TRABAJO DE GRADO
Opción Seminario-Diplomado.

Amazon web services (AWS)

Corporación Universitaria Remington.
Facultad de ingeniería.
ingeniería de sistemas.

Camilo Andres Berrocal Ruiz.
Juan Pablo Berrio.
Opción de Trabajo de grado Seminario.
2025

Dedicatoria

Le dedico este trabajo a mis padres que siempre han estado ahí para mí, apoyándome en todo momento y sirviendo de guía, mi madre que nunca dejo de ayudarme sin importar que, y mi padre que siempre fue un apoyo incondicional, siempre estuvieron para mí, gracias por todo lo que han hecho por mí.

Tabla de Contenidos

Resumen

En el siguiente trabajo se va a hablar de algunos servicios que trae Amazon web services (AWS), cómo funcionan esos servicios de AWS, cual es el funcionamiento de esos servicios, se mostrara algunas pruebas que se hicieron con algunos de los distintos servicios, por ejemplo, el servicio de EC2, se usaran recursos como el auto scaling, se podrá evidenciar en ocasiones el uso de esos recursos y como usarlos, se mostrara el uso de la herramienta de balanceador de carga. Se hará un breve recuento de que es la computación en la nube, sus orígenes, se mostrara una línea de tiempo donde se explicara algunos de los momentos clave para la computación en la nube, en que año fueron y cual fue ese hito, también se hablara brevemente sobre algunos otros servicios en la nube, como lo son Azure y GCP, donde se hará una comparación de esos servicios de computación en la nube.

Palabras clave

- Computación en la nube.
- Instancias.
- Vpc (virtual private cloud).
- Balanceador de carga.
- Auto scaling.
- Docker.
- Comandos de Linux.

Marco conceptual y contextual

Marco conceptual

Computación en la nube: la computación en la nube es el uso de recursos como almacenamiento, procesamiento o bases de datos a través del internet. (Zettle, s.f.)

AWS (Amazon web services): es un servicio de computación en la nube que permite el uso de distintas herramientas, para el desarrollo, prueba e implementación de aplicaciones, con una gran seguridad, brinda también servicios de virtualización. Redes virtuales, entre otros servicios. (Marujita, muy tecnológicos, 2023)

EC2: ec2 es un servicio dentro de AWS que ofrece la posibilidad de crear instancias, crear AMI, usar balanceador de carga. (tarlogic, s.f.)

Instancia: la instancia es una máquina virtual creada dentro del servicio de ec2 de AWS con los recursos que requiera la maquina como por ejemplo un sistema operativo ya sea Linux o Windows, se le puede colocar un almacenamiento interno, memoria RAM y otros recursos que podría tener una maquina física (Documentación aws, s.f.).

Vpc (virtual private cloud): es una red virtual, que sería similar a una red física, con los beneficios que trae AWS y que permite hacer pruebas de manera pública a través de una IP pública o hacer pruebas locales con una red privada. (documentación AWS, s.f.)

Ip publica: una IP pública es identificador numérico único que identifica a cada dispositivo en el internet, esa dirección se puede usar para conectar con diferentes maquinas remotamente. (Marujita, muy tecnológicos, 2023)

Balanceador de carga: un balanceador de carga es un recurso que permite distribuir el trafico entre distintas instancias o aplicaciones asociadas al balanceador, para no saturar una sola instancia, configurando previamente el balanceador y las instancias manualmente. (AWS Amazon, s.f.)

Auto scaling group: es un servicio que permite la creación de instancias automáticamente de acuerdo con el consumo que este teniendo una instancia, ya sea en almacenamiento o CPU, configurando que recurso va a estar vigilando; el auto scaling permite gestionar la cantidad de instancias que se podrán crear de manera automática basado en una AMI creada previamente. (Documentación aws, s.f.)

AMI (imagen de máquina de Amazon): es una imagen de una instancia creada a partir de una instancia que ya estaba previamente creada y configurada anteriormente. (Documentación aws, s.f.)

Target group: los targe group o grupo de destino son los encargados de redirigir las solicitudes a las instancias mediante protocolos o puertos que se especifiquen. (Documentación aws, s.f.)

Puertos virtuales: los puertos virtuales en internet hacen referencia a distintos números puertos utilizados para hacer conexiones en la red o con servicios como lo es el puerto para la web http que es el puerto 80. (akamai, s.f.)

Docker: es una aplicación que permite empaquetar aplicaciones o software en un contenedor con todas las herramientas necesarias para que la aplicación pueda funcionar correctamente. (10code, 2024)

Nginx: es un servidor web que sirve para hacer proxy reverse y funciona de balanceador de carga. (B G. , 2025)

Proxy reverse: el proxy reverso o proxy inverso tiene la función de recibir las solicitudes de los clientes procesar esas solicitudes y luego enviarlas a los servidores adecuados. (B D. , 2025)

Marco contextual

El objetivo del trabajo es mostrar los temas que se abarcaron en el seminario de Amazon web services (AWS), mostrando varios servicios que fueron utilizados para hacer diferentes pruebas durante los distintos encuentros virtuales y mostrando evidencia visual, como, imágenes donde se pueda ver los resultados que se van dando con el uso de esos servicios, dependiendo los requerimientos solicitados para realizar las distintas pruebas, también se hablará de temas conceptuales explicando que es la computación en la nube, hablar sobre cuáles fueron los orígenes de la computación en la nube, se mostrara una línea de tiempo donde se podrá ver cuales fueron esos hitos que marcaron en distintas fechas la computación en la nube, y que hizo que la computación en la nube tenga tanta relevancia, se mostrara una breve comparación con otros servicios de computación en la nube que existen como lo son Azure y GCP.

Para este trabajo, se mostraran diversas imágenes mostrando el uso de instancias, que fueron creadas para hacer algunas pruebas, evidenciando además el uso del recursos que tiene AWS que es el balanceador de carga, se podrá observar la prueba de estrés que se le hizo a una instancia Linux para activar el balanceador de carga y el auto scaling que se configuro para que fuera creando de manera automática varias instancias en base a una AMI que se creó previamente de ese servidor Linux, si el consumo de CPU supera cierto umbral el auto sacling creara de manera automática otras instancias, hasta un máximo de 4 instancias.

Se mostrará también una instancia con sistema Linux en el cual se hicieron configuraciones para que tenga funcionando unos contenedores o Docker, que contengas unos archivos index que se mostraran en el navegador, se podrá ver los distintos contenedores y los puertos que fueron seleccionados para realizar las pruebas, se mostrara una configuración de proxy reverse que se hizo en el sistema Linux para redireccionar por los puertos a los

que se quiero visitar los sitios, todo esto mediante un repositorio de nginx, se configuraron los puertos en el target group o grupo objetivo para que se pudieran acceder a ellos en AWS.

Desarrollo e implementación del aprendizaje

¿Qué es la virtualización?

La virtualización es una tecnología que se puede usar para crear representaciones virtuales de servidores, almacenamiento, redes y otras máquinas físicas. El software virtual imita las funciones del hardware físico para ejecutar varias máquinas virtuales a la vez en una única máquina física. También potencia los servicios de computación en la nube que ayudan a las organizaciones a administrar la infraestructura de manera más eficaz. (AWS Amazon, s.f.)

Inicios de la virtualización

En la década de 1960, múltiples ingenieros informáticos se dieron la tarea de encontrar una manera que los ordenadores hicieran múltiples tareas simultáneamente, ya que en ese tiempo los ordenadores solo podían ejecutar un programa a la vez, sin embargo, con el tiempo los ingenieros encontraron la manera de crear un sistema que realizara varios programas a la vez; lo que se conoció como "tiempo compartido".

En la década de 1970 la virtualización comenzó a aplicarse en mainframes, (computadoras de alto rendimiento), ya que los ingenieros desarrollaron sistemas que permitían que múltiples máquinas virtuales ejecutarse en un solo mainframe. En 1990 la virtualización empezó aplicarse a ordenadores personales, que de ahí mismo fueron surgiendo empresas tales como VMware y Virtual PC, que permitía a sus usuarios ejecutar múltiples sistemas operativos en un solo ordenador.

Desde entonces la virtualización ha ido evolucionando y se ha convertido en una herramienta importante en la industria de la informática. Hoy en día la virtualización se ha ido utilizando en una variedad de aplicaciones, desde la virtualización de servidores hasta la virtualización de escritorios. (redhat, 2024)

¿Qué es la computación en la nube?

Es un modelo que entrega servicios a través de TI, (tecnología de información), a través de internet, es decir los usuarios pueden acceder a cualquiera de estos recursos de manera remota desde cualquier dispositivo con conexión a internet, gracias a esta herramienta, los usuarios pueden disfrutar a cualquier hora y en cualquier lugar sus servicios, ya sea en Smartphone y Tablet, también es posible, almacenar archivos, trabajar y crear copias de seguridad sin ocupar espacio en el dispositivo. (wikipedia, s.f.)

Inicio de la computación en la nube

El inicio de la computación en la nube nos lleva al señor HERB GROSH, quién fue uno de los primeros en usar ese concepto y quien además dijo afirmar en la década de 1950 que una mejor manera para que las economías se adaptaran mejor si confiaban en el almacenamiento centralizado; que es donde entra el señor JHON McCarthy, quien introdujo el concepto de inteligencia artificial y también mencionó sobre el tiempo compartido, que entre la década de 1960 y 1970 fue implementado en muchas empresas de EE.UU, ya que este ofrecía transferencias de horas, muy parecido al que usamos en la actualidad AMAZON AWS o GLOOGLE CLOUD. (infranet working, 2017)

Línea de tiempo
Inicio de la virtualización:
1960: creación de un sistema que fuera capaz de realizar múltiples tareas simultáneamente.

1970: La virtualización empezó aplicarse en computadoras de alto rendimiento (mainframes).

1990: La virtualización empezó aplicarse en ordenadores personales y surgen nuevas empresas como: WMware y Virtual PC.

1995: Se lanza el primer servicio de almacenamiento en la nube llamado NetStorage**.**

1999: Se lanza (enlace unavailable), código de estado HTTP lo que significa que el servidor del sitio web simplemente no está disponible en este momento, una de las primeras aplicaciones de software como servicio (SaaS).

2002: Se lanza Amazon Web Services (AWS), una de las primeras plataformas de infraestructura como servicio (IaaS).

2006: Se lanza Google Apps, una suite de aplicaciones de productividad en la nube.

2008: Se lanza Microsoft Azure, una plataforma de computación en la nube.

2010: La madurez de la computación en la nube, así como también se lanza iCloud, un servicio de almacenamiento en la nube de Apple.

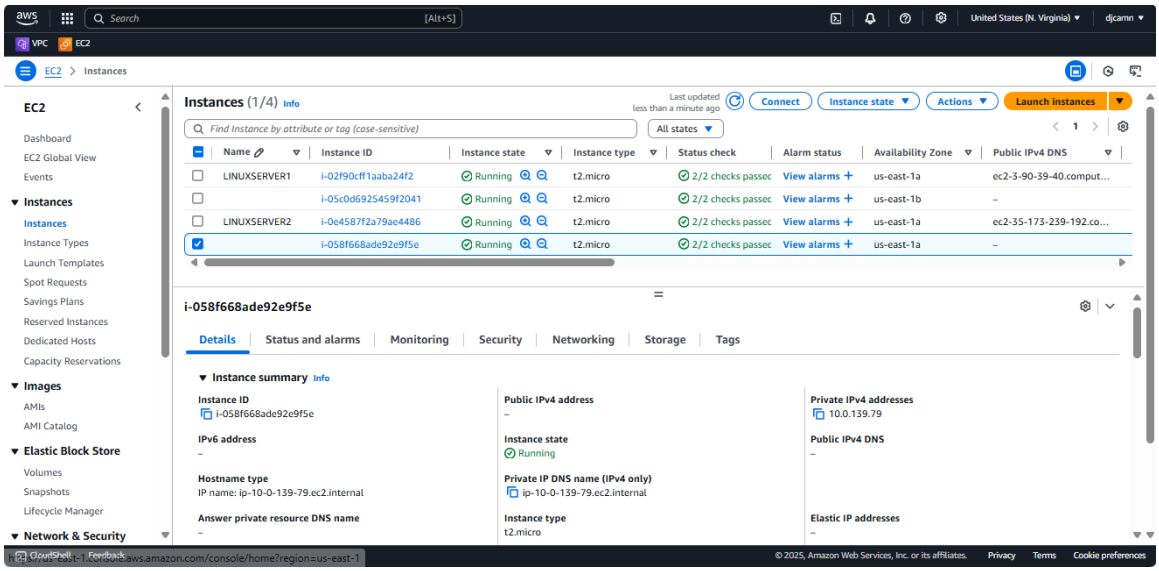
2015: Se lanza Amazon Lambda, un servicio de computación sin servidor.

2020: En este año se logran lanzar 2 cosas innovadoras de la computación en la nube híbrida y multicloud, se lanza Azure Arc, un servicio de computación híbrida de Microsoft y google Anthos, un servicio de computación híbrida de Google. (time toast, s.f.)

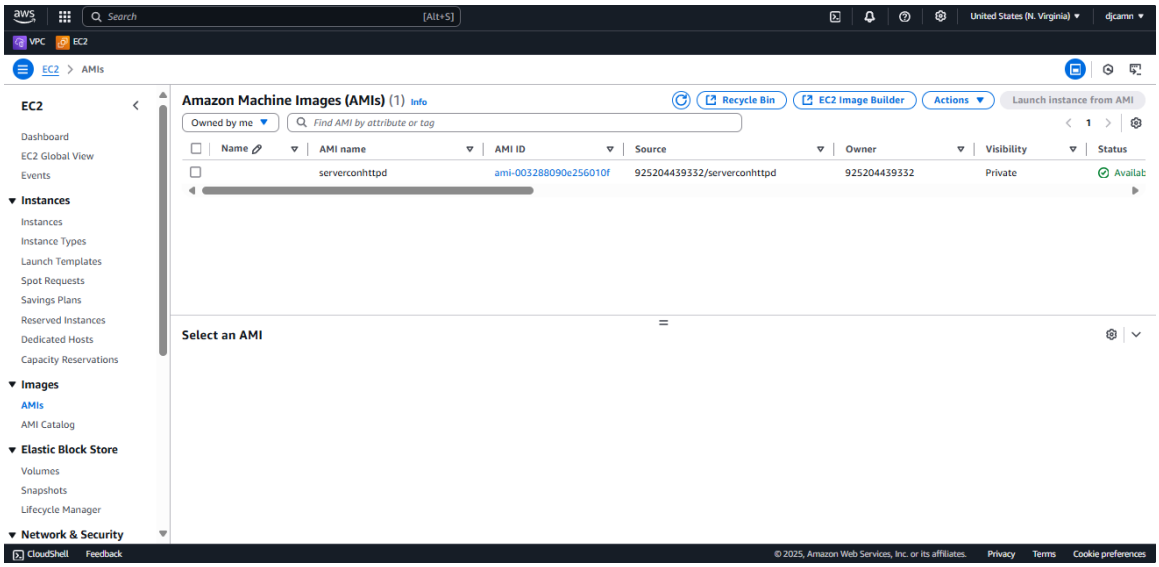
DIFERENCIAS ENTRE AWS, AZURE Y GCP
AWS, AZURE y GCP, son los tres principales proveedores de servicios en la nube, sin embargo, cada uno cuenta con diferentes funcionamientos, aunque también tienen un poco de similitud como, por ejemplo: empecemos con AWS, es el gigante de la nube, debido a que cuenta con mayor variedad de servicios, tales como, computación, almacenamiento y machine learning. (subcampo de inteligencia artificial que usa algoritmos para aprender de los datos), en otras palabras, es un supermercado de servicios en la nube, algo muy servicial y práctico para sus usuarios, por otro lado, AZURE, está integrado con servicios de Microsoft como Windows y Office. Es como un proveedor de la nube especializado en Microsoft, lo cual es ideas para empresas que ya utilizaban estos productos.

GCP, es un proveedor de servicios en la nube que está enfocado en la inteligencia artificial y machine learning, lo que lo hace ideal para proyectos que requieres tecnología. Algunas similitudes que podemos encontrar, es que los tres proveedores cuentan con flexibilidad, almacenamiento en la nube, seguridad etc. Esto permite que sus usuarios puedan crear y configurar sus propias máquinas virtuales y redes, sin tener que preocuparse por la infraestructura y así mismo también se permite recuperar sus datos de manera segura. Todo esto con enfocarse en la seguridad y cumplimiento. (paradigma digital, 2019)

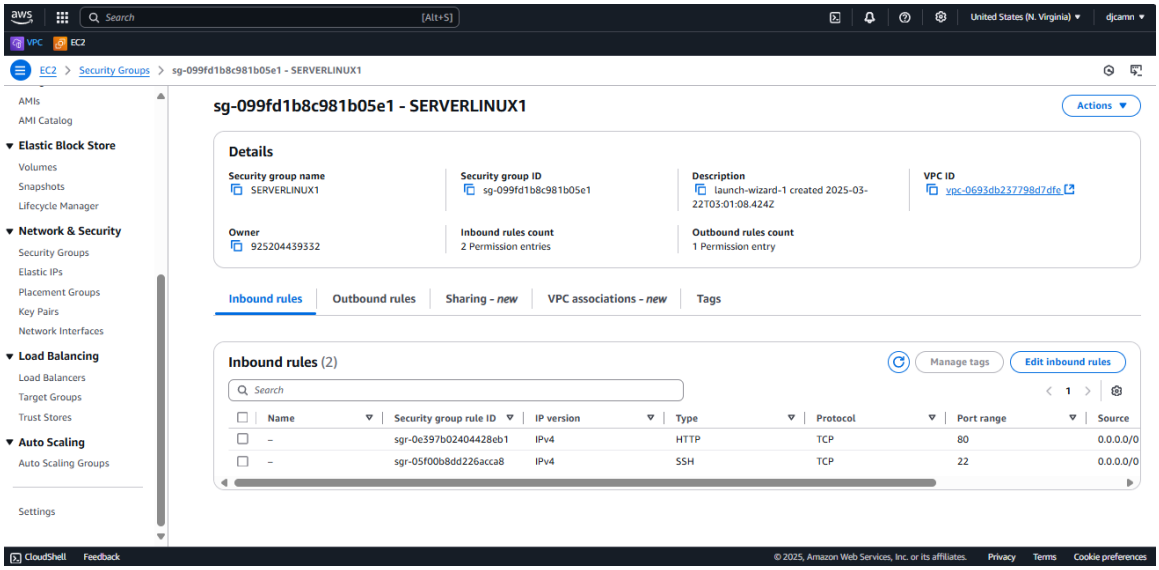
Evidencia de realización de prácticas del seminario.



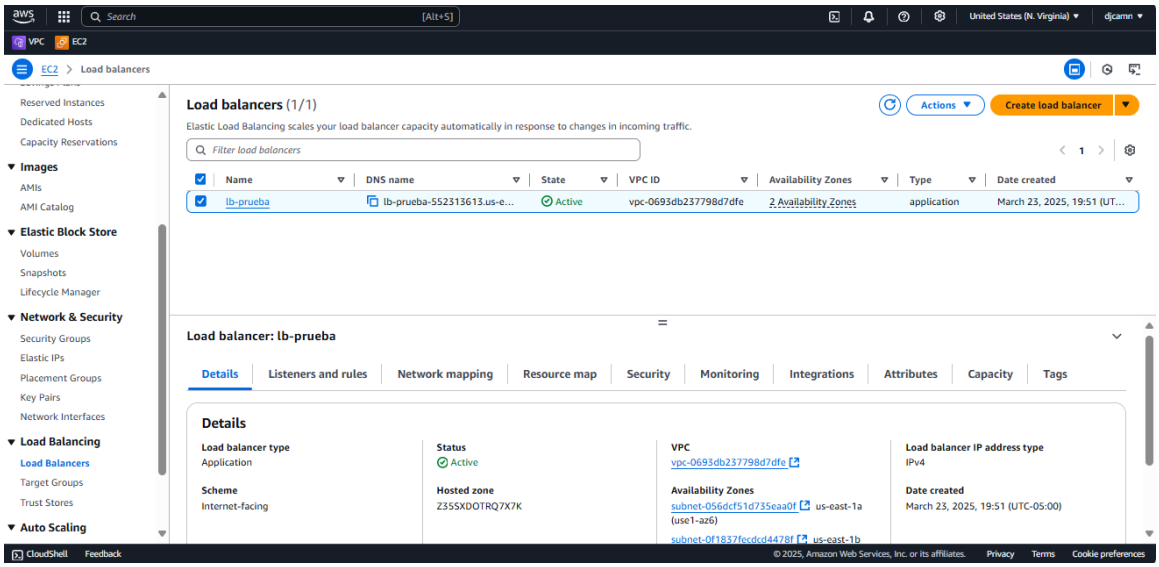
Evidencia de las distintas instancias creadas, dos creadas manualmente con sistema Linux, las otras creadas atreves de la AMI de la instancia LINUXSERVER1 con ayuda del auto escalado.



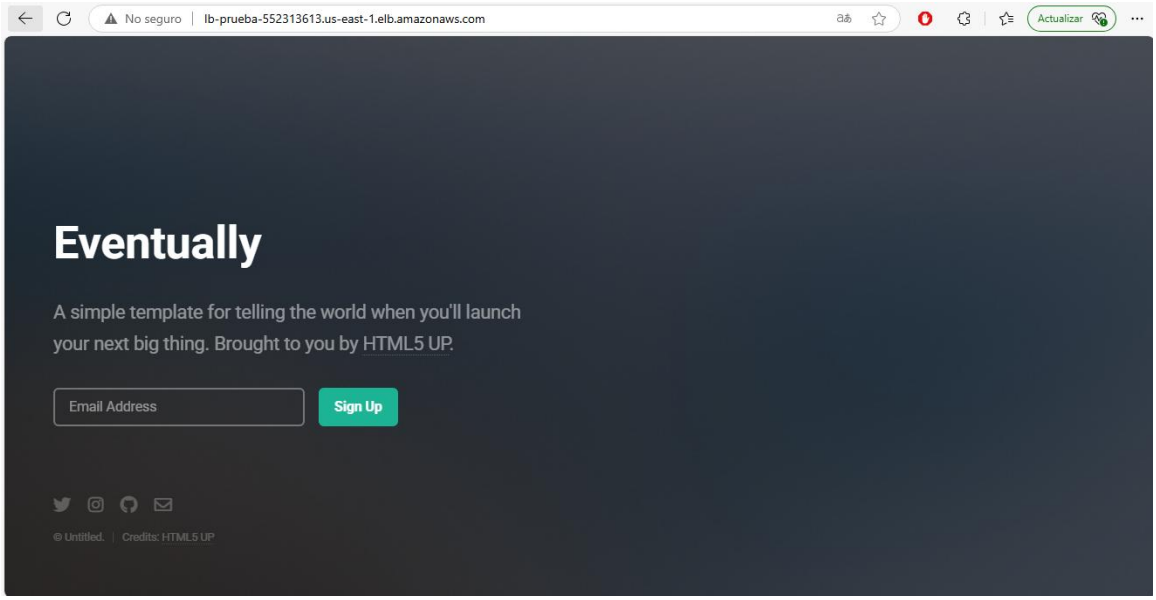
AMI que se creó de la instancia LINUXSERVER1 que había sido configurada con el servicio de apache para la prueba.



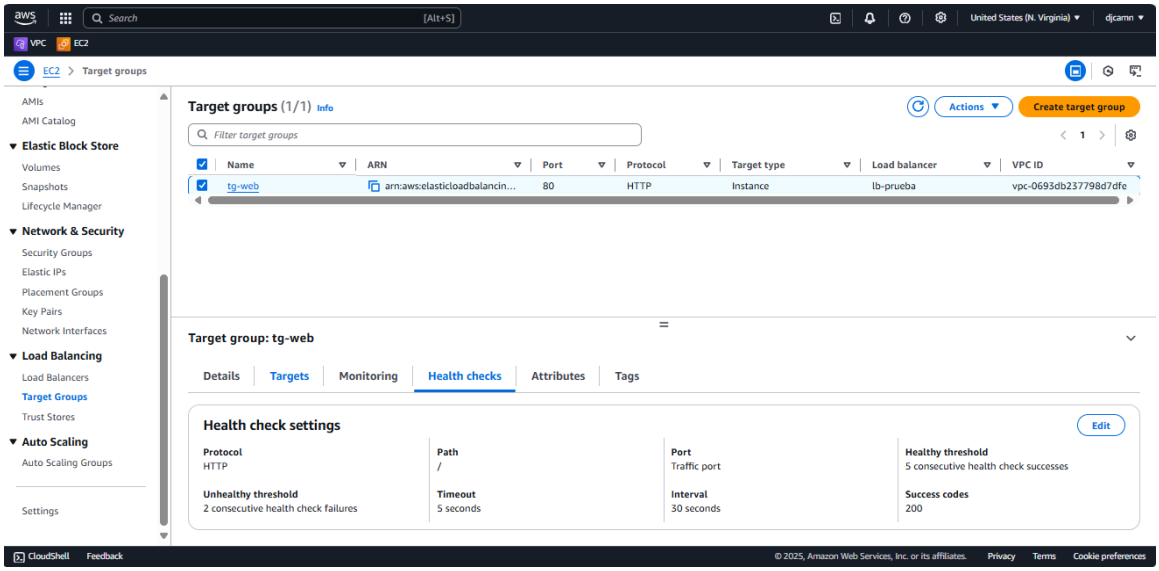
El security group seleccionado para la AMI y el auto escalado fue el security group creado para la instancia de LINUXSERVER1.



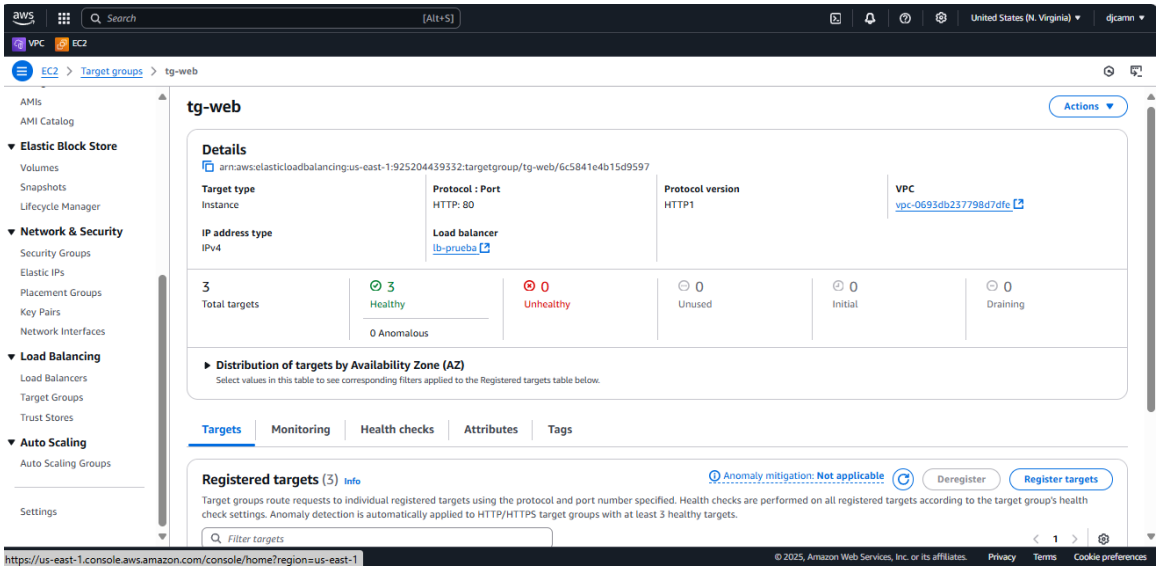
Balanceador de carga creado para la prueba y que se utilizó en el auto escalado, configurado con las vpc privadas y que solo pudiera entrar a través del DNS, ya que las instancias que se crearan con el auto escalado serían creadas con una VPC PRIVADA.



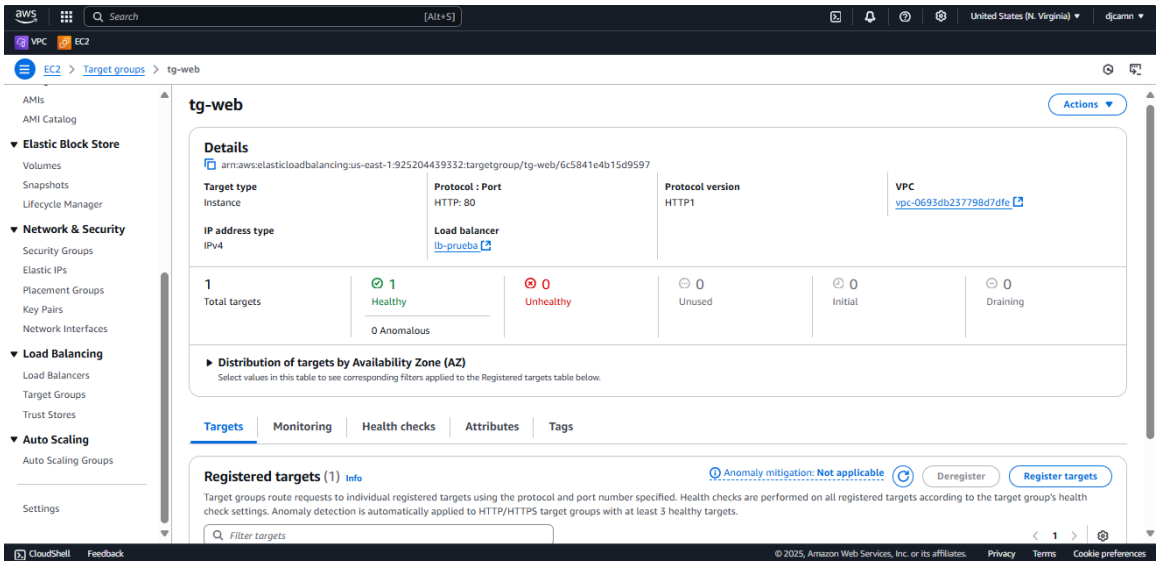
Evidencia de que load balance está enviando a las instancias que tienen configurado el servicio de apache.



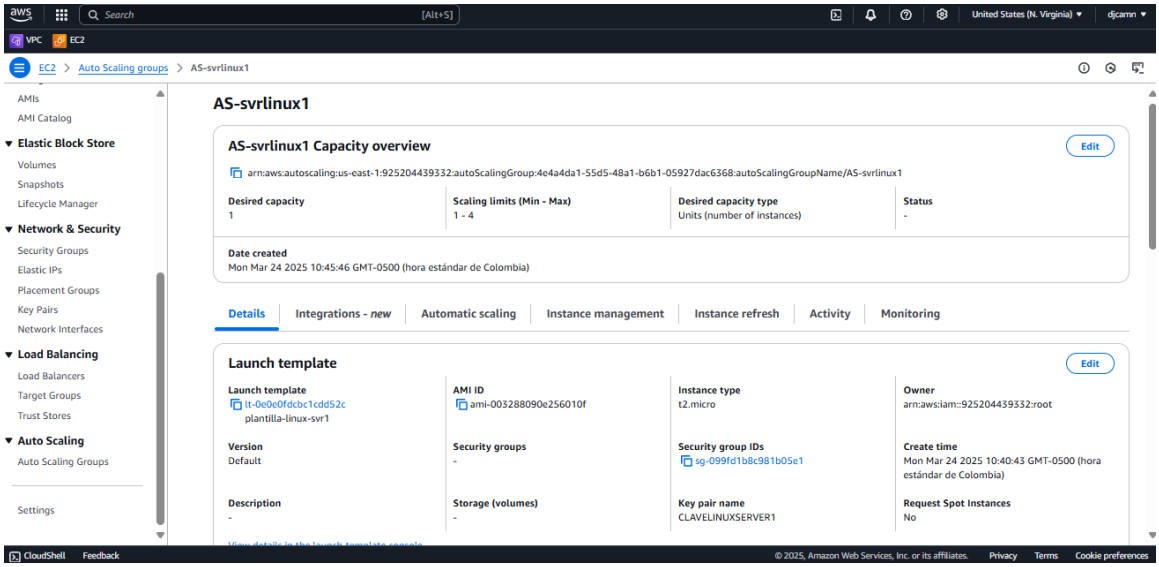
Target group creado para la comunicación, se utilizó con el loab balancer y se configuró la carpeta raíz, el puerto del tráfico, el código de confirmación a recibir, se habilito el puerto 80 http.



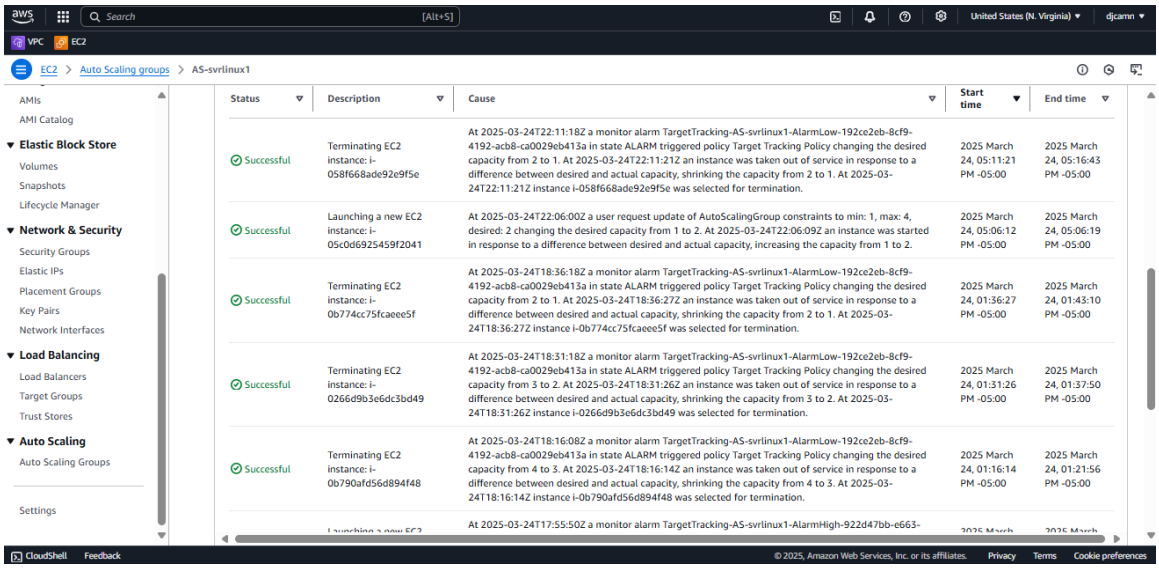
Evidencia de la comprobación que se muestra en el TG creado de las máquinas virtuales que se van creando, en el momento mostraba 3 ya que se había realizado una prueba de estrés de la instancia creada por el servicio de auto escalado.



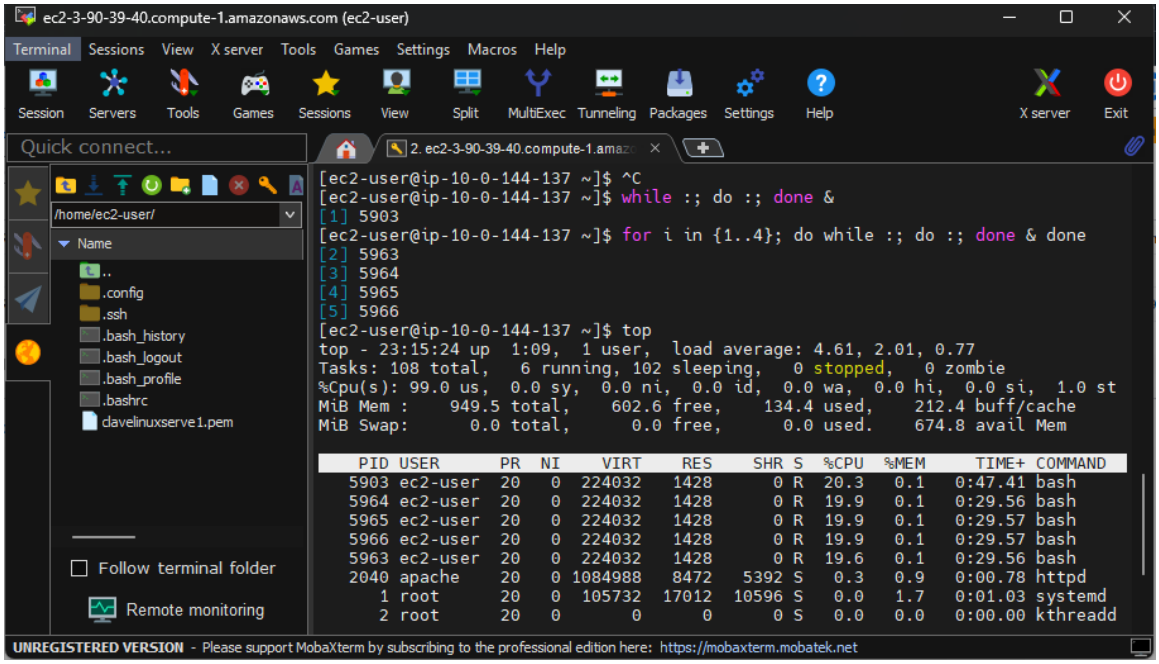
Tg funcionando con la configuración creada con el menor de objetivos creados, se configuro con un mínimo de 1 y un máximo de 4, se cambió para dos y creo una segunda instancia, pero al ser poco el consumo de la CPU se regresó a 1 y la configuración coloco como mínimo 1.



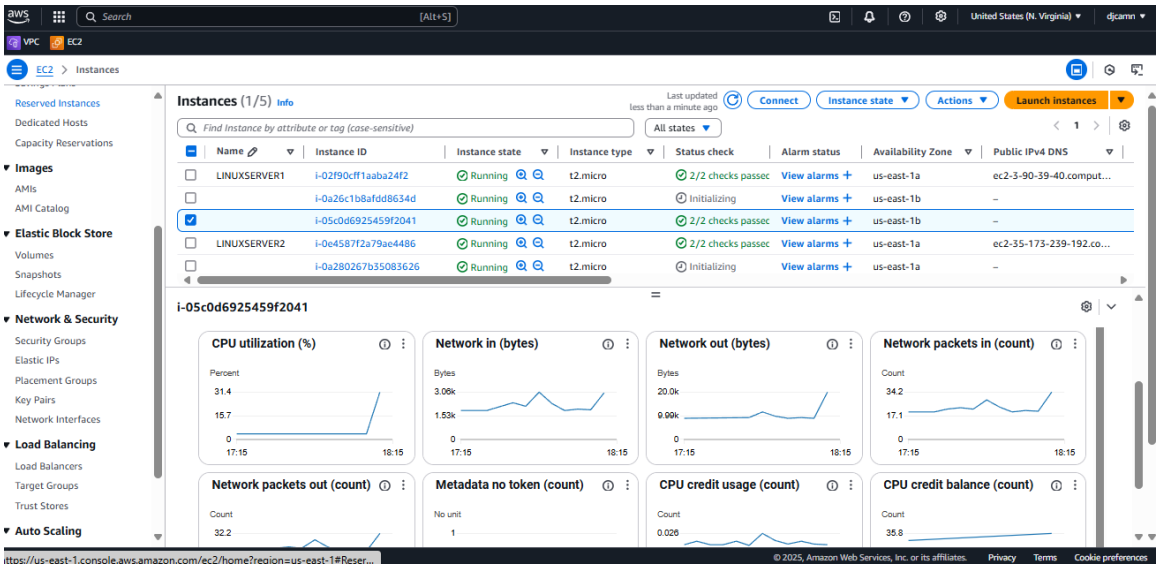
Configuración de auto scaling, de instancias mínimas y máximas agregadas, se configuro para que, si el rendimiento de la CPU aumentaba a más del 40%, para que se creara un máximo de 4 instancias y cuando el rendimiento comience a bajar en la CPU del 40% se comiencen a eliminar las instancias.



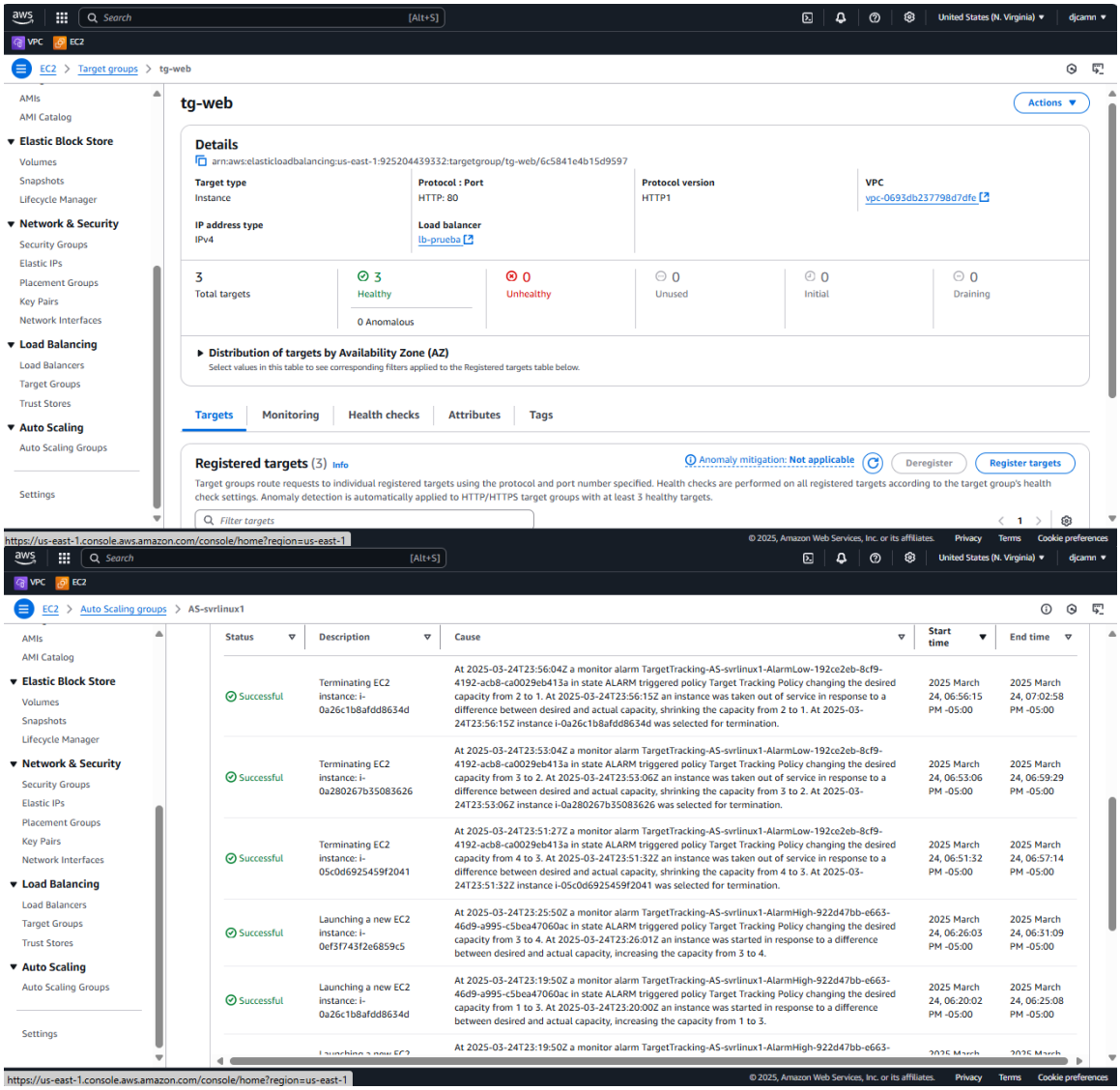
Evidencia de la creación de instancias y eliminación de las instancias de manera automatizada por el auto escaldo.

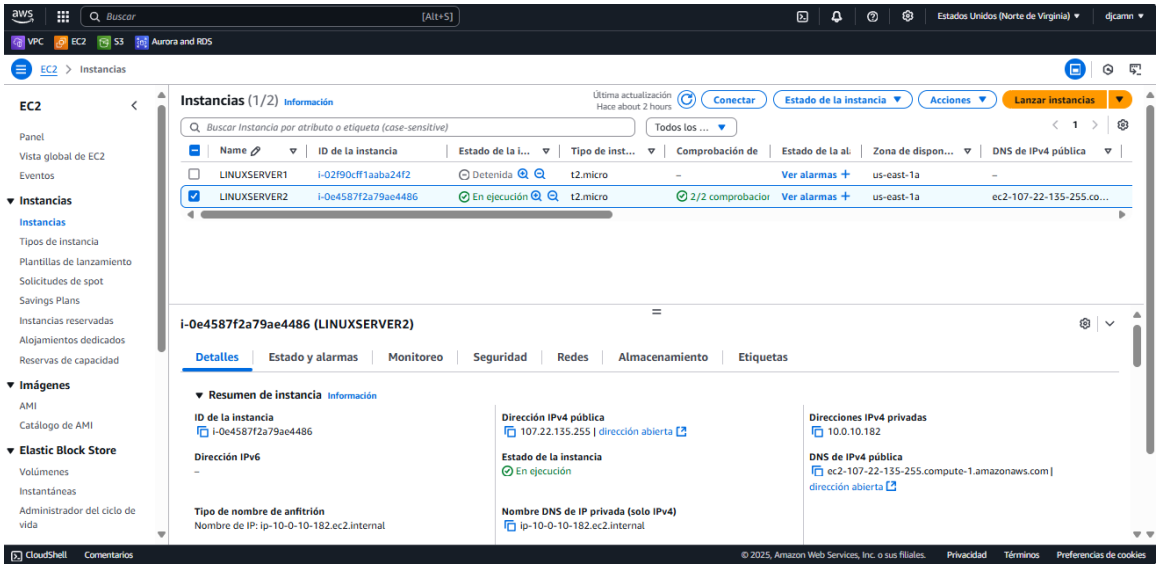


Evidencia de la prueba de estrés realizada al servidor creado por el auto escalado para aumentar la CPU y se crearan otras instancias automáticamente.

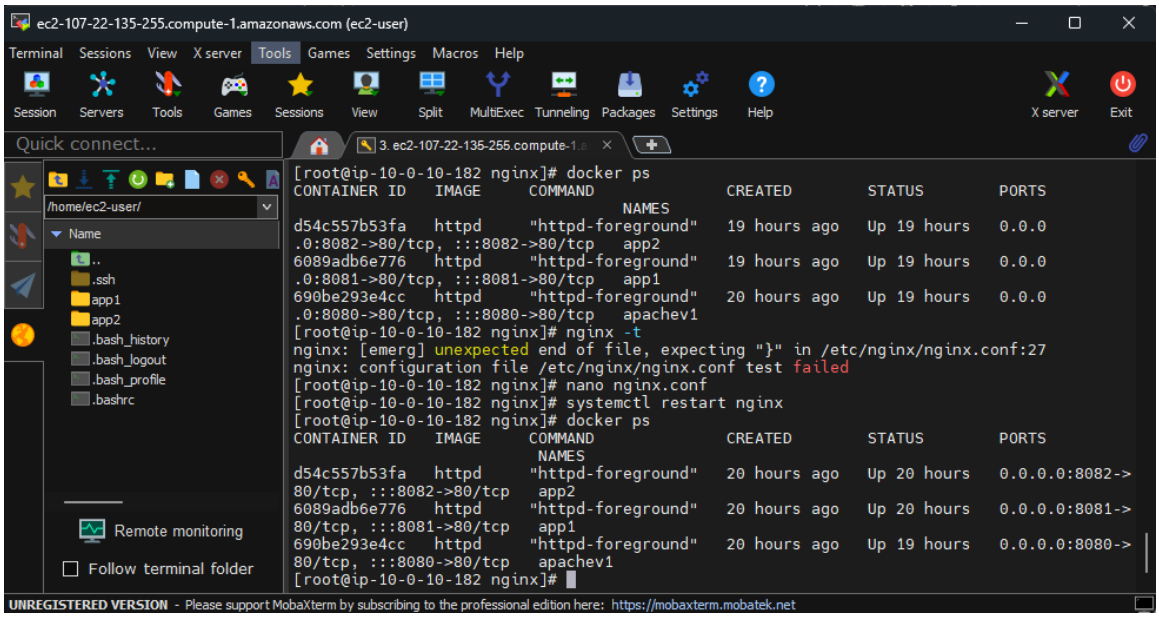


Evidencia de las instancias que se fueron creando, automáticamente por el auto escalamiento.

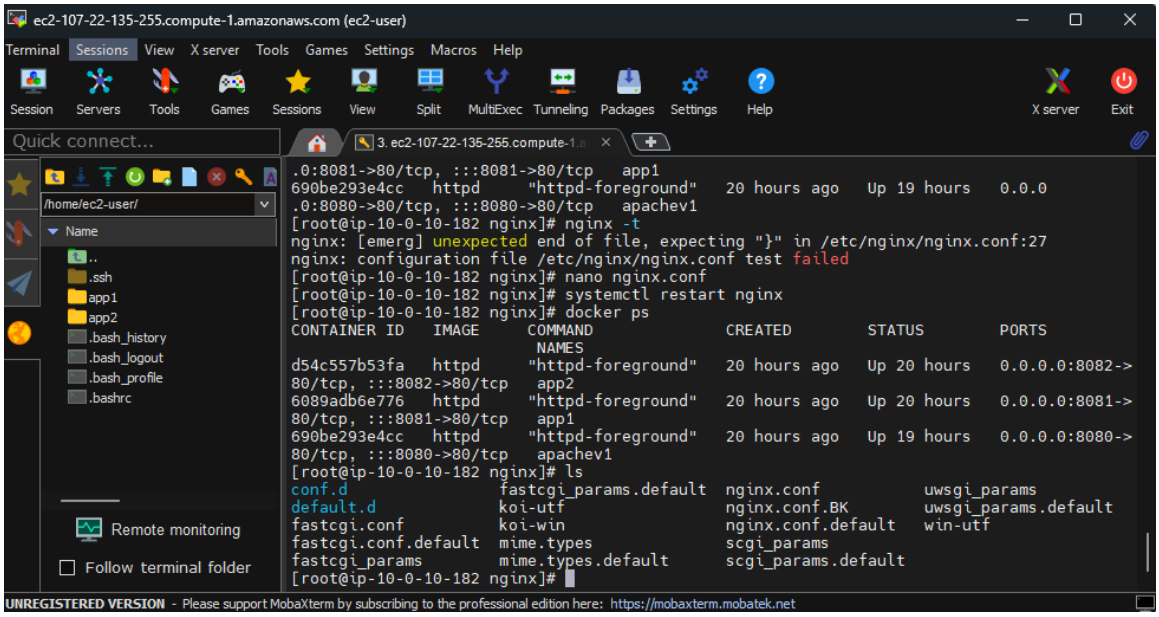

tg-web
Actions
Details
arn:aws:elasticloadbalancing:us-east-1:925204439332:targetgroup/tg-web/6c5841e4b15d9597
Target type
Instance
Protocol : Port
HTTP: 80
Protocol version
HTTP1
VPC
vpc-0693db237798d7dfe
IP address type
IPv4
Load balancer
lb-prueba
3
Total targets
3
Healthy
0 Anomalous
0
Unhealthy
0
Unused
0
Initial
0
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags
Registered targets (3) Info
Anomaly mitigation: Not applicable
Deregister
Register targets
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.
Filter targets
EC2 > Target groups > tg-web
EC2 > Auto Scaling groups > AS-svrlinux1
Status
Description
Cause
Start time
End time
Successful
Terminating EC2 instance: i-0a26c1b8afdd8634d
At 2025-03-24T23:56:04Z a monitor alarm TargetTracking-AS-svrlinux1-AlarmLow-192ce2eb-8cf9-4192-acb8-ca0029eb413a in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 2 to 1. At 2025-03-24T23:56:15Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 2 to 1. At 2025-03-24T23:56:15Z instance i-0a26c1b8afdd8634d was selected for termination.
2025 March 24, 06:56:15 PM -05:00
2025 March 24, 07:02:58 PM -05:00
Successful
Terminating EC2 instance: i-0a280267b35083626
At 2025-03-24T23:53:04Z a monitor alarm TargetTracking-AS-svrlinux1-AlarmLow-192ce2eb-8cf9-4192-acb8-ca0029eb413a in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 3 to 2. At 2025-03-24T23:53:06Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 3 to 2. At 2025-03-24T23:53:06Z instance i-0a280267b35083626 was selected for termination.
2025 March 24, 06:53:06 PM -05:00
2025 March 24, 06:59:29 PM -05:00
Successful
Terminating EC2 instance: i-05c0d6925459f2041
At 2025-03-24T23:51:27Z a monitor alarm TargetTracking-AS-svrlinux1-AlarmLow-192ce2eb-8cf9-4192-acb8-ca0029eb413a in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 4 to 3. At 2025-03-24T23:51:32Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 4 to 3. At 2025-03-24T23:51:32Z instance i-05c0d6925459f2041 was selected for termination.
2025 March 24, 06:51:32 PM -05:00
2025 March 24, 06:57:14 PM -05:00
Successful
Launching a new EC2 instance: i-0ef3f743f2e6859c5
At 2025-03-24T23:25:50Z a monitor alarm TargetTracking-AS-svrlinux1-AlarmHigh-922d47bb-e663-46d9-a995-c5bea47060ac in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 3 to 4. At 2025-03-24T23:26:01Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 3 to 4.
2025 March 24, 06:26:03 PM -05:00
2025 March 24, 06:31:09 PM -05:00
Successful
Launching a new EC2 instance: i-0a26c1b8afdd8634d
At 2025-03-24T23:19:50Z a monitor alarm TargetTracking-AS-svrlinux1-AlarmHigh-922d47bb-e663-46d9-a995-c5bea47060ac in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 1 to 3. At 2025-03-24T23:20:00Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 1 to 3.
2025 March 24, 06:20:02 PM -05:00
2025 March 24, 06:25:08 PM -05:00
At 2025-03-24T23:19:50Z a monitor alarm TargetTracking-AS-svrlinux1-AlarmHigh-922d47bb-e663-
https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



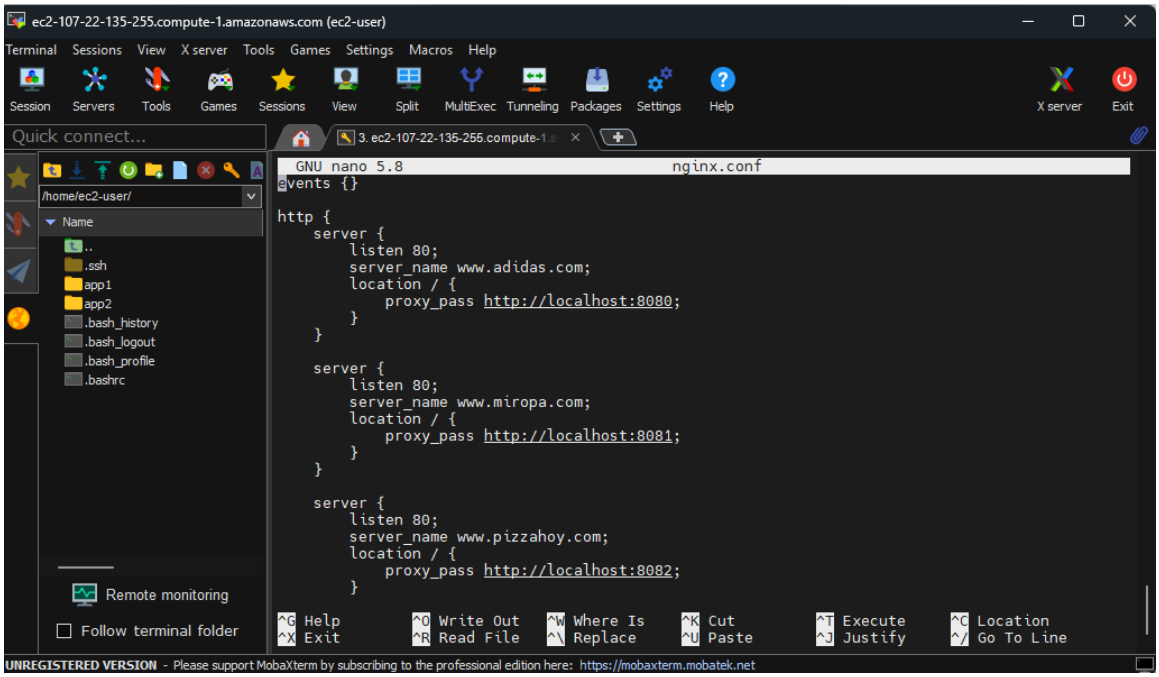
Para la realización de la segunda prueba se utilizó la instancia con el nombre Linuxserver2, una instancia que anteriormente se había creado, pero solo se le había probado con el servicio de httpd, se instaló el servicio de Docker y nginx para poder realizar la prueba.



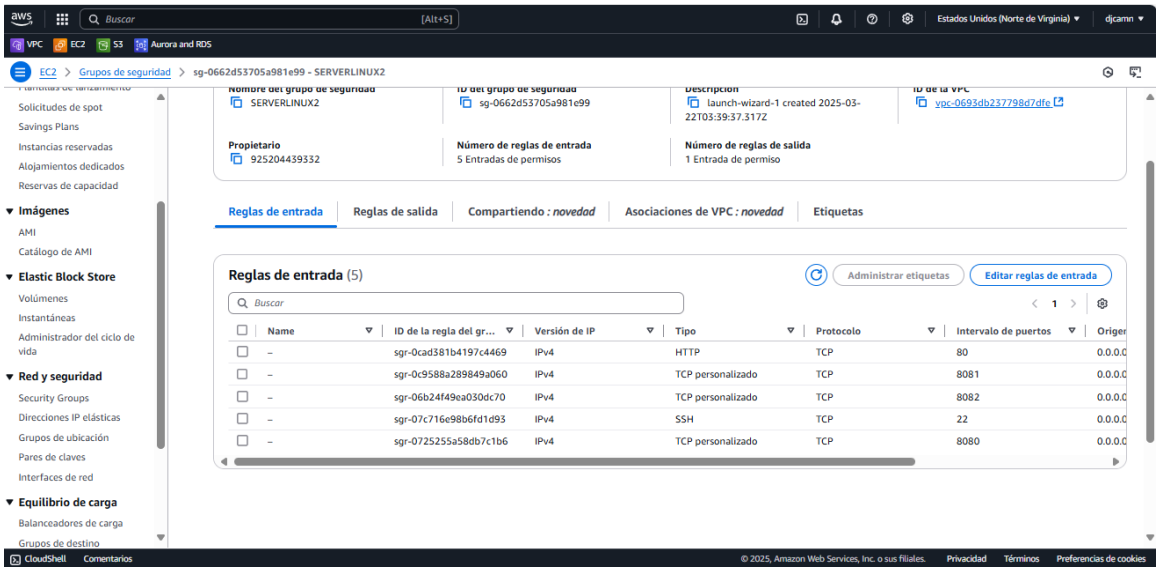
Se crearon 3 contenedores para esta actividad, con diferentes puertos, el contenedor 1, funciona en el puerto 8080, el contenedor 2 está funcionando en el puerto 8081 y el contenedor 3 funcionando en el puerto 8082.



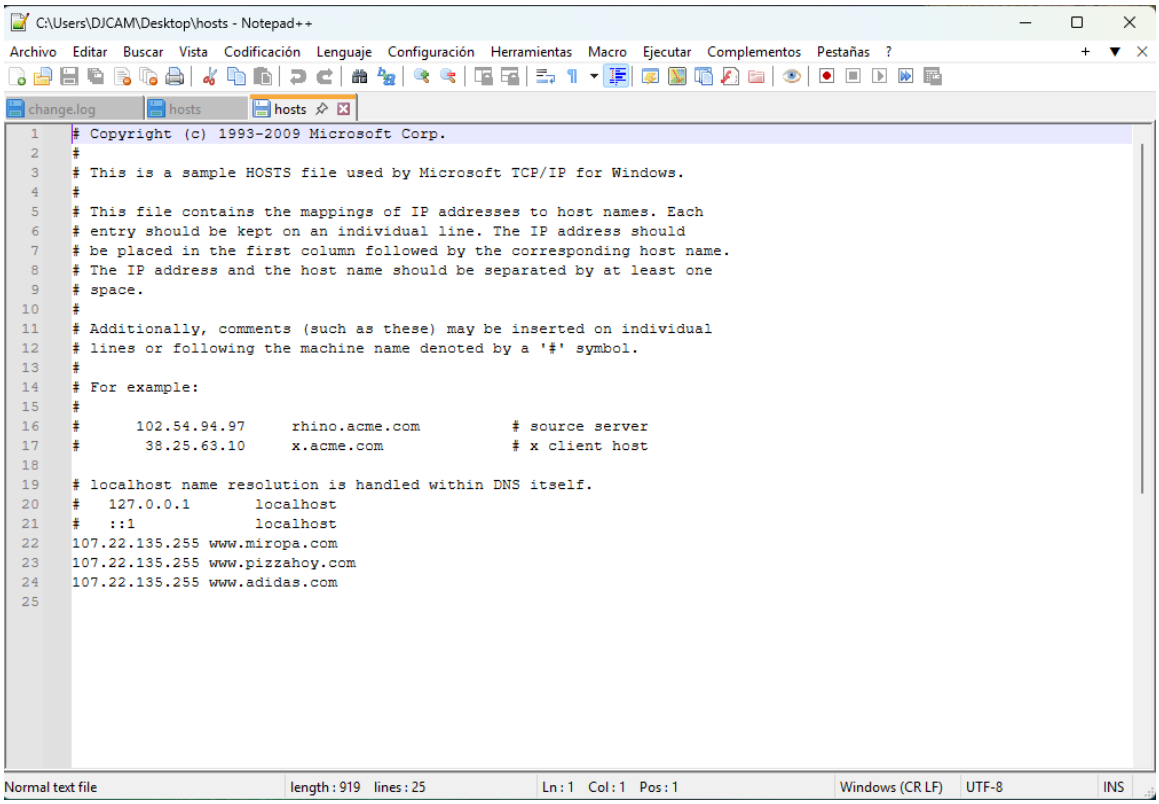
Se configuro el archivo nginx.conf, pero con anterioridad a configurar el archivo se creó una copia del archivo con el nombre de nginx.conf.BK, similar al nombre explicado en la clase.



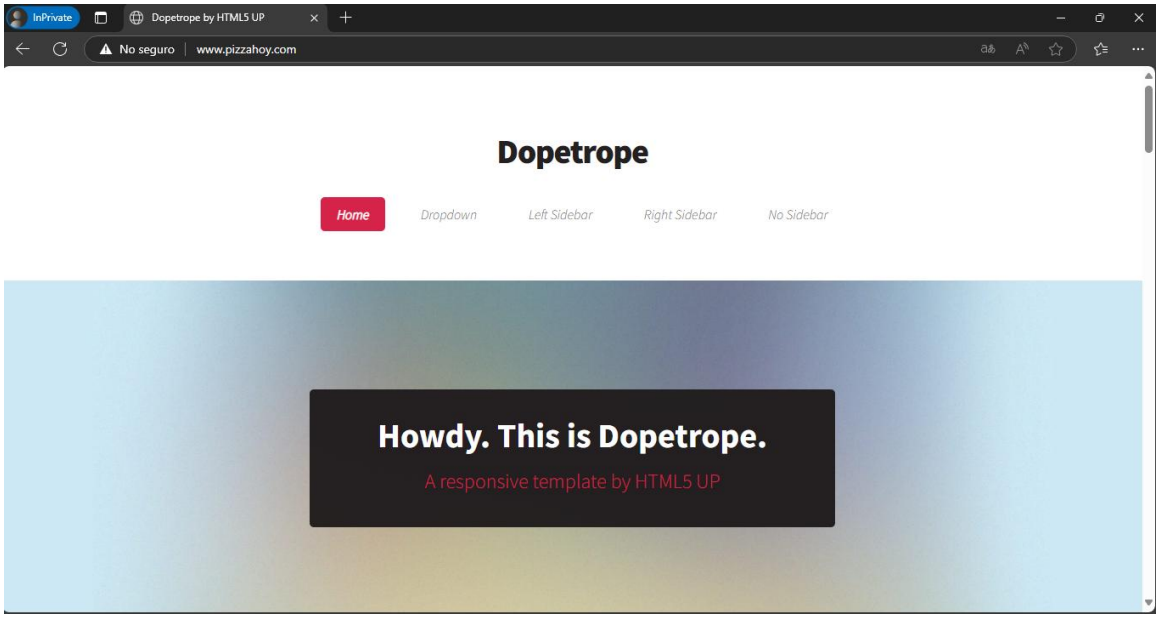
Se utilizaron las siguientes líneas de comando en el documento nginx.conf para hacer el redireccionamiento para que la IP publica de la instancia mostrara lo que estaba almacenado en los distintos contenedores, dependiendo el nombre que se colocara en el navegador, para que el proxy reverso haga el trabajo de redireccionar. Los nombres que se utilizaron para la prueba fueron: www.adidas.com, www.miropa.com y www.pizzahoy.com.



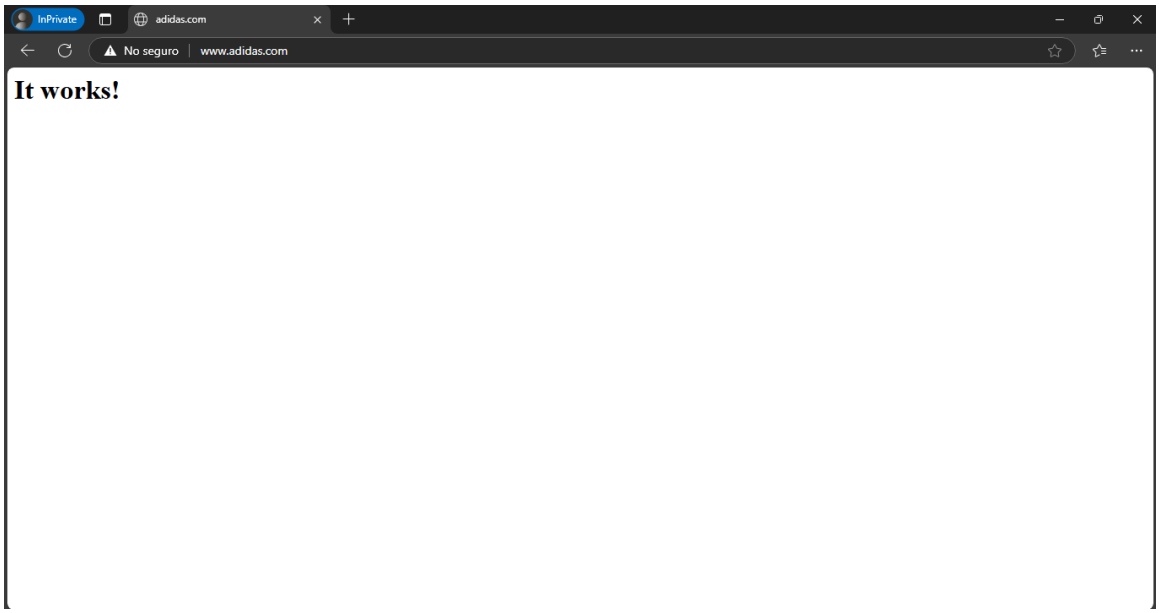
Configuración de las reglas de entrada, donde se adicionaron los puertos por los cuales se iba hacer las peticiones para acceder a los sitios que se configuraron en los contenedores.



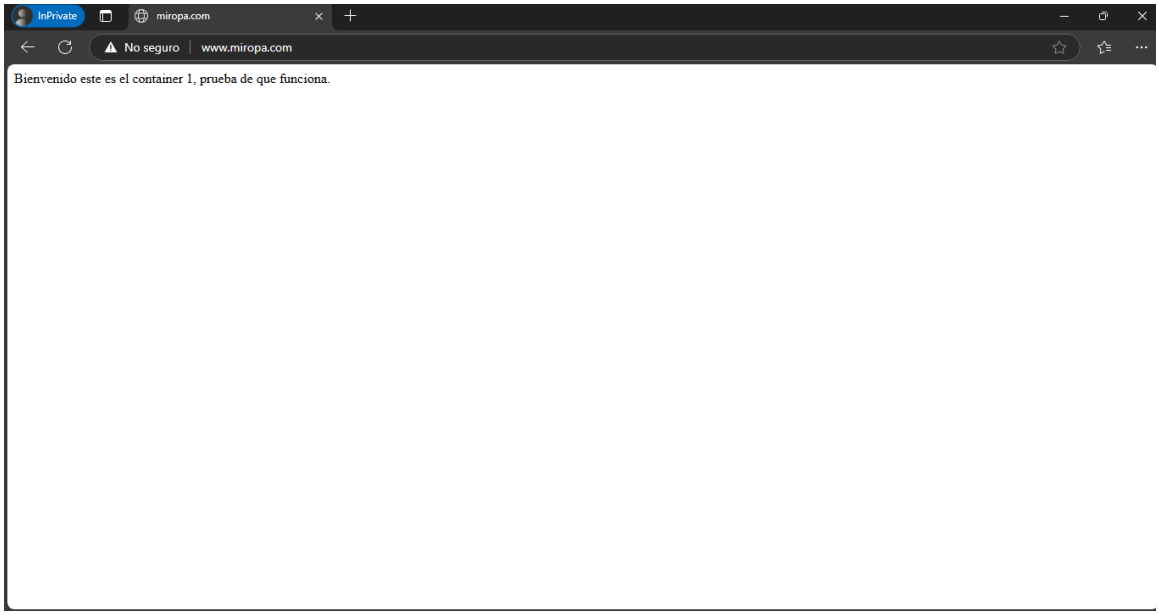
Cambios hechos en el archivo host del sistema operativo local para la prueba, en el cual se incluyeron los nombres a los que irían a la IP publica de la instancia.



Evidencia de resultado de la prueba con la dirección www.pizzahoy.com.



Evidencia de la prueba realizada con la dirección www.adidas.com.



Evidencia de la prueba realizada con la dirección www.miropa.com.

Conclusiones

Con el presente trabajo, pude evidenciar los conocimientos que adquirir sobre lo que es Amazon web services, conocer y aprender todo lo que la plataforma de AWS permite hacer, y algunas funcionalidades que pueden ayudar a una empresa a economizar en costos gracias a la gran variedad de funciones en la nube que tiene la plataforma, evitando que una empresa tenga que infraestructura en infraestructura física, se puede aprender más sobre la computación en la nube, sobre su historia, además gracias al seminario y a las diversas actividades realizadas en el mismo, pude poner en práctica los conocimientos adquiridos en los diferentes temas tratados en el seminario.

Referencias

10code. (10 de 06 de 2024). Obtenido de 10code.es: https://10code.es/docker-para-que-sirve/

akamai. (s.f.). Obtenido de www.akamia.com: https://www.akamai.com/es/glossary/what-are-ports

AWS Amazon. (s.f.). Obtenido de AWS.amazon.com: https://aws.amazon.com/es/what-is/virtualization/

AWS Amazon. (s.f.). Obtenido de aws.amazon.com: https://aws.amazon.com/es/elasticloadbalancing/#:~:text=Elastic%20Load%20Balancing%20%28ELB%29%20distribuye%20autom%C3%A1ticamente%20el%20tr%C3%A1fico,en%20una%20o%20varias%20zonas%20de%20disponibilidad%20%28AZ%29.

B, D. (18 de 03 de 2025). *hostinger*. Obtenido de www.hostinger.com: https://www.hostinger.com/es/tutoriales/como-configurar-proxy-inverso-nginx

B, G. (13 de 03 de 2025). *hostinger*. Obtenido de www.hostinger.com: https://www.hostinger.com/es/tutoriales/que-es-nginx

documentación AWS. (s.f.). Obtenido de docs.aws.amazon.com: https://docs.aws.amazon.com/es_es/vpc/latest/userguide/how-it-works.html

Documentación aws. (s.f.). Obtenido de docs.aws.amazon.com: https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/Instances.html

infranet working. (20 de 06 de 2017). Obtenido de blog.infranetworking.com: https://blog.infranetworking.com/computacion-en-la-nube-la-historia/

Marujita. (07 de 07 de 2023). *muy tecnológicos*. Obtenido de muytecnologicos.com: https://muytecnologicos.com/diccionario-tecnologico/ip-publica

Marujita. (07 de 07 de 2023). *muy tecnológicos*. Obtenido de muytecnologicos.com: https://muytecnologicos.com/diccionario-tecnologico/amazon-web-services-aws

paradigma digital. (04 de 04 de 2019). Obtenido de www.paradigmadigital.com: https://www.paradigmadigital.com/dev/comparativa-servicios-cloud-aws-azure-gcp/

redhat. (09 de 12 de 2024). Obtenido de www.redhat.com: https://www.redhat.com/es/topics/virtualization/what-is-virtualization

tarlogic. (s.f.). Obtenido de www.tarlogic.com: https://www.tarlogic.com/es/glosario-ciberseguridad/instancia-amazon-ec2/

time toast. (s.f.). Obtenido de www.timetoast.com: https://www.timetoast.com/timelines/linea-de-tiempo-de-la-evolucion-de-la-computacion-en-la-nube

wikipedia. (s.f.). Obtenido de www.wikipedia.com: https://es.m.wikipedia.org/wiki/computacion_en_la_nube

Zettle, K. (s.f.). *www.atlassian.com*. Obtenido de atlassian: https://www.atlassian.com/es/microservices/cloud-computing#:~:text=La%20computaci%C3%B3n%20en%20la%20nube%20es%2

0la%20entrega,de%20software%2C%20a%20trav%C3%A9s%20de%20Internet%20%28la%20nube%29.