TRABAJO DE GRADO
Opción Seminario-Diplomado.

Título

Diseño y Automatización de Servicios AWS con EC2 y Docker

Corporación Universitaria Remington

Facultad de Ingenierías

Ingeniería de Sistemas

Autores del trabajo de grado:

Sergio Andrés Gamboa Torres

Luis Miguel Bertel Berrío

Tutor del trabajo de grado:

Juan Pablo Berrio López

2025

Dedicatoria

Dedicamos este trabajo a nuestras familias, por su apoyo incondicional y constante motivación durante todo el proceso académico.

Agradecimientos

Agradecemos al docente Juan Pablo Berrio López por su orientación y acompañamiento durante el desarrollo de este trabajo. También extendemos nuestra gratitud a la Corporación Universitaria Remington por brindarnos los espacios y recursos para nuestra formación profesional.

Contenido

Lista de Figuras

Lista de Tablas

Resumen

El presente trabajo de grado aborda el diseño y la implementación de una red en la nube sobre Amazon Web Services (AWS). Para ello se despliegan dos instancias EC2, una con sistema operativo Windows y otra con Linux, configuradas con un servidor web funcional (IIS para Windows y Apache para Linux), permitiendo el acceso público y conectividad interna en una VPC. Se crean subredes públicas, un Internet Gateway, y se generan reglas en los grupos de seguridad específicos que permiten tráfico HTTP desde cualquier dirección IP mientras que el acceso remoto (RDP y SSH) queda restringido únicamente a la IP del usuario.

El trabajo de grado incluye una fase avanzada con la implementación de contenedores Docker para simular la carga web de diferentes tipos, permitiendo ejecutar varios servicios ligeros en el mismo equipo y resulta en la demostración de que los contenedores son más eficientes y escalables que la virtualización convencional.

Esta solución práctica permite llevar a cabo una validación de conceptos clave en la infraestructura como servicio (IaaS), las redes virtuales, los protocolos de seguridad, los contenedores y la administración en la nube. Se concluye que AWS proporciona un entorno capaz de implementar arquitecturas mixtas y que, además, incrementa las habilidades técnicas que se requieren en el escenario profesional actual.

Palabras clave

AWS, EC2, VPC, Docker, Infraestructura como servicio

1. Marco conceptual y contextual

El siguiente trabajo queda encuadrado dentro del Seminario de grado sobre Amazon Web Services (AWS) cuya finalidad era proporcionar a los alumnos conocimientos prácticos de infraestructura como servicio (IaaS), redes virtuales y servicios en la nube. En este sentido se expone el desarrollo de un proyecto técnico cuyo objetivo principal fue diseñar, desplegar y configurar una red que permita acceder a los servicios web que se encuentran hospedados en instancias EC2 con sistemas operativos Linux y Windows y que, además, integre las tecnologías de contenedores que permiten simular cargas de trabajo (Docker).

AWS, conceptualmente, constituye la plataforma de desarrollo más potente de la computación en la nube. Esta proporciona servicios bajo demanda para aprovisionar recursos informáticos de forma flexible, escalable y con eficiencia de costos (Amazon Web Services, 2023). La utilización de EC2 (Elastic Compute Cloud) permite una implementación de máquinas virtuales personalizadas para diferentes sistemas operativos y el empleo de VPC (Virtual Private Cloud) permite implementar entornos de red aislados lógicamente en la nube teniendo control sobre el direccionamiento IP, subredes, tablas de enrutamiento y gateways (Hwang, Dongarra & Fox, 2013).

Desde la perspectiva de la seguridad en la nube, es indispensable el aplicar adecuadamente las configuraciones por medio de los grupos de seguridad que funcionan como firewalls virtuales al controlar el tráfico de red que entra y sale de las instancias (Zissis & Lekkas, 2012). El presente trabajo aplica buenas prácticas como el acceso SSH o RDP restringido a la IP del administrador, esto da como resultado la administración segura de los servidores.

Asimismo, el presente trabajo incorpora el uso de contenedores Docker, tecnología que permite ejecutar aplicaciones de manera ligera, portable usando el mismo kernel del sistema operativo del huésped. Esto implica un ahorro de recursos en comparación con la virtualización clásica (Merkel, 2014), siendo la tecnología ideal para pruebas de carga, escalado horizontal y despliegue continuo de los servicios web.

En contexto, el presente trabajo se desarrolla como ejercicio académico simulado, orientado a tener un refuerzo importante de las competencias de los estudiantes en un entorno real de infraestructura de red. La Corporación Universitaria Remington brinda este espacio formativo por medio del programa de Ingeniería de Sistemas, capacitando a los estudiantes para dar respuesta a problemáticas de alta demanda de la tecnología en entornos corporativos y de negocio donde las soluciones en la nube son cada vez más demandadas.

La experiencia práctica adquirida mediante esta implementación permite a los participantes comprender cómo construir soluciones robustas, seguras y escalables en la nube, alineadas con los estándares y tendencias actuales de la industria.

2. Objetivos

2.1 Objetivo General:

Diseñar, desplegar y documentar una red en AWS que incluya dos instancias EC2 (una Windows y una Linux), asegurando su accesibilidad pública, conectividad entre ellas y la instalación de un servidor web funcional en cada instancia.

2.2 Objetivos Específicos:

Configurar correctamente la red en AWS mediante una VPC y subredes públicas.

Implementar reglas de seguridad para permitir el acceso remoto seguro (SSH, RDP) y tráfico HTTP.

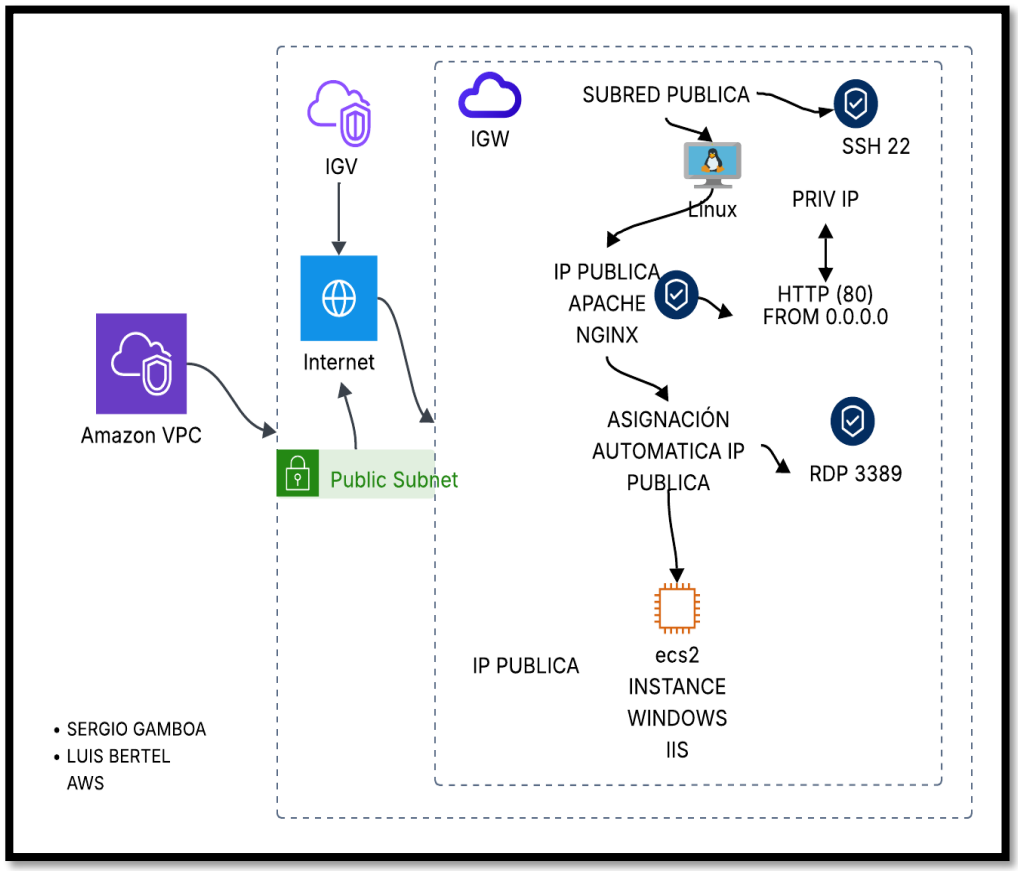
Instalar y configurar servidores web IIS en Windows y Apache en Linux.

Validar el correcto funcionamiento de los servicios mediante pruebas de conectividad y accesibilidad.

3. Desarrollo e implementación del aprendizaje

Figura 1

Diagrama



Nota. Fuente. Elaboración propia 2025.

3.1 Descripción de la arquitectura:

Red Creada:

Se ha utilizado una VPC, ya sea la predeterminada o una personalizada, como red virtual aislada. Dentro de ella, se configuró al menos una subred pública en cada zona de disponibilidad donde se desplegaron las instancias. Para garantizar el acceso a Internet, estas subredes están definidas como públicas, lo que significa que su tabla de rutas incluye una entrada que dirige el tráfico hacia un Internet Gateway.

Cada instancia recibe automáticamente una dirección IP privada al estar dentro de una subred. Además, si se utiliza la VPC por defecto, esta configuración permite la asignación automática de una IP pública al momento del lanzamiento.

Se ha asociado un Internet Gateway a la VPC para habilitar la conexión de las instancias con la red pública. Las instancias EC2 seleccionadas son del tipo t2.micro, de bajo costo y aptas para el nivel gratuito de AWS (Free Tier), y cuentan con una capacidad de 8 GB adecuada para pruebas básicas. Se eligió Windows Server 2016 Base para la instancia Windows y una Amazon Machine Image (AMI) con kernel 6.1 Linux 2023 para la instancia Linux. En ambas instancias se instalaron los servidores web Apache y Nginx, utilizando MobaXterm para la gestión.

En cuanto a la seguridad, se definieron reglas estrictas en los grupos de seguridad: se permite el acceso RDP (puerto 3389) únicamente desde la IP pública del usuario hacia la instancia con Windows, y el acceso SSH (puerto 22) también solo desde la IP pública del usuario para la instancia Linux. El tráfico HTTP (puerto 80), en cambio, se habilitó para cualquier origen (0.0.0.0/0) en ambas instancias.

Adicionalmente, se suele permitir temporalmente el tráfico ICMP (ping) entre las instancias para comprobar la conectividad interna. En resumen, el uso de una VPC con subredes públicas y un Internet Gateway permite el acceso externo, mientras que los grupos de seguridad regulan estrictamente el tráfico de entrada según protocolos y direcciones IP permitidas.

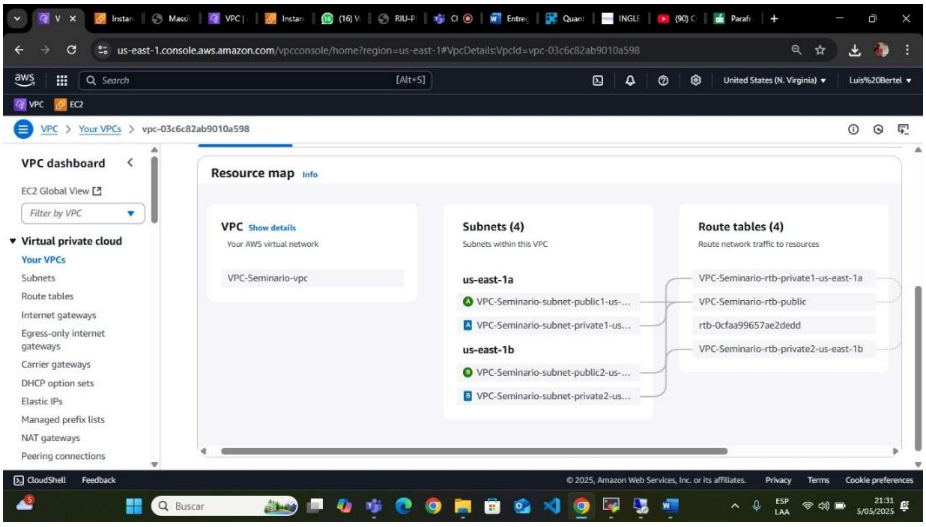
3.2 Configuraciones realizadas:

Creación VPC

Al momento de crear una nube privada virtual, es fundamental establecer su propósito. En este caso, se necesita acceso remoto desde cualquier ubicación a través de internet, por lo que es indispensable habilitar tanto la resolución DNS como las redes públicas.

Figura 2

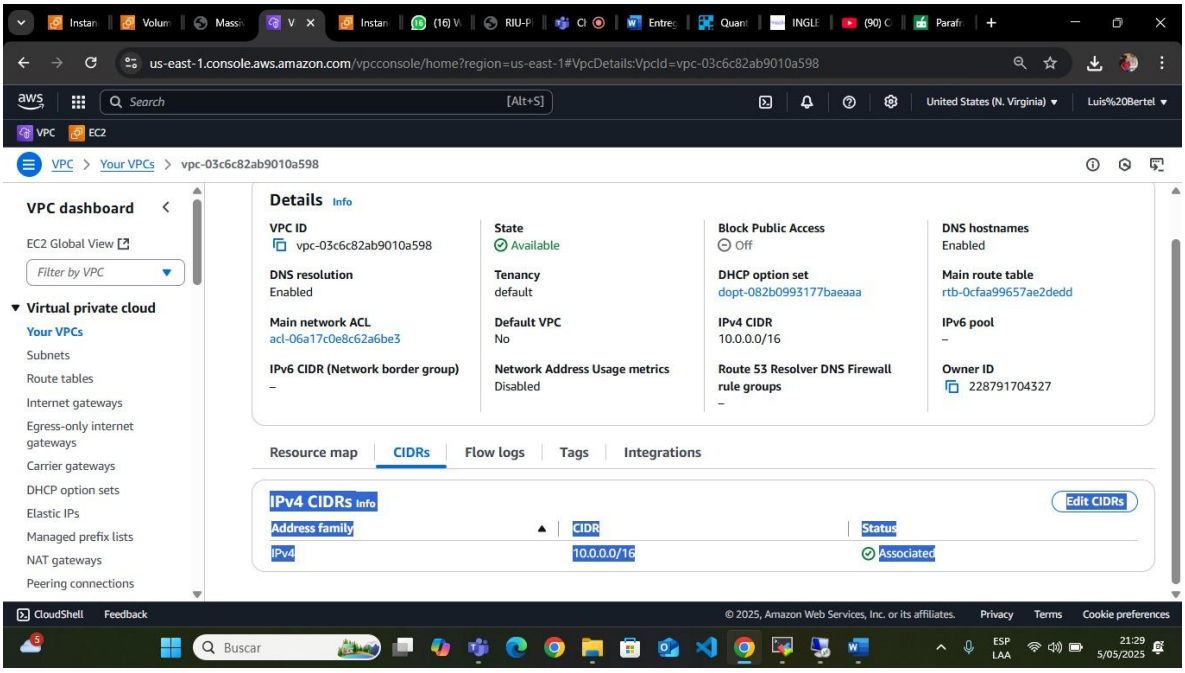
Creación VPC



Nota. Fuente. Elaboración propia 2025.

Figura 3

Creación grupo de seguridad

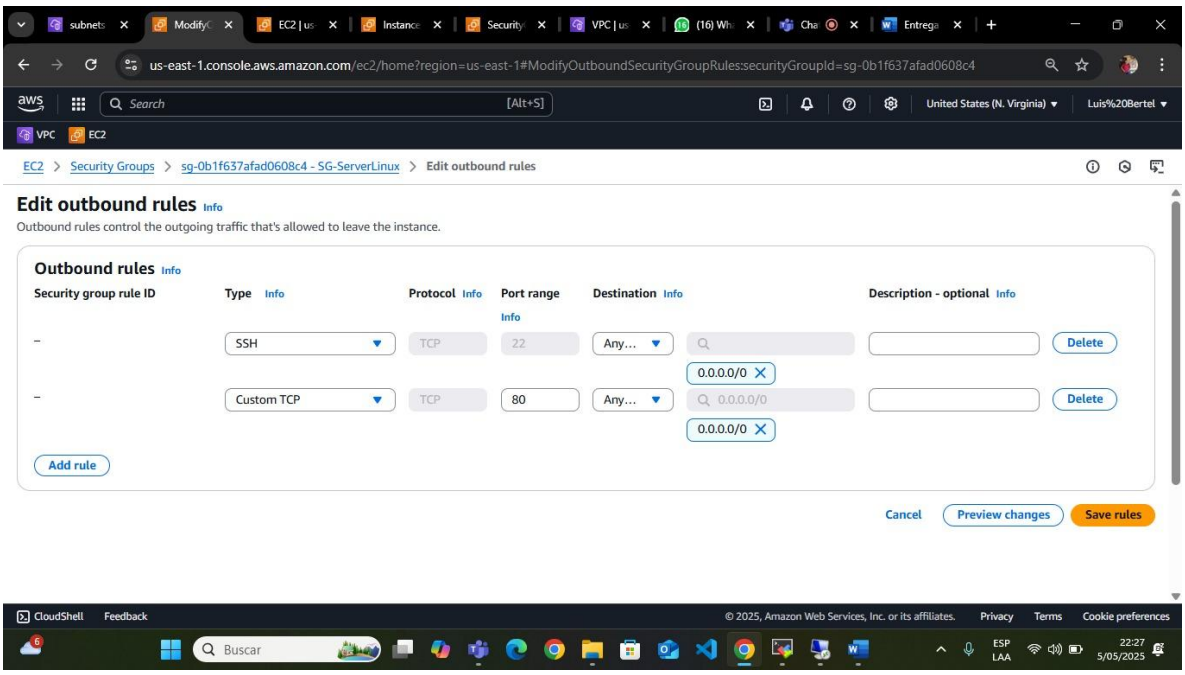


Nota. Fuente. Elaboración propia 2025.

Para establecer cualquier tipo de conexión con la nube creada o con las instancias configuradas, es indispensable el uso de protocolos y puertos. Por esta razón, es fundamental definir grupos de seguridad que permitan gestionar y controlar dichas conexiones. Al crear estos grupos de seguridad, es necesario configurar en las reglas de entrada el puerto SSH (22), el cual permite el acceso remoto al servidor o instancia. Además, para habilitar el acceso a sitios web, se deben establecer reglas que permitan conexiones HTTP o HTTPS (seguras), especificando el puerto correspondiente para recibir las solicitudes:

Figura 4

Creación de Instancias

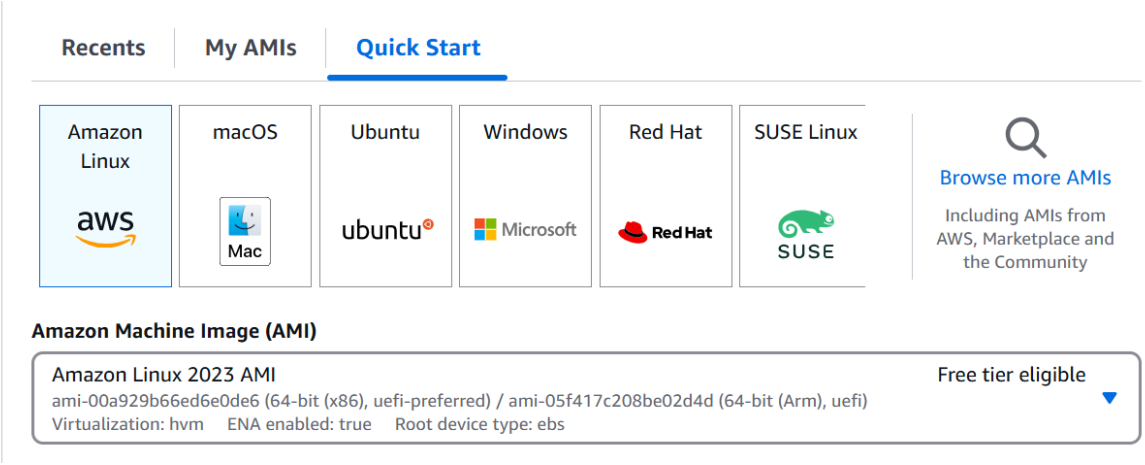


Nota. Fuente. Elaboración propia 2025.

Crear una instancia consiste en desplegar una máquina virtual que opera dentro de un entorno "virtual", con un sistema operativo seleccionado y recursos asignados según las necesidades específicas. Es fundamental tener definido el propósito y el alcance que se le dará a dicha instancia, ya que esto permitirá elegir el sistema operativo y las características más adecuadas. En este ejercicio práctico, utilizaremos recursos básicos y el sistema operativo será Linux.

Figura 5

Linux.

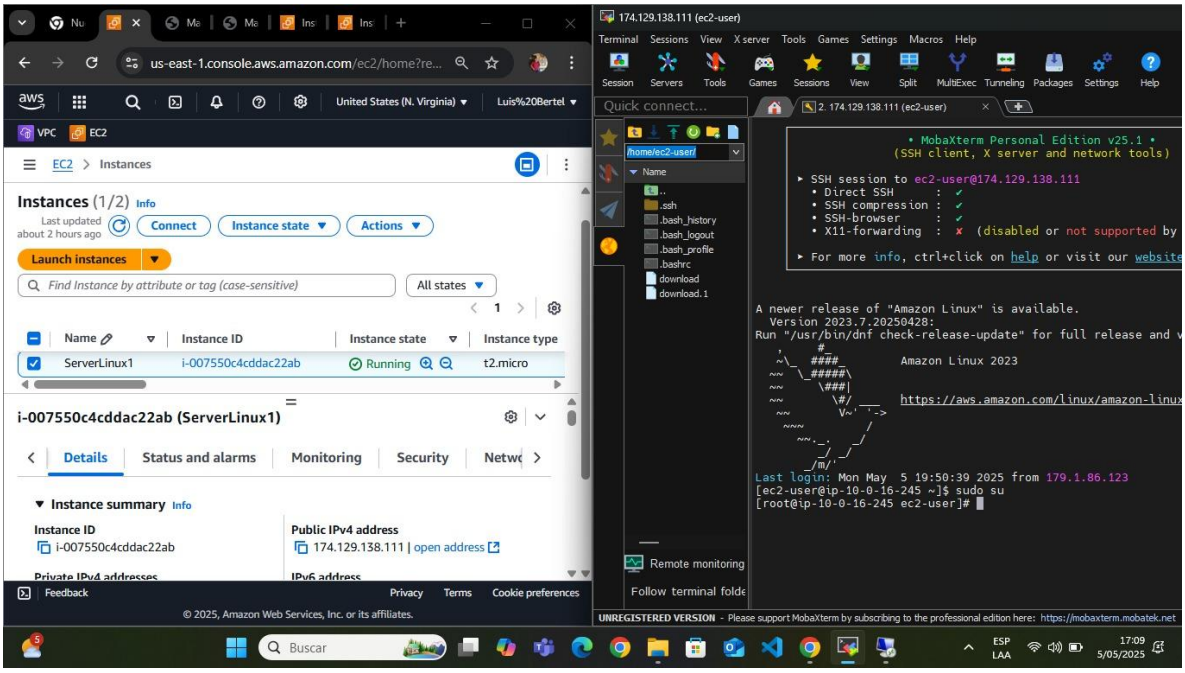


Nota. Fuente. Elaboración propia 2025.

Para que la instancia pueda ser accedida desde internet, es necesario habilitar la dirección IP pública, ubicarla en una subred pública y asegurarse de que su grupo de seguridad permita el tráfico de entrada a través de los puertos utilizados por la instancia. En este caso, se utilizó el grupo de seguridad configurado previamente.

Figura 6

Creación de Instancias



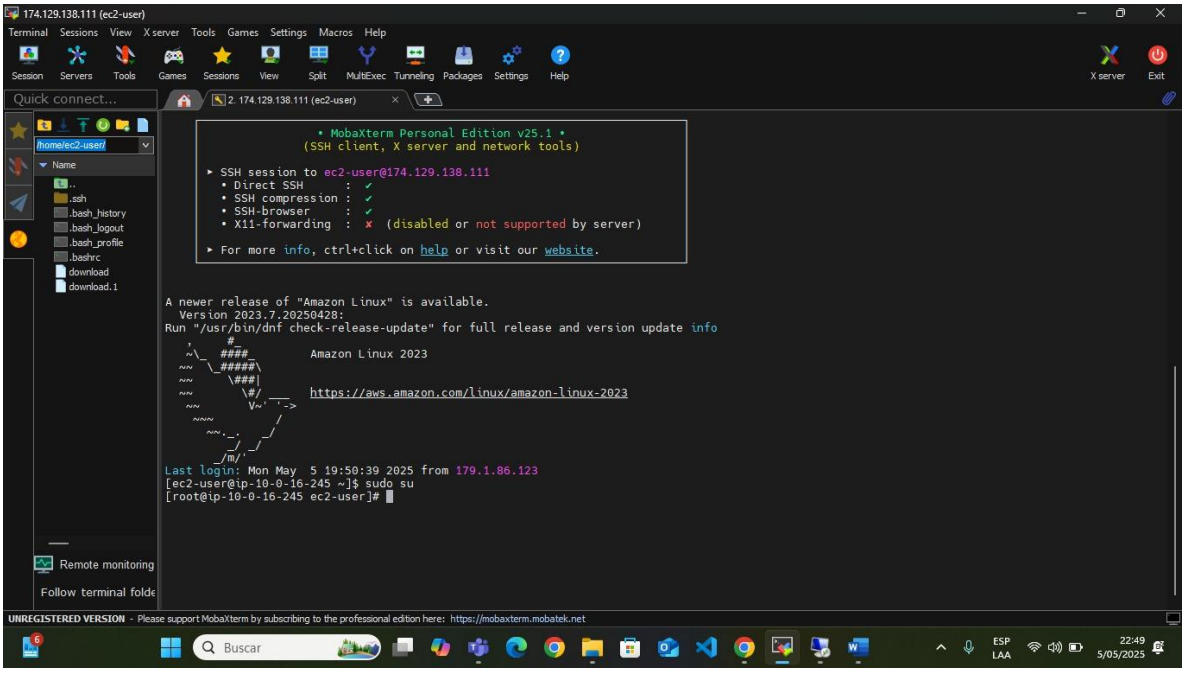
Nota. Fuente. Elaboración propia 2025.

3.3 Procedimiento de acceso:

Se utilizó MobaXterm para establecer la conexión con la instancia previamente creada, empleando el protocolo SSH a través del puerto 22.

Figura 7

Procedimiento de acceso



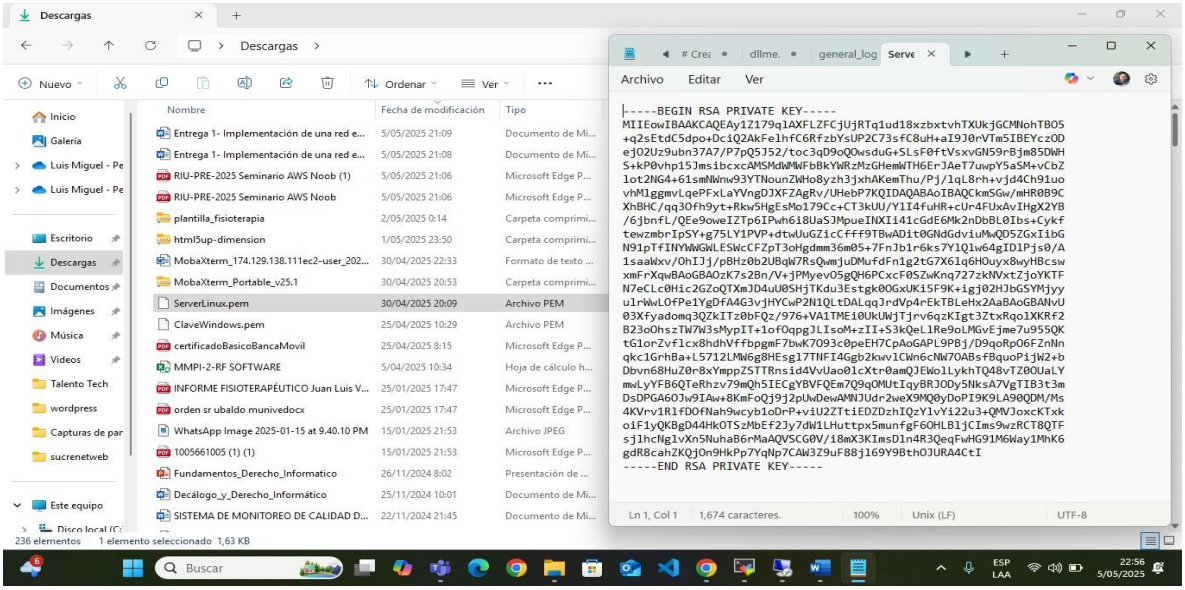
Nota. Fuente. Elaboración propia 2025.

3.4 Consideraciones de seguridad:

Las claves privadas (.pem) deben resguardarse adecuadamente y no divulgarse, con el fin de prevenir accesos no autorizados.

Figura 8

Consideraciones de seguridad



Nota. Fuente. Elaboración propia 2025.

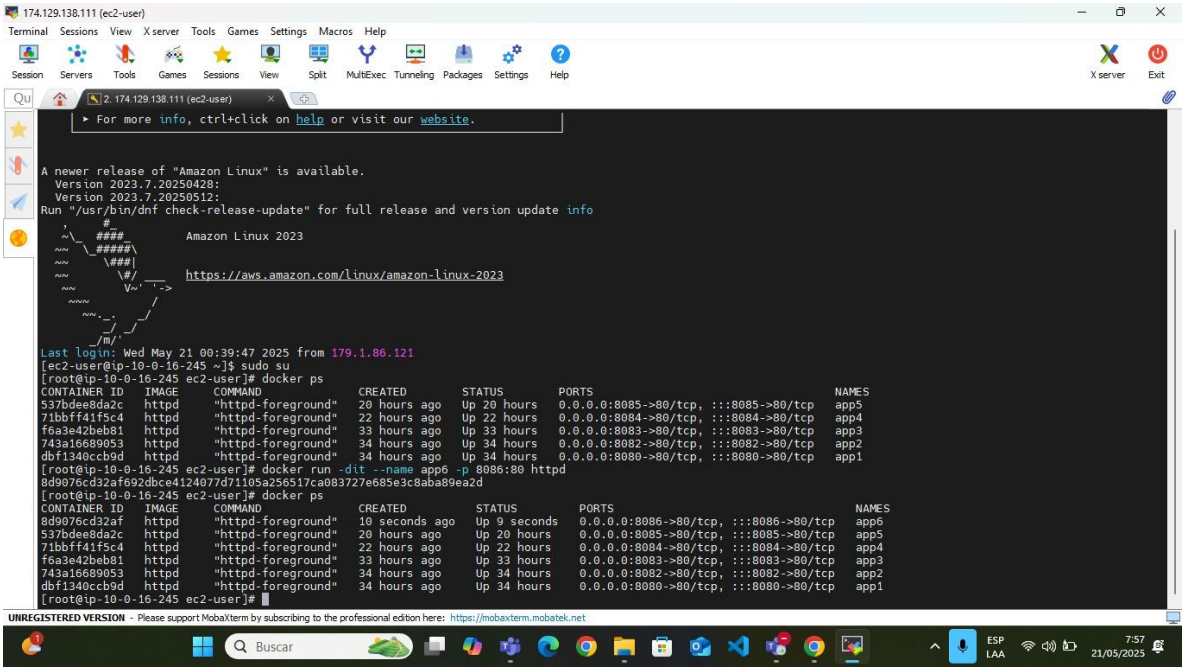
Link video explicando la otra parte y ejecución:

https://youtu.be/8o3zhY8WGNw

Se implementa el servicio de Docker y se crean 6 instancias las cuales funcionan con diferentes puertos desde 8081 hasta 8086.

Figura 9

Servicio de Docker

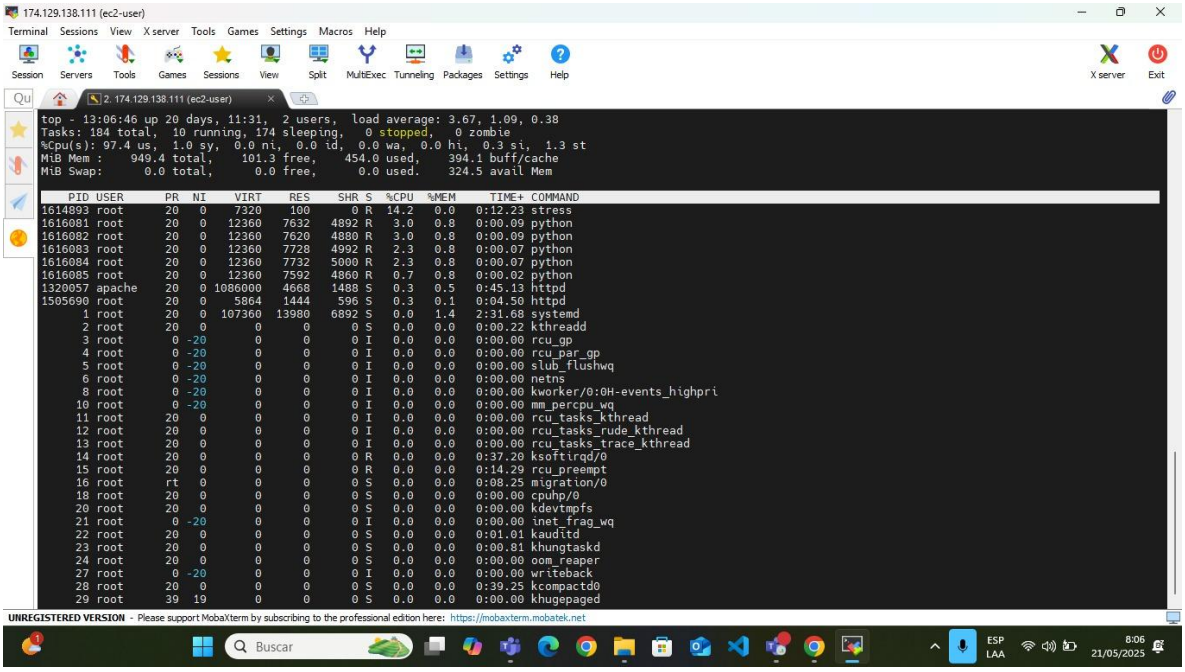


Nota. Fuente. Elaboración propia 2025.

En la siguiente imagen observamos el monitoreo de CPU sin errores demostrando su buen uso, observamos su uso es del 97.4% lo que indica que es una carga generada, como scripts o herramienta de prueba.

Figura 10

monitoreo de CPU



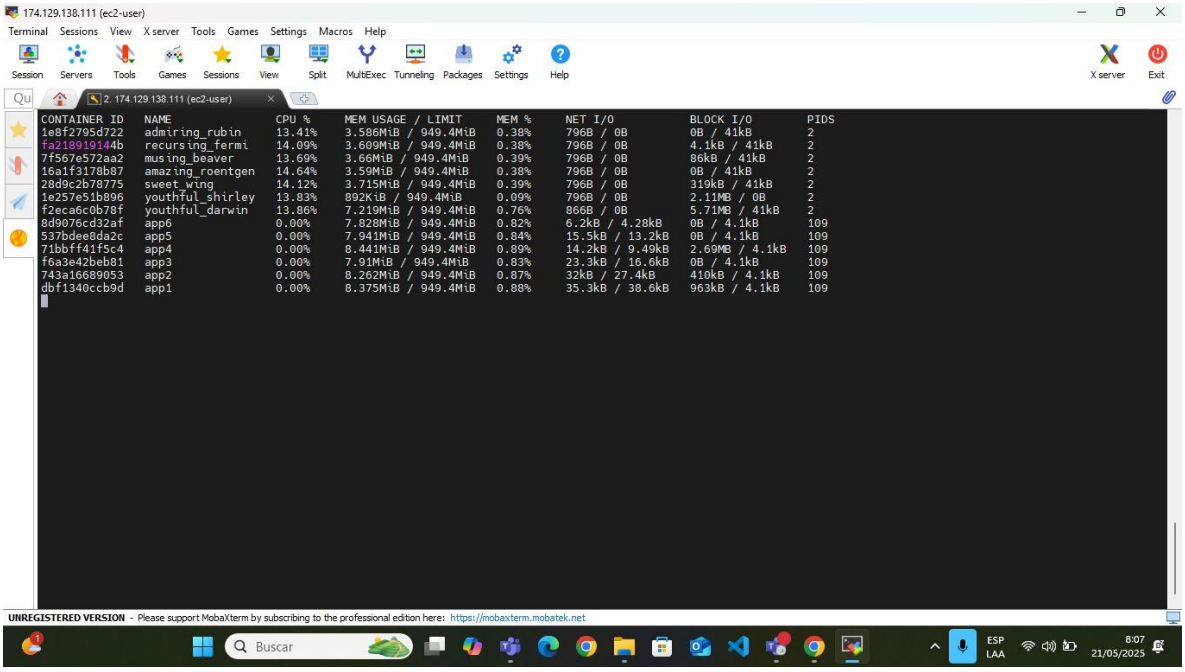
Nota. Fuente. Elaboración propia 2025.

En la siguiente imagen observamos el uso de 6 contenedores, con un uso superior al 13% cada uno, lo cual suma un consumo significativo, en un recurso limitado.

Al realizar la acción se evidencia que está simulando la carga de procesamiento, como parte de la prueba de estrés del rendimiento de la instancia.

Figura 11

uso de 6 contenedores



Nota. Fuente. Elaboración propia 2025.

3.5 Ejecución con sitios web con contenido estático:

Tabla 1

Se pueden ejecutar 3 formas las cuales son

Tipo de carga	Numero de contenedores
Muy ligero (nginx + HTML/CSS)	15—25 contenedores
Ligero con logging	10—15 contenedores
Ligero con TLS(HTTPS)	5—10 contenedores

Nota. Fuente. Elaboración propia 2025.

El límite dependerá de la configuración, tráfico y uso de CPU por conexiones.

Tabla 2

Ventajas tiene esto frente a la virtualización tradicional:

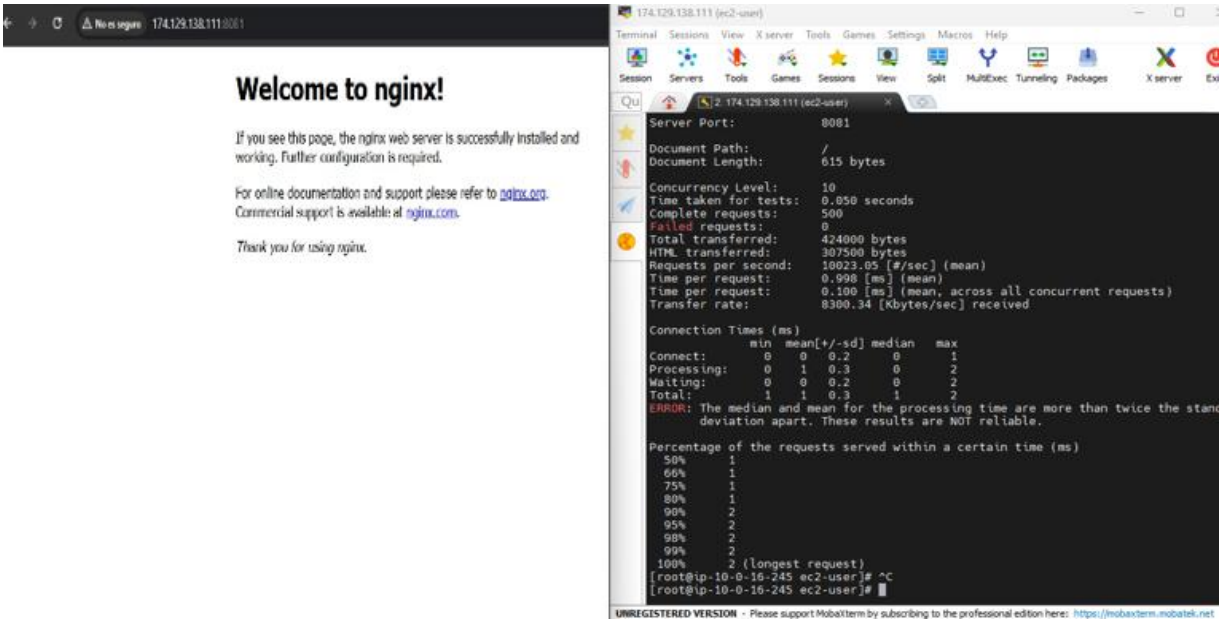
Docker (Contenedores)	**Virtualizacion tradidional (VMS)**
Ligero (Usa l mismo kernel de host)	Pesado (cada VM tiene su propio SO)
Inicia en milisegundos	Inicia en segundos/minutos
Alta densidad de instancias	Menos densidad (más RAM/CPU por VM)
Mejor rendimiento en l/O y CPU	Overhead por virtualización
Facil empaquetado y despliegue	Mas complejo de administrar
Ideal para microservicios y CI/CD	Mejor para sistemas monolíticos o SO completas pruebas

Nota. Fuente. Elaboración propia 2025.

Para ingresar a la información contenida o cargada en cada uno de los contenedores se hace a través de la ip publica de la instancia y el puerto del contendor

Figura 12

ingresar a la información contenida



Nota. Fuente. Elaboración propia 2025.

4. Conclusión

La realización de una red en Amazon Web Services (AWS) mediante instancias EC2, con sistemas operativos diferentes (Windows y Linux), mostró que las soluciones en la nube evidencian versatilidad, escalabilidad y robustez en entornos mixtos; una vez realizada una correcta configuración de la Virtual Private Cloud (VPC), de las subredes públicas, de los grupos de seguridad, de los sistemas de soporte, por ejemplo, el Internet Gateway, se obtiene conectividad entre las instancias y se disponen de los servicios web accesibles desde Internet.

Con las distintas partes del proyecto se logra conseguir el cumplimiento de unos objetivos propuestos; de esta forma se demuestra que el planificar en condiciones permite realizar el despliegue de arquitecturas funcionales y seguras, además de accesibles, aplicando buenas prácticas en infraestructura como servicio (IaaS). El uso de Docker, también permite realizar una simulación del entorno en el que se pueden utilizar múltiples contenedores, aprovechando mejor los recursos que en esquemas de virtualización convencional.

Los aprendizajes más significativos que emergen de las experiencias de la formación en el sistema de aprendizaje por competencias son los siguientes:

El fortalecimiento de las competencias técnicas en AWS, de forma que se amplían considerablemente las oportunidades laborales en el ámbito de la gestión de la administración de arquitecturas en la nube.

La importancia de la documentación y los ejercicios prácticos como elementos fundamentales para fijar la experiencia teórica y operativa de las tecnologías cloud.

La comprensión del modelo de computación en la nube como modelo de transformación digital, lo que permite diseñar soluciones eficientes, fiables y adaptables a las exigencias del mercado actual.

En síntesis, esta experiencia no solo constituyó una base sólida para comprender e implementar servicios en la nube, sino que también fijó habilidades clave para el desarrollo de soluciones tecnológicas innovadoras en el ámbito empresarial y académico.

Referencias

Amazon Web Services. (2023). *Amazon EC2 documentation*. Recuperado de https://docs.aws.amazon.com/ec2/

Hwang, K., Dongarra, J., & Fox, G. (2013). *Distributed and Cloud Computing: From Parallel Processing to the Internet of Things*. Elsevier.

Merkel, D. (2014). Docker: lightweight Linux containers for consistent development and deployment. *Linux Journal*, 2014(239), 2.

Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues. *Future Generation Computer Systems*, 28(3), 583–592. https://doi.org/10.1016/j.future.2010.12.006