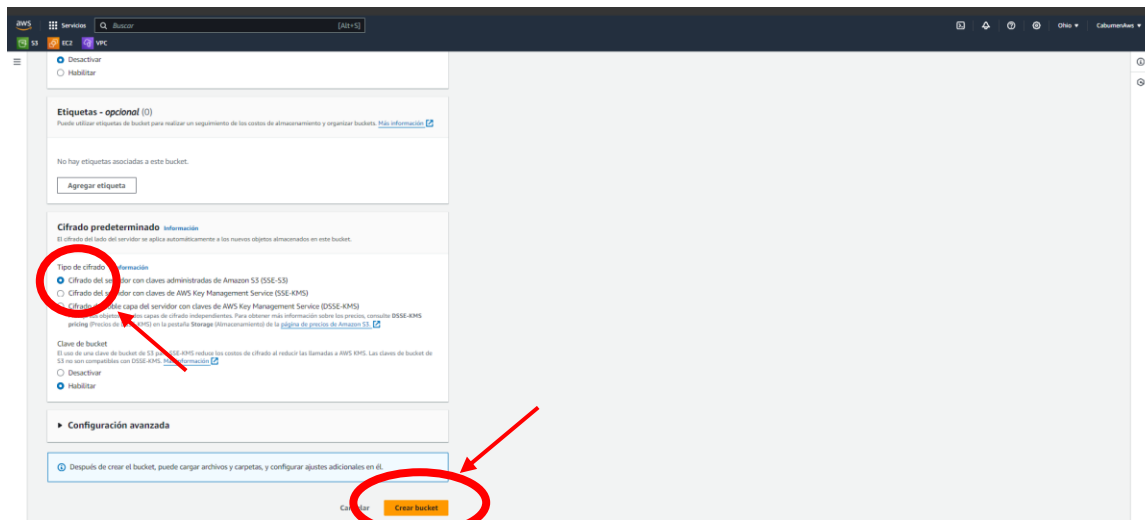


Imagen 69

Si se cumplió a satisfacción toda la configuración nos debe mostrar el siguiente cuadro con el Bockets creado.

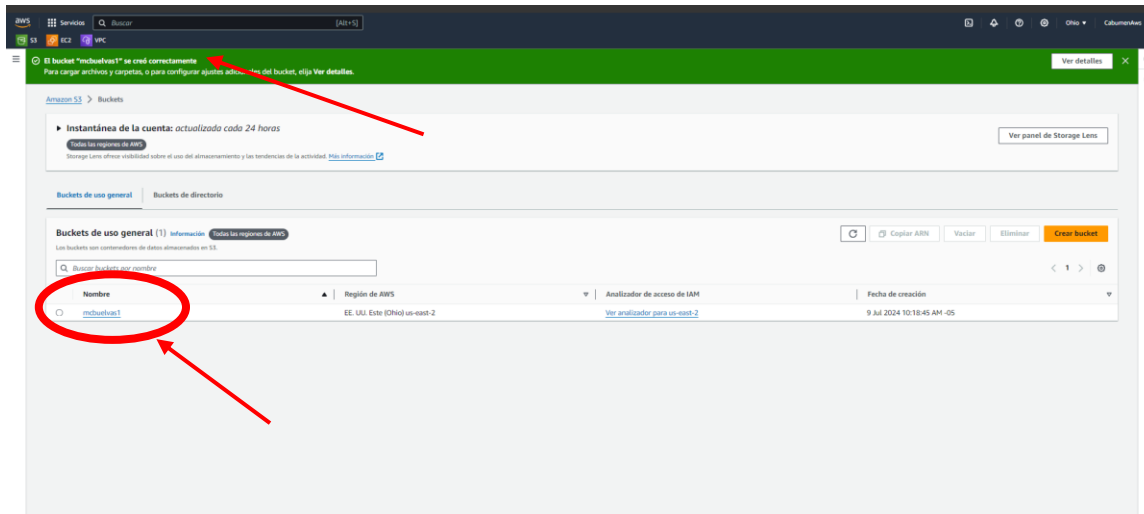


Imagen 70

Al ingresar al Bockets que se creó, nos debe mostrar el siguiente cuadro que es un asistente para cargar información y configuración.

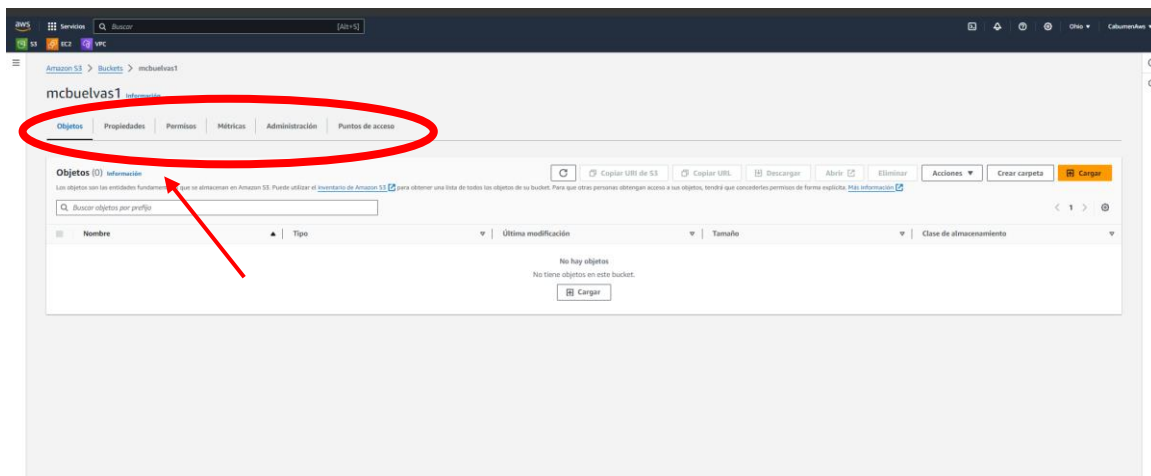


Imagen 71

Acá podemos cambias la versión y guardamos los cambios. Lo contrario a lo que habíamos hecho en la *imagen68*.

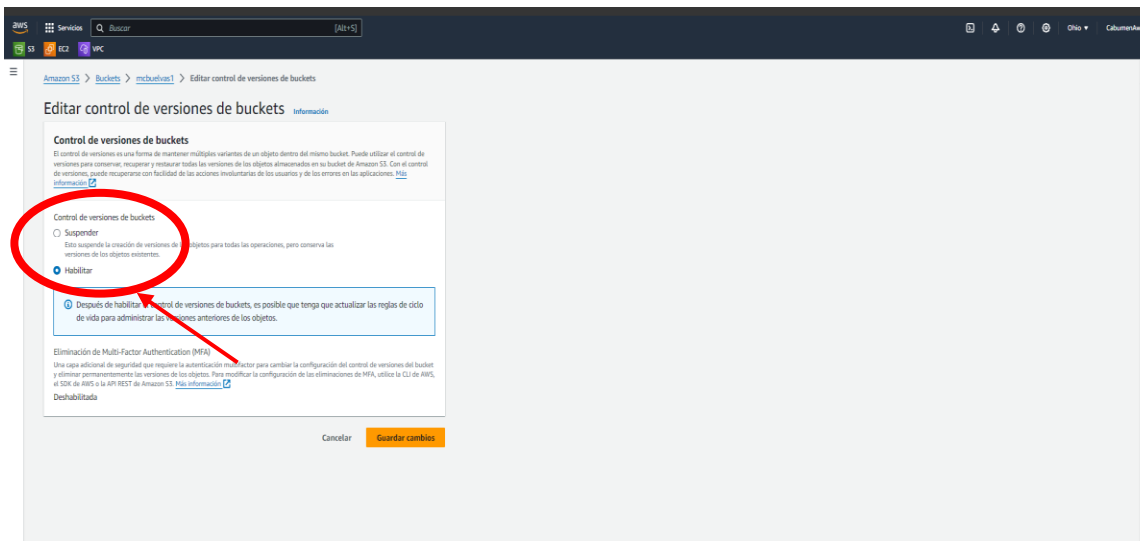


Imagen 72

Desde esta pantalla inicial del bockets podemos empezar a cargar los objetos, se pueden crear también carpetas dentro de otras carpetas, parecido a cómo funciona una estructura de directorio en un sistema operativo.

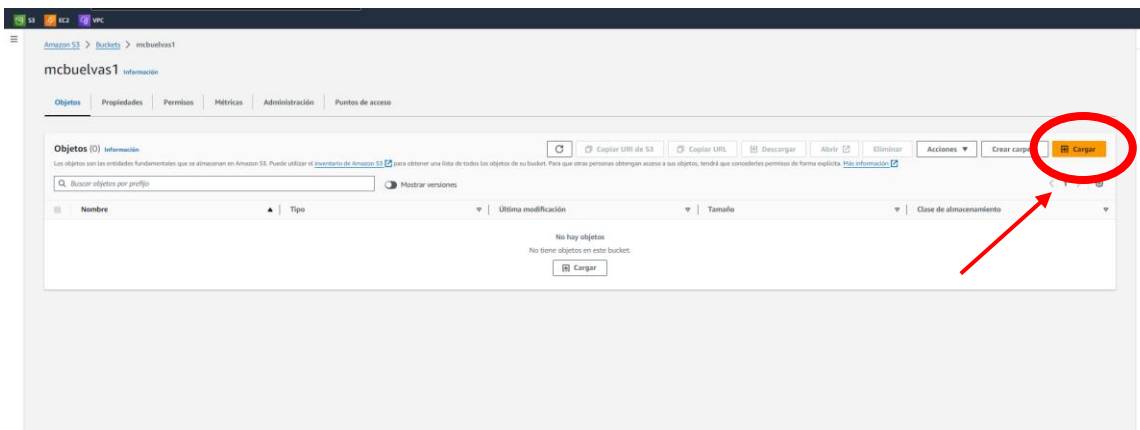


Imagen 73

Al seleccionar la opción de cargar, nos debe desplegar el siguiente cuadro para subir diferentes archivos.

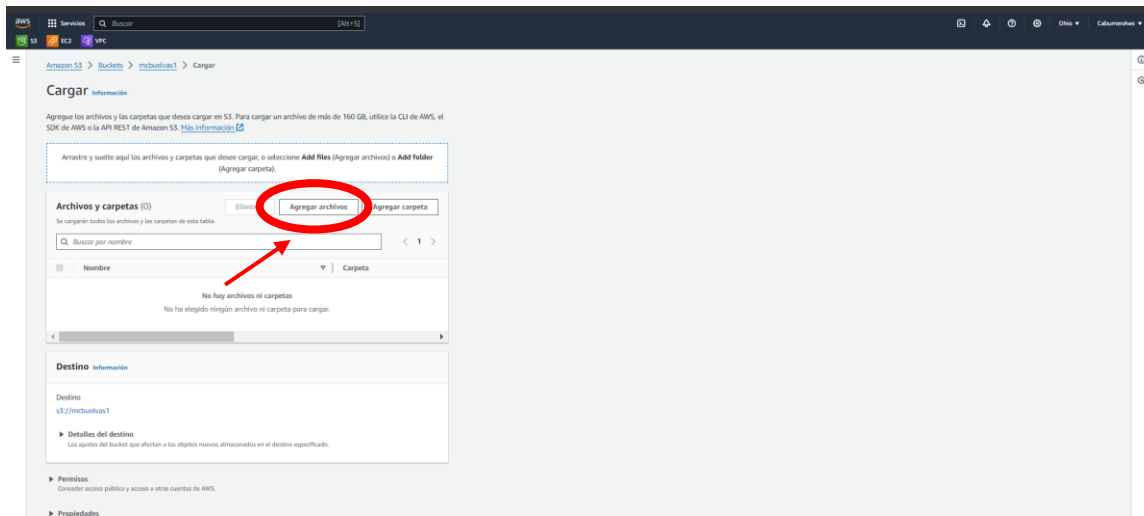


Imagen 74

Cuando selecciono la opción de cargar archivos me debe aparecer este recuadro para buscar y subir el archivo que necesito.

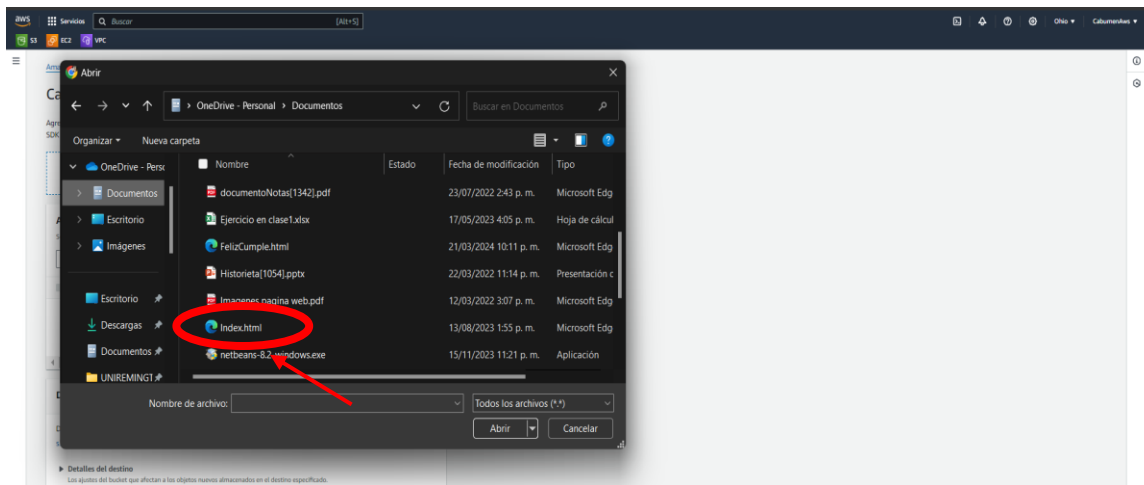


Imagen 75

Con nuestro archivo ya seleccionado entonces le damos subir o cargar, como muestra la imagen76.

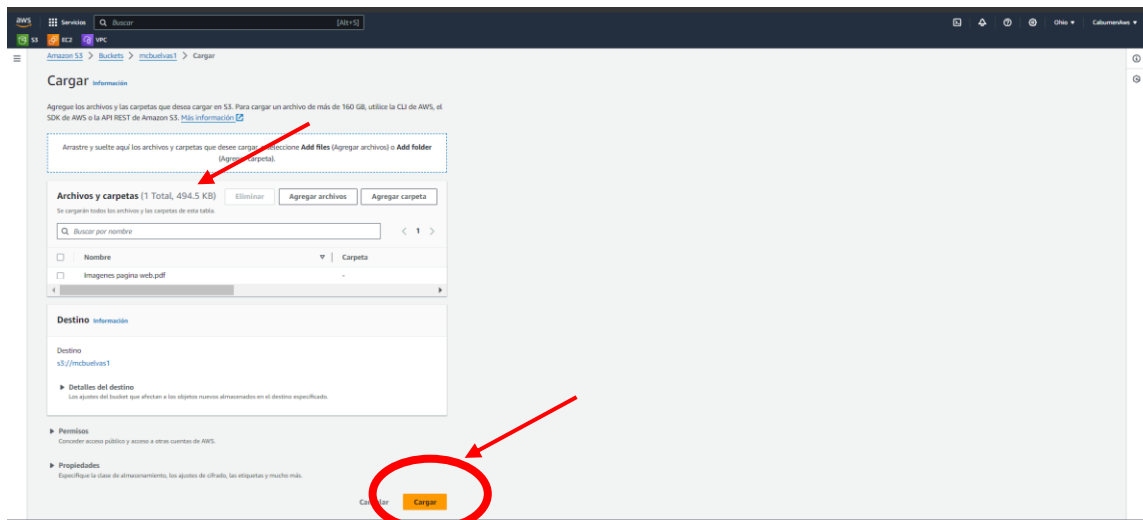


Imagen 76

Cuando nuestro archivo haya cargado, este es el cuadro que nos debe aparecer con la información, se ha realizado la carga correctamente y en resumen debe estar nuestro archivo cargado.

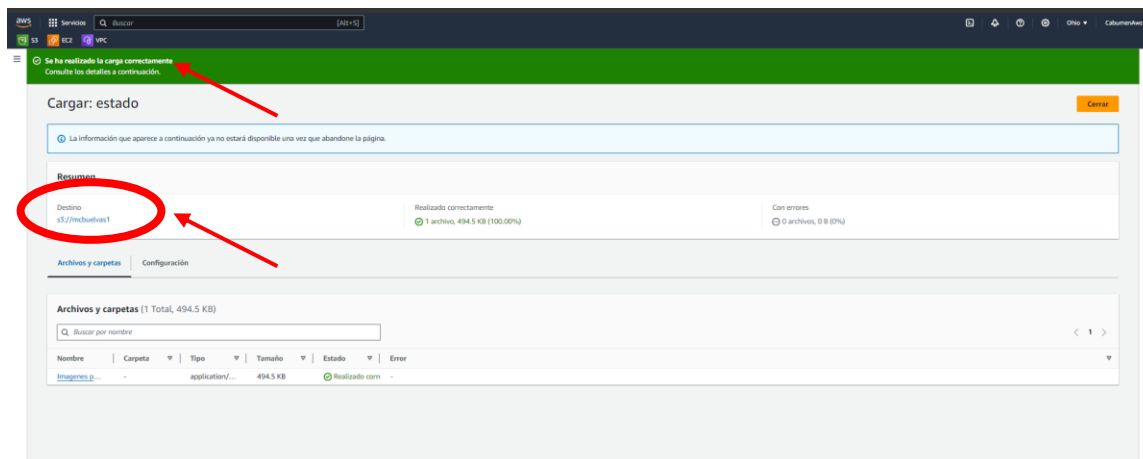


Imagen 77

Cuando ingresamos a nuestro archivo nos debe aparecer esta información, de cuando se creó, que contiene, la URL para acceder a él.

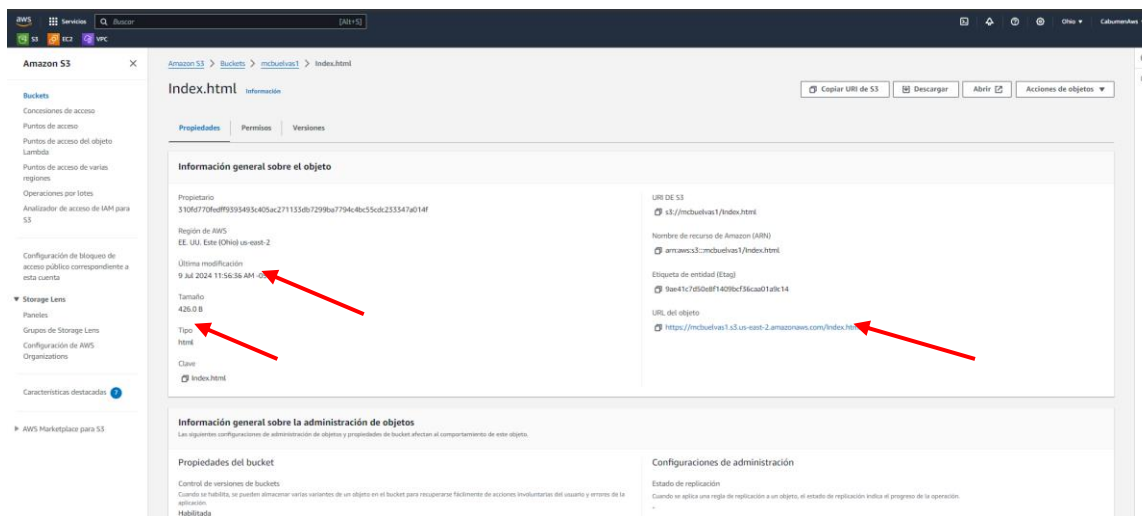


Imagen 78

Si usted va a intentar ingresar a la URL que muestra la imagen77 le va a parecer esta información, aunque la URL se la hayan creado publica, pero la configuración de nuestro objeto es privada y no se va a poder ver desde internet, por eso aparece acceso denegado.

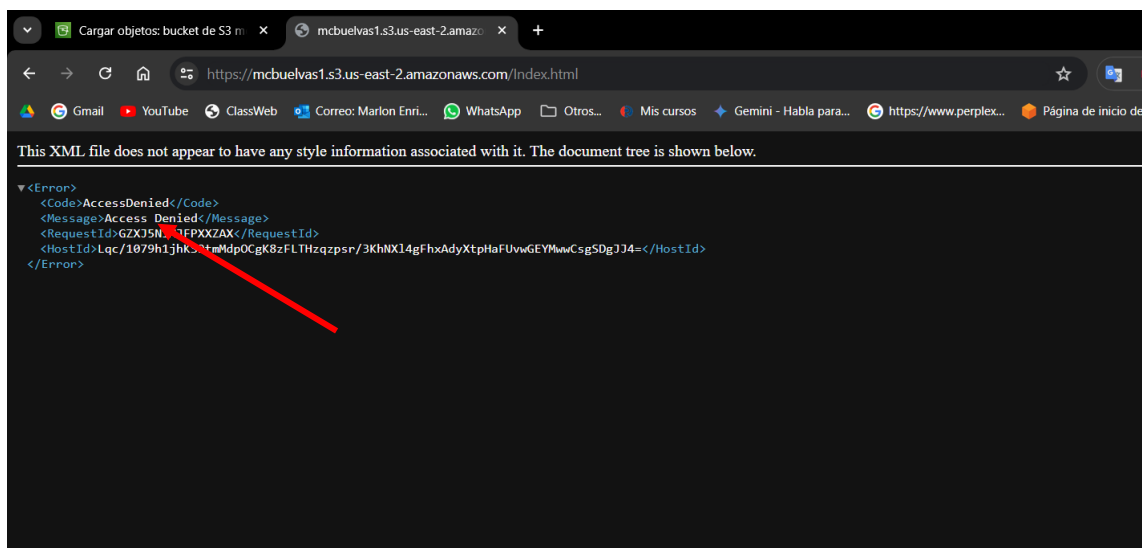


Imagen 79

Aquí podemos ver el historial de las versiones subiendo el mismo archivo que se le hicieron algunas modificaciones.

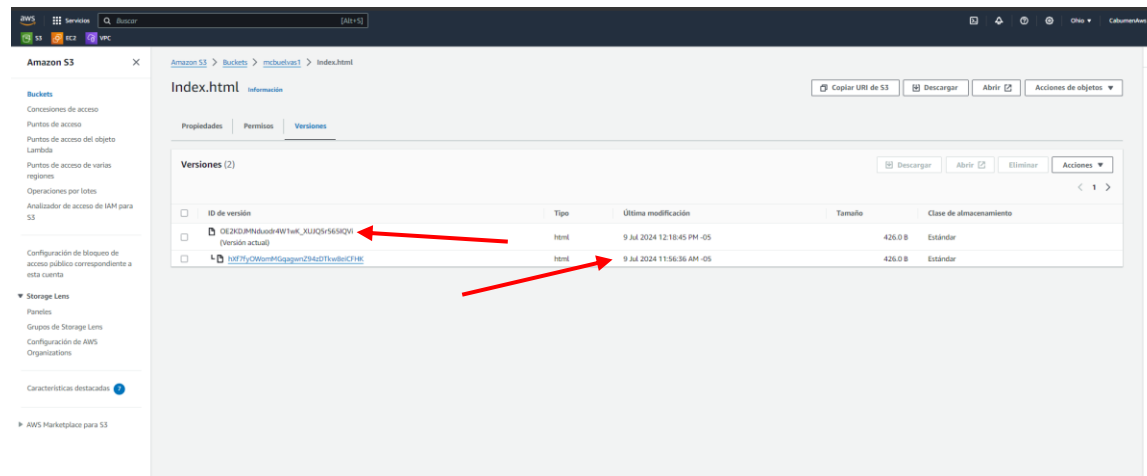


Imagen 80

Sitio web estático, es un sitio que muestra información desde el navegador sin funciones de cálculos desde algún lado de un servidor.

Como funciona esto, vamos a tomar uno de los Bockets que se crearon y nos vamos a las propiedades.

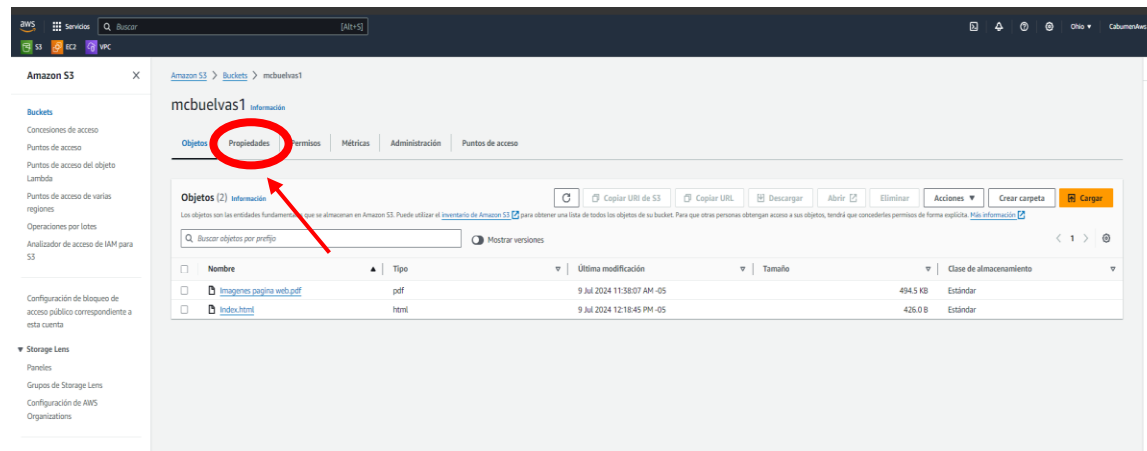


Imagen 81

La opción propiedades nos va a mostrar este cuadro con esta información y vamos a seleccionar editar alojamiento de sitio web estático.

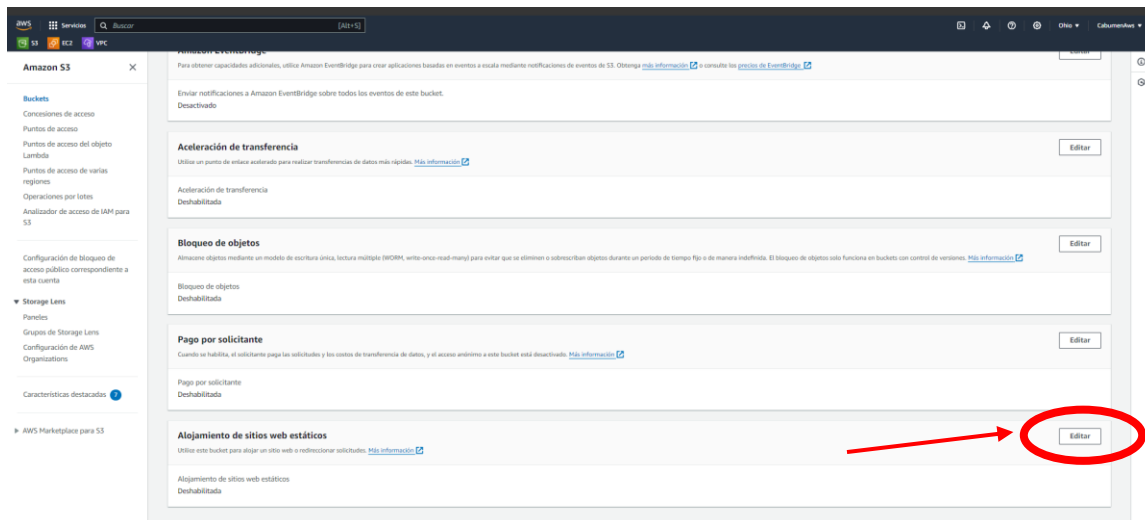


Imagen 82

Nos va a abrir el siguiente cuadro, que por defecto trae el alojamiento deshabilitado y toca habilitar, una vez se habilita muestra toda la información.

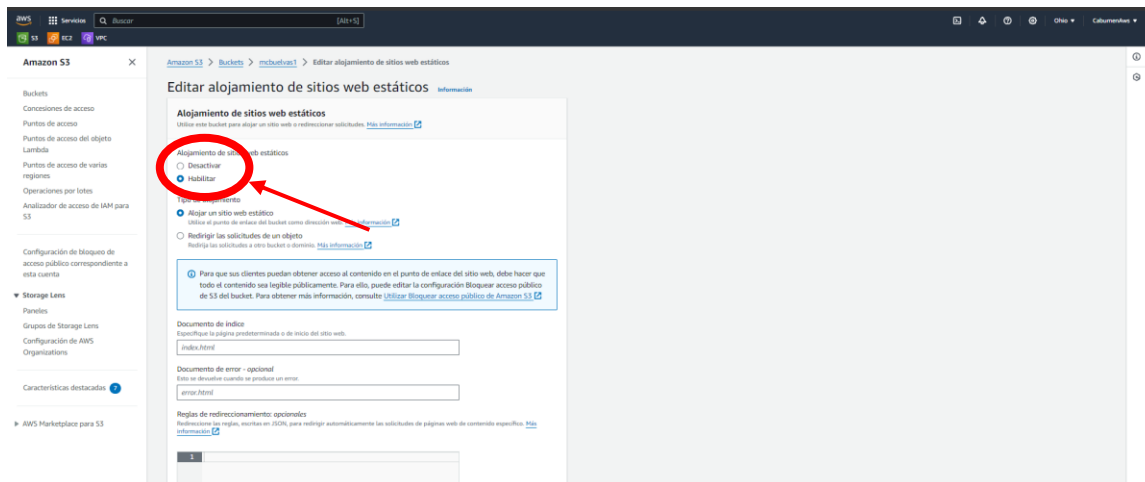


Imagen 83

Cuando ya este habilitado, le decimos que vaya a buscar un archivo que se llama `index.html`.
antes se deben subir los archivos de ese sitio y le damos guardar.

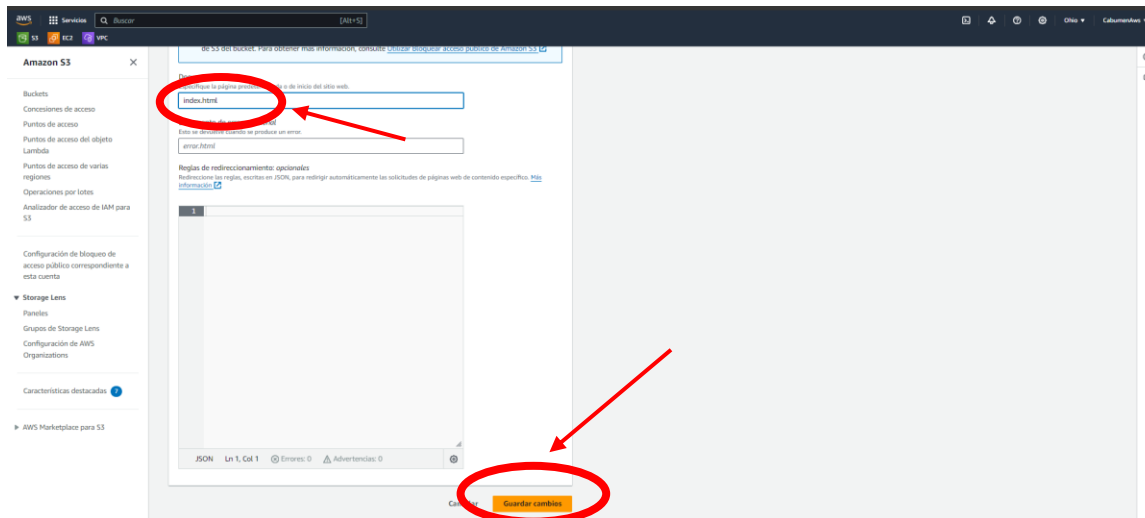


Imagen 84

Con la configuración ya guardada y configurado como un sitio web, vamos a subir los archivos de ese sitio, estando ya cargados debe mostrar toda la información de descarga.

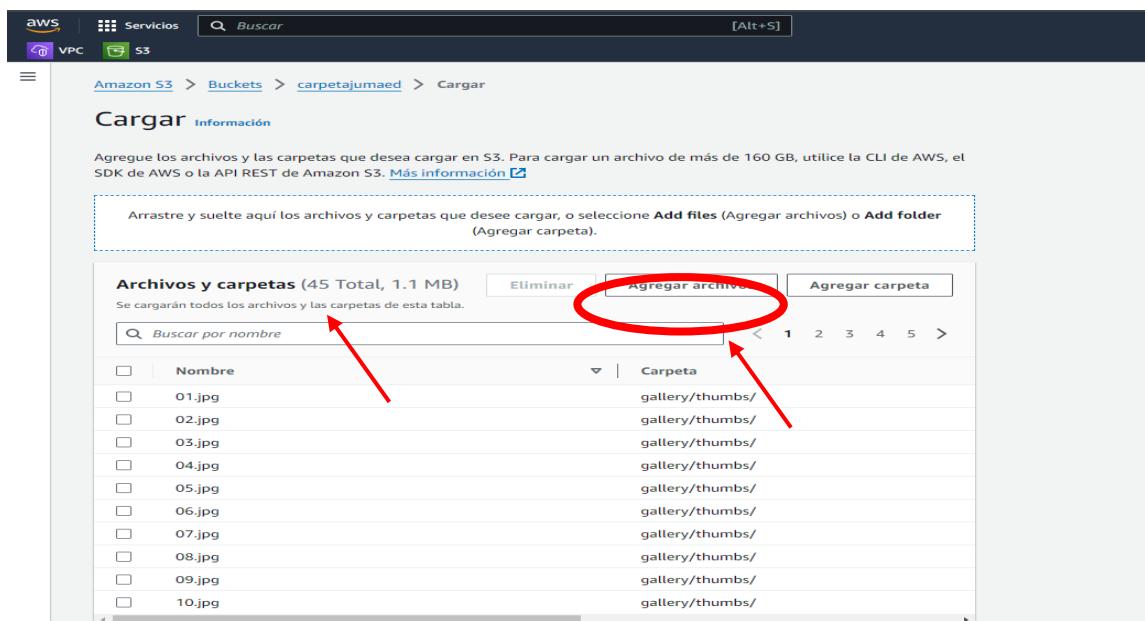


Imagen 85

Así debe quedar cargada la información con los archivos del sitio web.

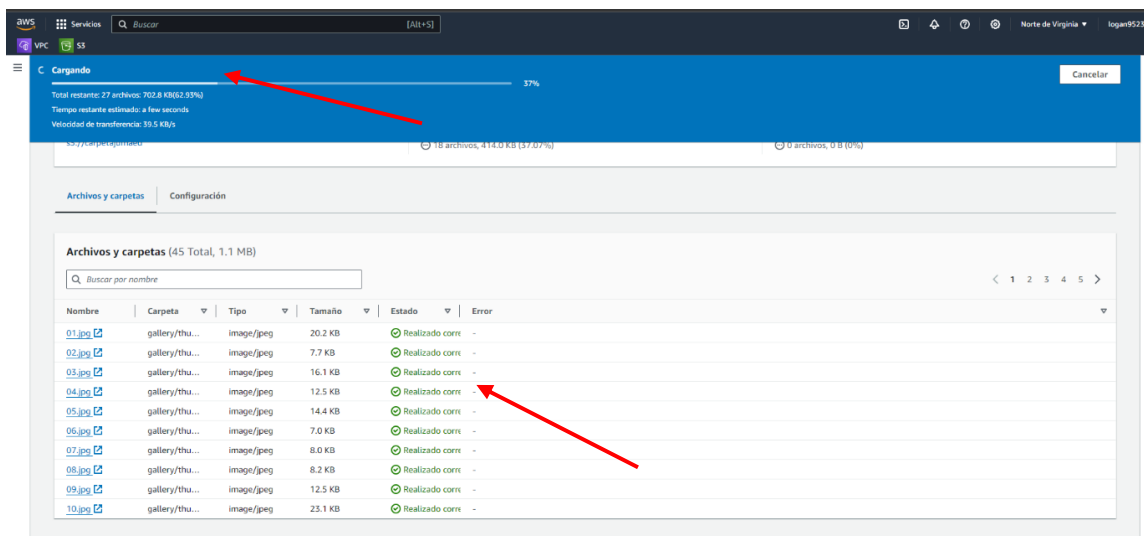


Imagen 86

Teniendo ya todo cargado, debemos habilitar los permisos para poder visualizar el sitio web y guardamos todos los cambios.

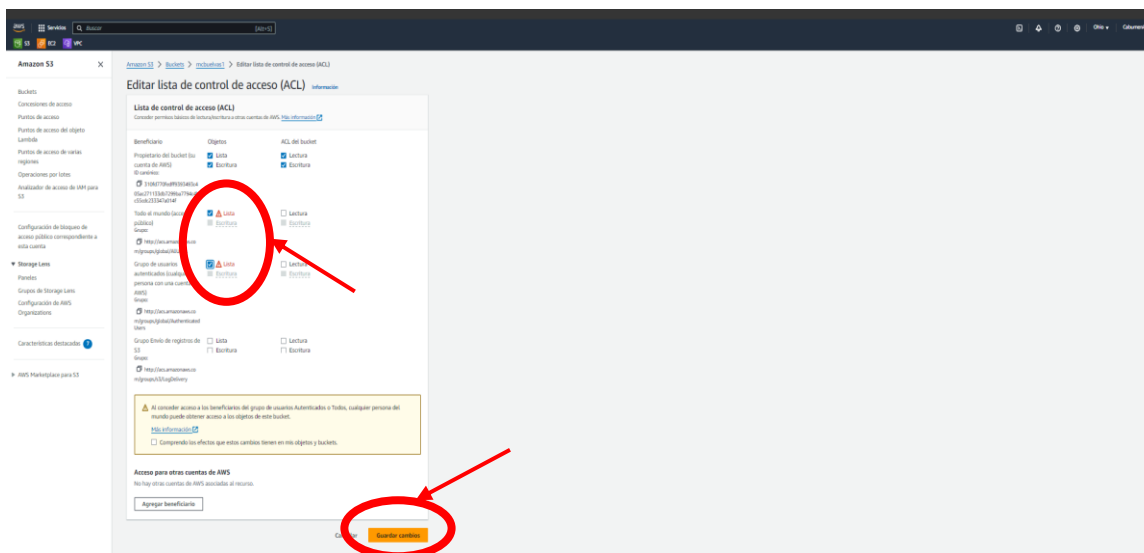


Imagen 87

Nos regresamos a nuestro Buckets y buscamos la URL que nos suministra por defecto AWS y probamos si funciona nuestro sitio web estático.

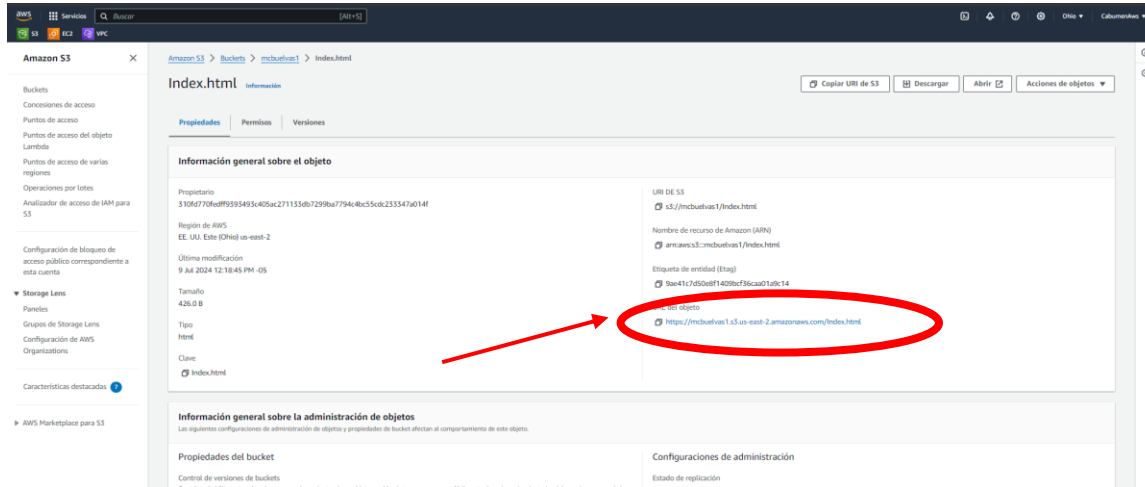


Imagen 88

Esto es lo que debería mostrar, la visualización de algunos archivos HTML toca hacer unos ajustes para que el sitio se muestre como debe ser y es que toca subir correctamente las carpetas que hacen parte del sitio web.



Paradigm Shift

A free responsive site template designed by [@ajlkn / HTML5 UP](#)

- [Next](#)

Magna sed nullam nisi adipiscing

Lorem ipsum dolor sit amet consectetur adipiscing elit. Duis dapibus rutrum facilisis. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Etiam tristique libero eu nibh porttitor amet fermentum. Nullam venenatis eu condimentum. Magna sed etiam consequat, et lorem adipiscing sed nulla. Voluptat nisi et tempus et dolor libero, feugiat magna tempus, sed et lorem adipiscing.

Feugiat consequat tempus ultrices

Etiam tristique libero eu nibh porttitor amet fermentum. Nullam venenatis erat id vehicula ultrices sed ultrices condimentum.

- Consequat tempus
- Etiam adipiscing
- Libero nullam
- Blandit condimentum
- Lorem ipsum dolor
- Nibh amet venenatis

Vehicula ultrices sed ultrices condimentum. Magna sed etiam consequat, et lorem adipiscing sed nulla. Voluptat nisi et tempus et dolor libero, feugiat magna tempus, sed et lorem adipiscing.

Ultrices erat magna sed condimentum

Integer mollis egestas nam maximus erat id euismod egestas. Pellentesque sapien ac quam. Lorem ipsum dolor sit nullam.

Erat aliquam

Vehicula ultrices dolor amet ultrices et condimentum. Magna sed etiam consequat, et lorem adipiscing sed dolor sit amet, consectetur amet do eiusmod tempor incididunt ipsum suspendisse ultrices gravida.

Nisi consequat

Aenean ornare velit lacus, ac varius enim ullamcorper eu. Proin aliquam sed facilisis ante interdum congue. Integer mollis, nisi amet convallis, porttitor magna ullamcorper, amet mauris. Ut magna finibus nisi nec lacinia ipsum maximus.

Imagen 89

Esta es la muestra de que esto si puede servir para hacer alojamientos en un sitio web, así mismo se hizo en el Ec2.

En el siguiente enlace podemos encontrar una breve explicación sobre de cuál de las dos alternativas aquí planteadas elegiríamos para implementar o mantener un sitio web en internet y cuáles son sus ventajas y desventajas de cada una.

<https://www.youtube.com/watch?v=mt9vn81LPGU>



3. Arquitectura de AWS para aplicación con dos instancias con balanceador de carga de contenedores en su interior.

Para este punto vamos a usar las instancias que se crearon en la *imagen12*.

Se crean las instancias.

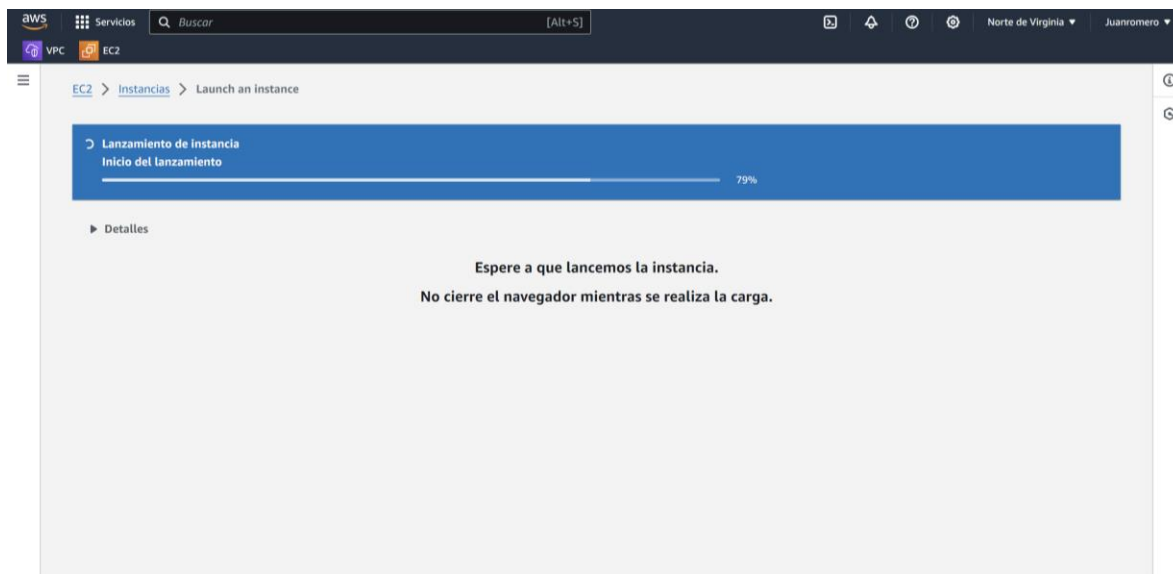


Imagen 90

Así deben quedar nuestras instancias creadas.

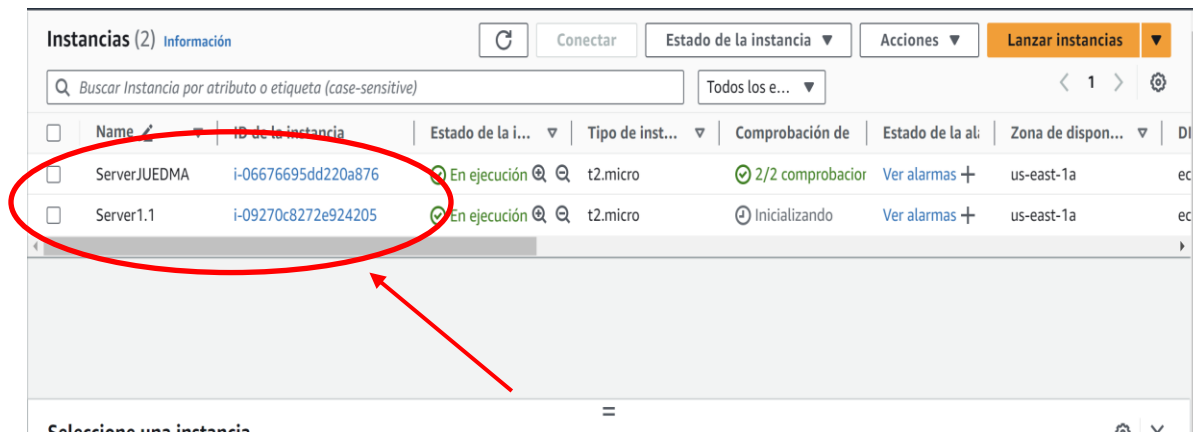


Imagen 91

Hacemos la selección de nuestro balanceador de cargas para nuestra aplicación, así debe mostrar.

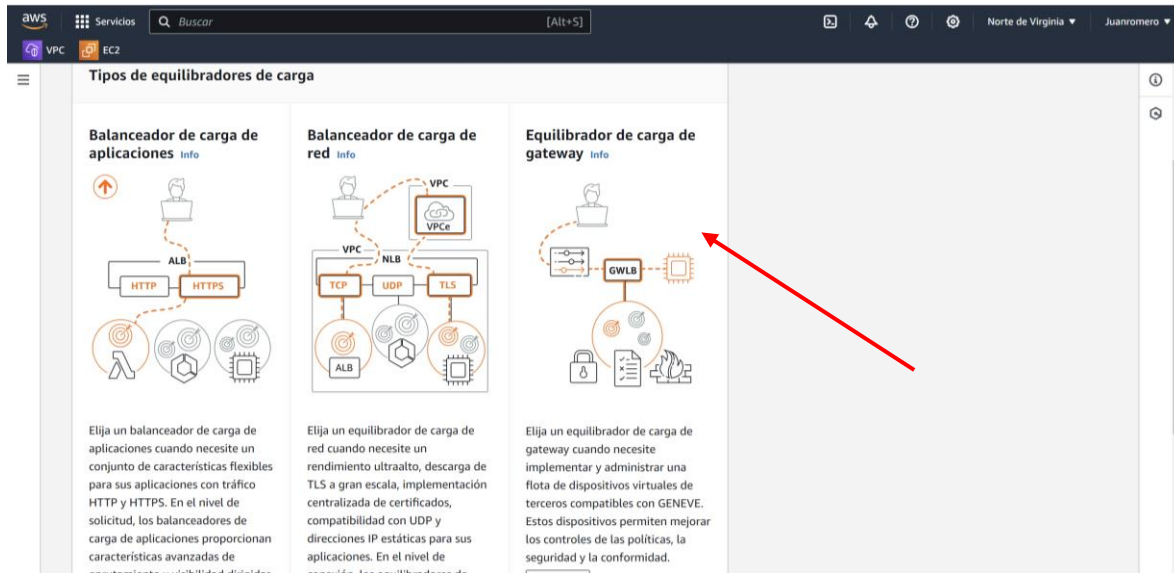


Imagen 92

Para conectar nuestras instancias, se debe descargar el archivo del servidor.

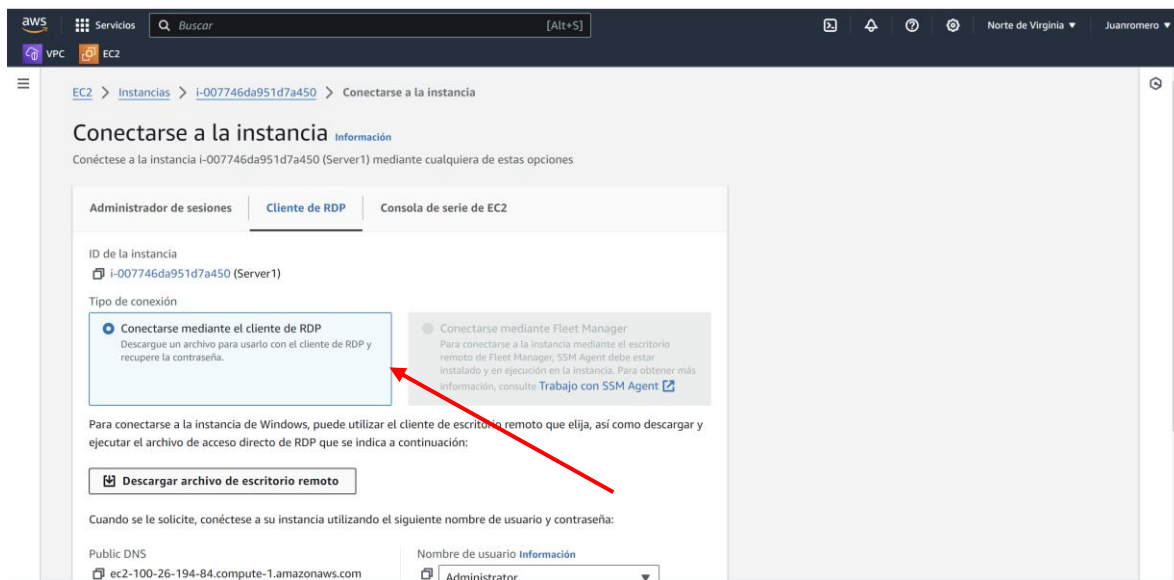


Imagen 93

Desciframos la clave del archivo que se descargó, se debe guardar porque si se pierde no se puede recuperar.

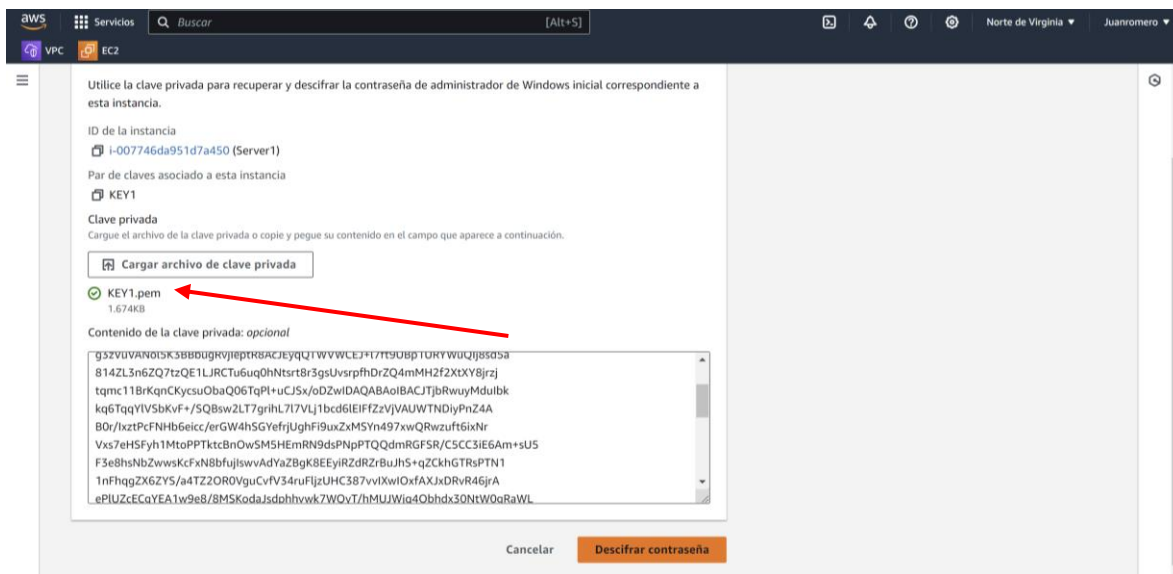


Imagen 94

Iniciamos nuestro servidor, que debe mostrarse de esta manera.

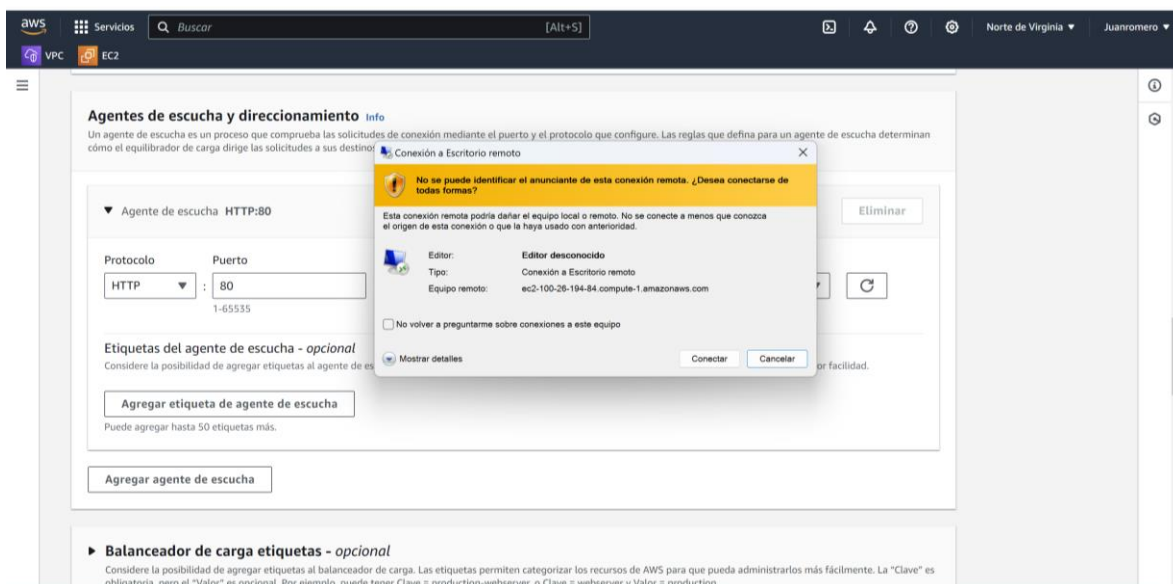


Imagen 95

Cuando hayamos iniciado el servidor, nos debe aparecer así.

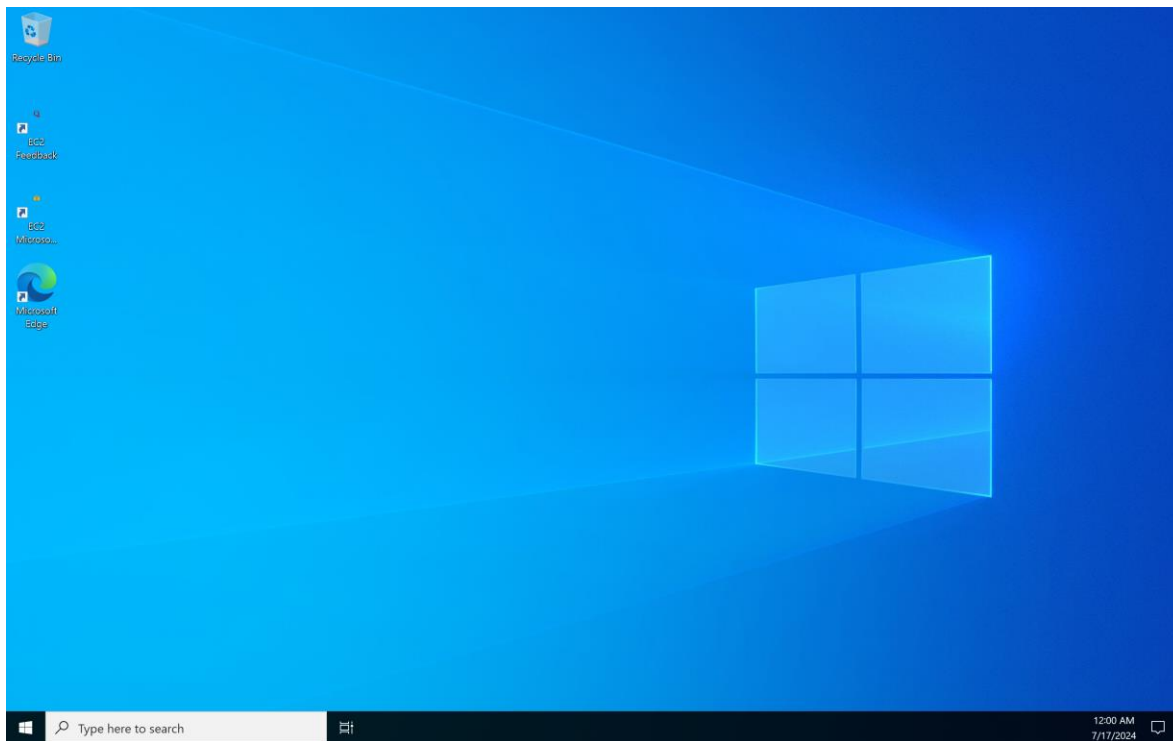


Imagen 96

Se debe cambiar la contraseña a nuestra instancia.

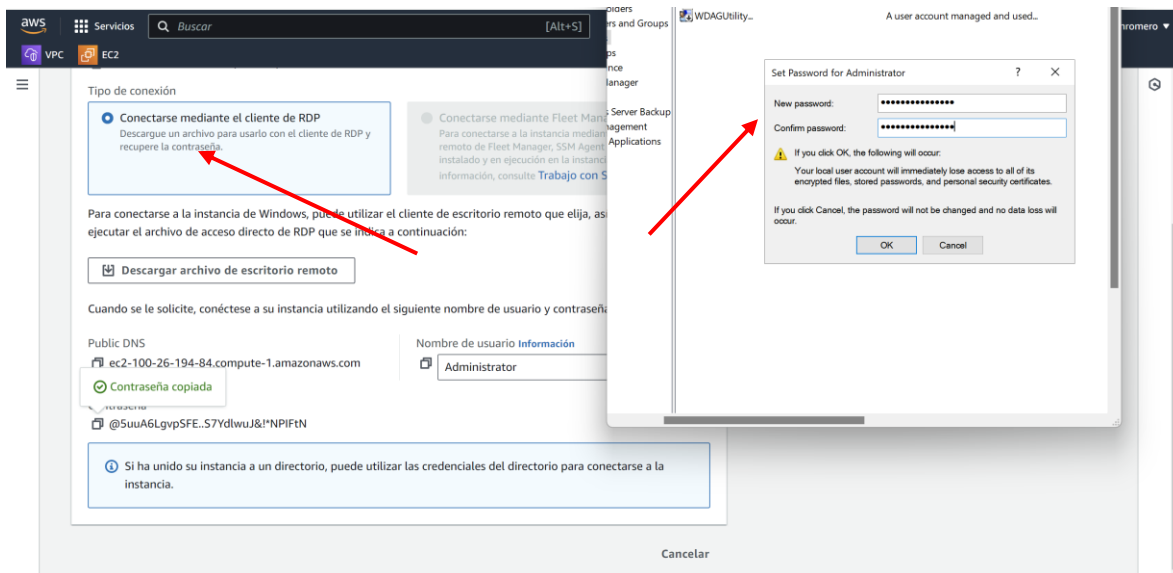


Imagen 97

Instalamos el IIS del servidor de esta manera.

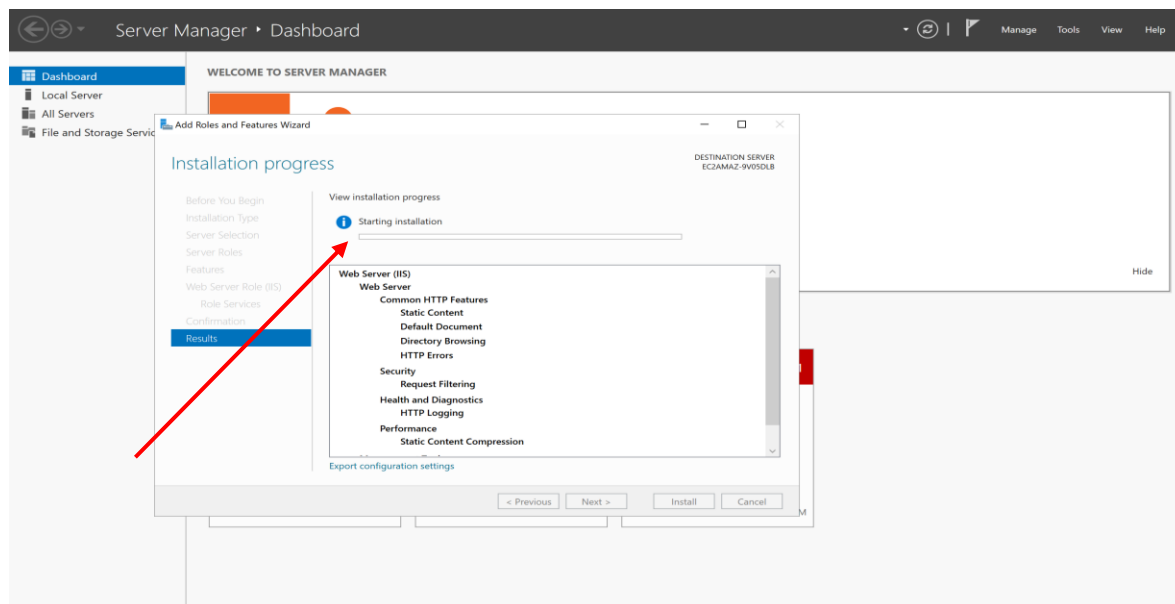


Imagen 98

Si la instalacion es correcta nos debe aparecer de esta manera el servidor.

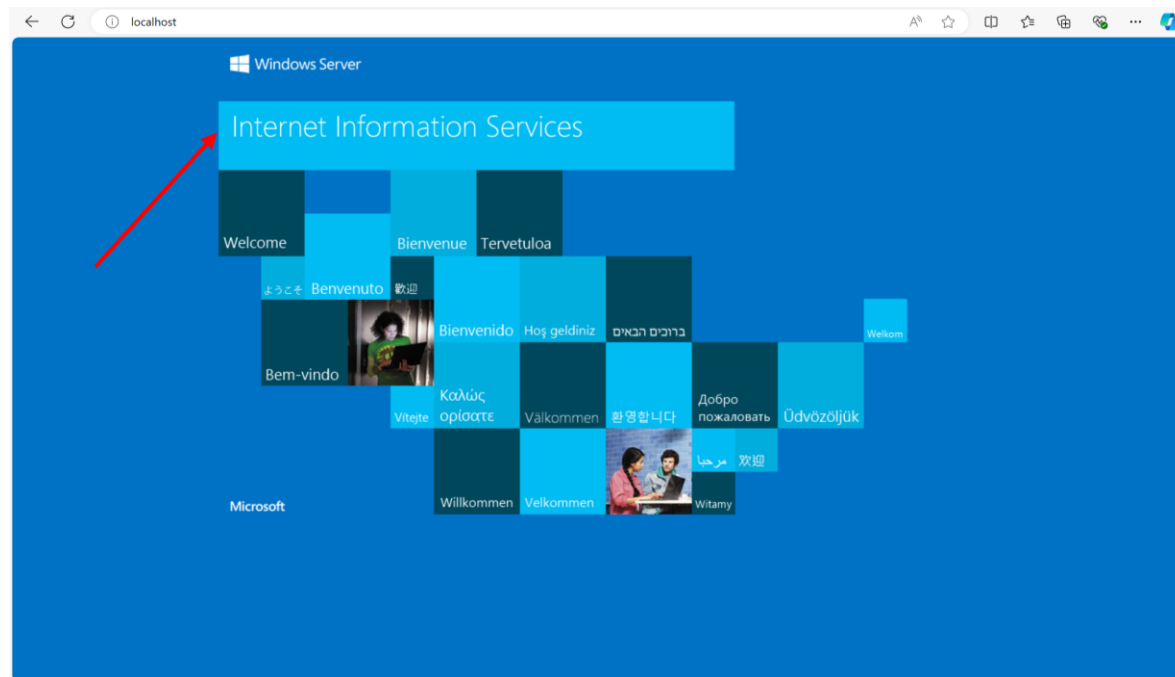


Imagen 99

Cargamos nuestra imagen a la instancia.

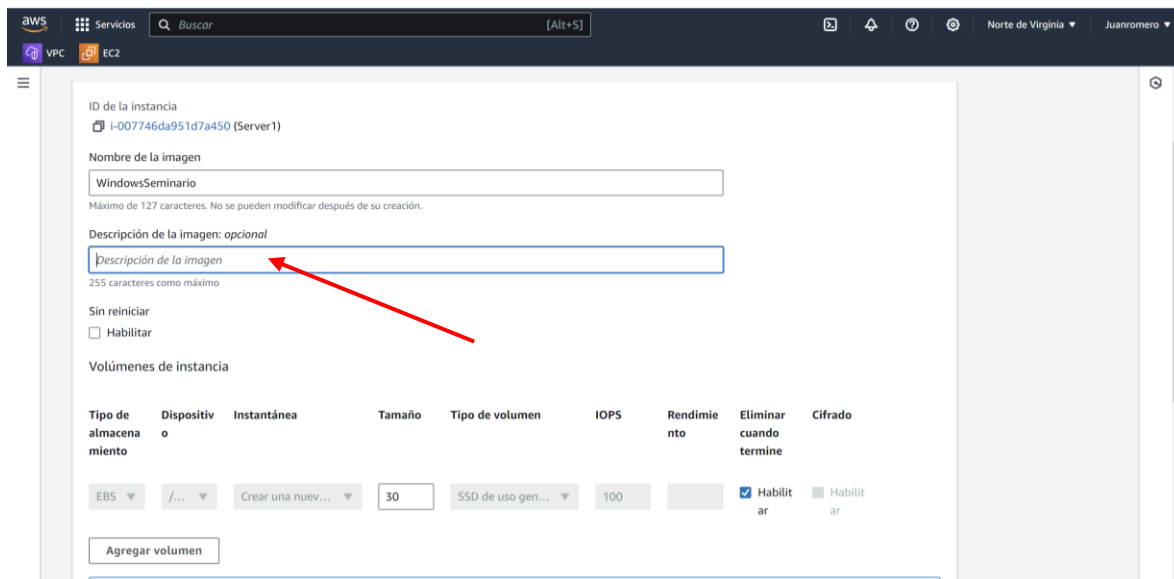


Imagen 100

Se crea la segunda instancia a través de la AMI de esta manera debe mostrar.

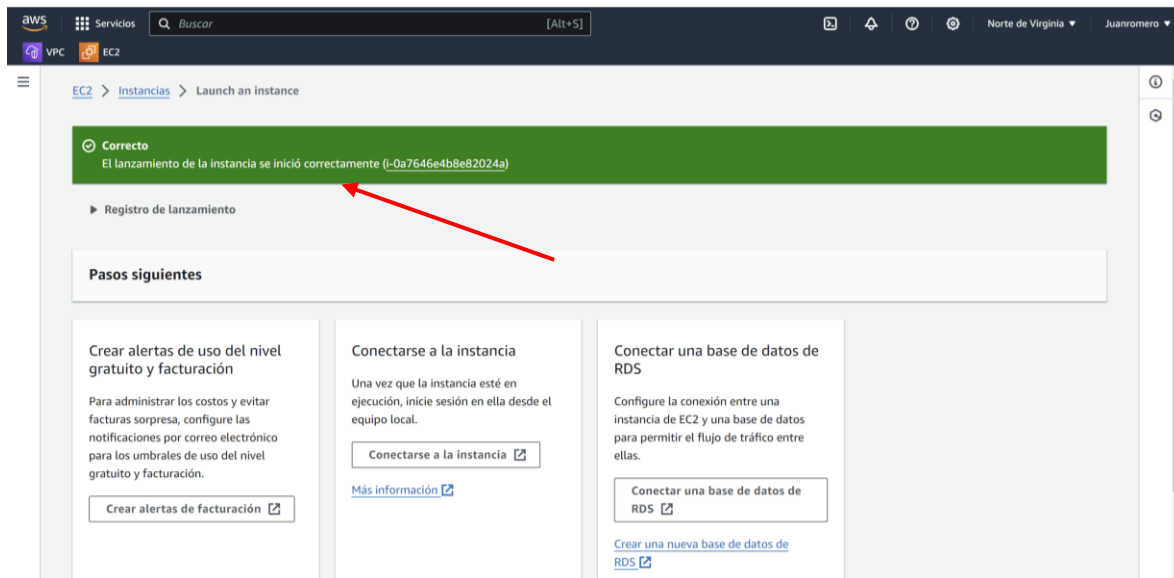


Imagen 101

De esta manera incluimos las dos instancias al TG.

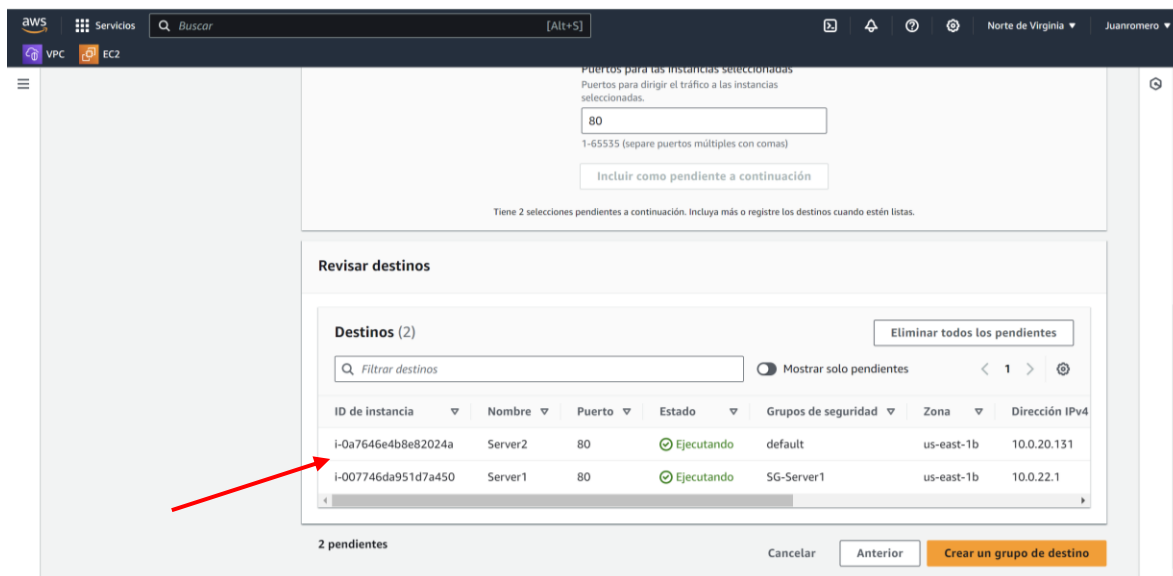


Imagen 102

Se crea nuestro balanceador de carga y debe mostrar este cuadro.

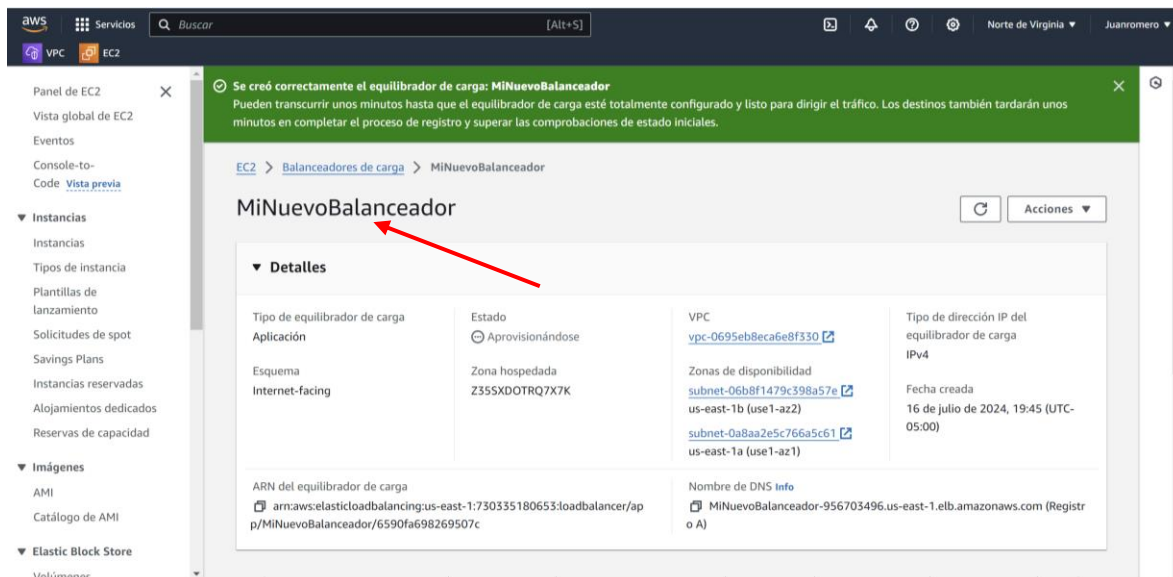


Imagen 103

El servidor 1 ya debe estar funcionando de forma local, así debe mostrar el contenido.

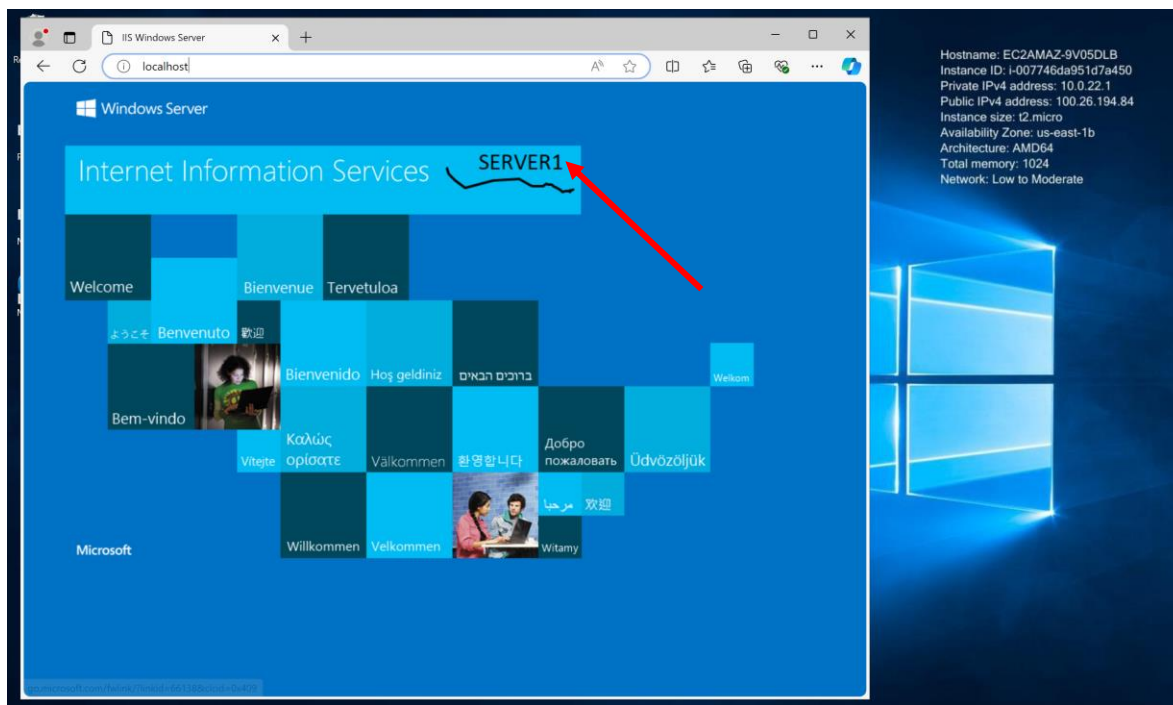


Imagen 104

Creamos nuestra plantilla de lanzamientos de la siguiente manera.

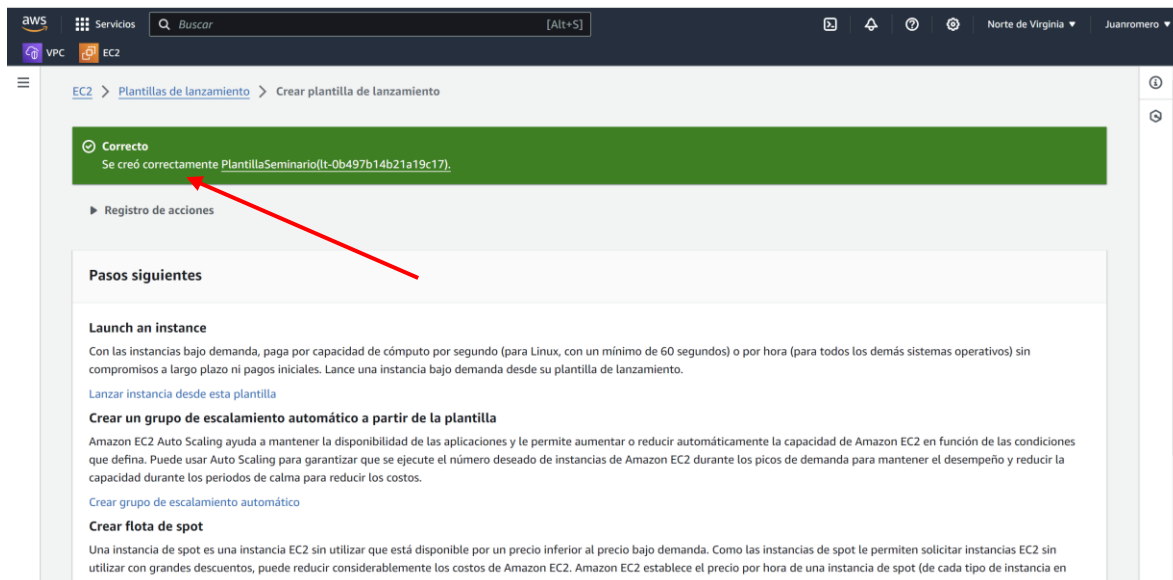


Imagen 105

Una vez ya creadas nuestras instancias, nos debe aparecer el siguiente cuadro con las configuraciones realizadas.

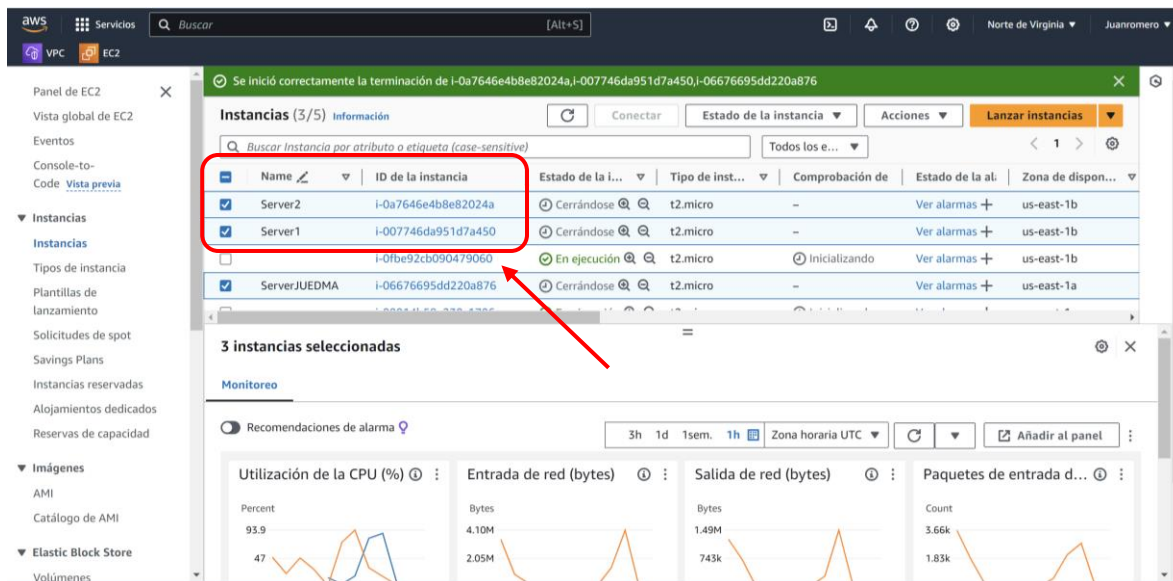


Imagen 106

El auto escalado debe mostrarnos la siguiente información.

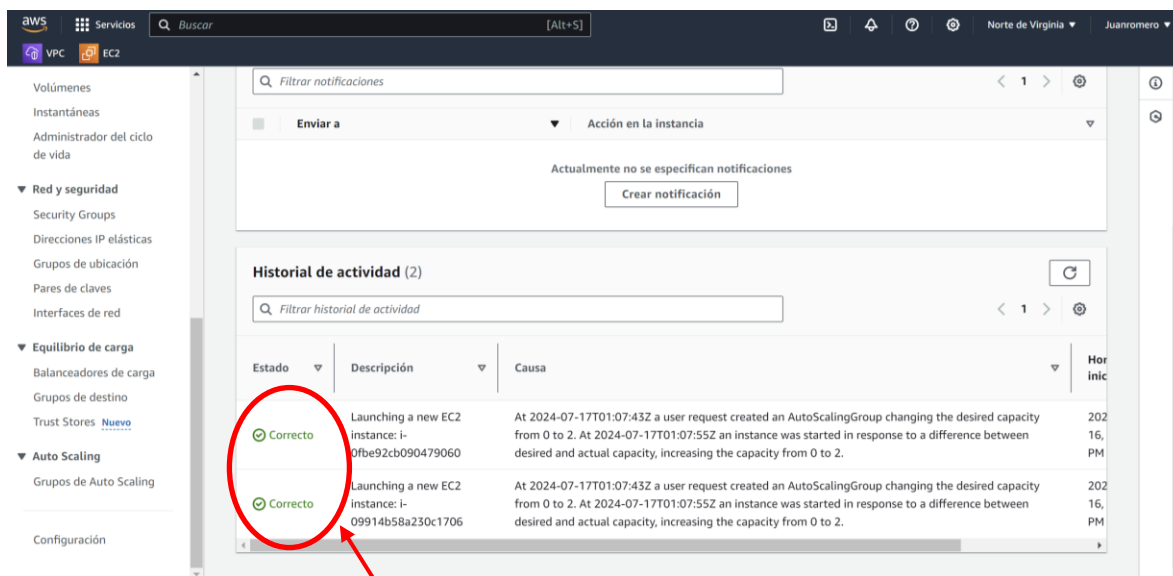


Imagen 107

En el siguiente link van a encontrar un video con una breve explicación de cómo funciona el balanceador de carga en el cual se evidencian los cambios sitios web que generan actividad en el grupo de auto escalado.

<https://youtu.be/J2JSe2PxOjE>



Conclusiones

Explorando la documentación y validando cada una de las clases del profesor, da una gran experiencia a los temas tratados en el curso, que despiertan ganas de seguir investigando de la creación de aplicaciones, sitios web y etc. en la nube para ofrecer una mejor asesoría.

AWS presenta variedad de servicios en la nube que brinda posibilidades a las empresas tanto pequeñas como grandes que necesiten un data center para controlar el funcionamiento interno o tan simple como montar una página web.

Cada empresa es un mundo y sus necesidades a un más grandes por eso es muy importante analizar con calma y detenimiento que tipo de facilidades tienen a la mano y es esto requiere planeación, pero los servicios en la nube son una gran opción no necesariamente con AWS ya que la seguridad y estabilidad que ofrecen tiende a ser más costosa de lo que ofrece otras plataformas de nube con menos estabilidad.

Bibliografía

- [1] «¿Qué Servicios Ofrece Amazon Web Services?». Disponible en: <https://codster.io/que-servicios-ofrece-amazon-web-services/>
- [2] «Beneficios de cloud computing | Seguridad en la nube | AWS», Amazon Web Services, Inc. Disponible en: <https://aws.amazon.com/es/products/>
- [3] M. Giménez, «Amazon Web Services (AWS): ¿qué es y qué ofrece?», Blog de hiberus. Disponible en: <https://www.hiberus.com/crecemos-contigo/amazon-web-services-aws-que-es-y-que-ofrece/>