



TRABAJO DE GRADO
Opción Seminario-Diplomado.

Implementación de Arquitectura en AWS con Balanceador de Carga y Contenedores

Corporación Universitaria Remington.
Nombre de la facultad: Ingeniería
Nombre del programa académico: Ingeniería de sistemas – Especialización en seguridad de la información

Nombres de los estudiantes autores del trabajo de grado.
Edwin Orlando Garzón Orjuela
Juan Aníbal Jiménez Cogollo
Luis Fernando Fabra Marquez
Nombre del Tutor del trabajo de grado: Juan Pablo Berrío López.
Opción de Trabajo de grado Seminario-Diplomado.
2025.

Agradecimientos

Quiero agradecer a mis compañeros por el esfuerzo y empeño en la realización de este seminario, también agradecer al profesor Juan Pablo Berrio por su paciencia en cada una de las clases, por aclarar cada una de las dudas que teníamos durante el desarrollo del seminario.

Tabla de Contenidos

Resumen.....	4
Marco conceptual y contextual	4
Desarrollo e implementación del aprendizaje.....	4
Conclusiones	33
Referencias.....	34

Resumen

Se diseñó una arquitectura para organizar los diferentes componentes de la aplicación AWS. Esto incluía decidir qué servicio utilizar (EC2, S3, RDS, etc.).

Se configuró un entorno de desarrollo para poder probar la aplicación antes de desplegarla en la nube.

Luego, procedí con crear una instancia, se configuró el entorno de ejecución, se implementó la aplicación Copiando los archivos de mi aplicación a la instancia y se configuró para que se ejecutara, se configuró el balanceador de carga que distribuye el tráfico entre múltiples instancias, se Configuró el almacenamiento

Lo que se aprendió:

- AWS: Ofrece una amplia gama de servicios para que se pueda construir aplicaciones escalables y confiables.
- La configuración: Configurar es clave para que todos los servicios de AWS tengan un correcto funcionamiento de la aplicación.
- La seguridad: Proteger la aplicación de ataques es una prioridad.

Palabras clave

Amazon CloudWatch

EC2 (Amazon Elastic Compute Cloud)

ELB (Elastic Load Balancer)

VPC (Virtual Private Cloud)

Security Groups (Grupos de Seguridad)

AutoScaling (Autoescalado)

AMI (Amazon Machine Image)

S3 (Simple Storage Service)

Marco conceptual y contextual

Marco conceptual:

AWS proporciona una amplia gama de servicios destinados a ejecutar instancias de forma escalable y eficaz.

los principales recursos que se pueden utilizar:

Amazon EC2, o Amazon Elastic Compute Cloud:

Pilar fundamental: Es el servicio de computación en la nube de AWS más utilizado. En sólo unos minutos, permite aprovisionar instancias virtuales, o servidores.

Numerosos tipos de instancias: instancias optimizadas para cargas de trabajo concretas (computación, memoria, almacenamiento, gráficos, etc.) e instancias de propósito general.

Escalabilidad: Puede ajustar la capacidad de sus instancias a sus necesidades, lo que le permite adaptarse a los cambios en la demanda.

Amazon Machine Images (AMIs)

Plantillas que vienen preconfiguradas e incluyen el sistema operativo, las aplicaciones y otra información necesaria para lanzar una instancia.

Galería masiva: Además de permitirle crear sus propias AMI exclusivas, AWS proporciona una amplia galería de AMI públicas.

Amazon Virtual Private Cloud (Amazon VPC)

La red de nube privada virtual de AWS le permite configurar una red segura y aislada en la que ejecutar sus instancias.

Control total sobre la configuración de la red: Se pueden definir subredes, tablas de enrutamiento, gateways de Internet, etc.

Amazon Elastic Block Store (Amazon EBS)

Almacenamiento persistente de bloque: Proporciona volúmenes de almacenamiento persistente que puede adjuntar a sus instancias.

Varios tipos de volúmenes: proporciona volúmenes de uso general, asequibles y de alto rendimiento para satisfacer sus demandas.

Amazon Elastic Load Balancing (Amazon ELB)

Al distribuir el tráfico entrante entre varias instancias, el equilibrio de carga mejora el rendimiento y la disponibilidad de sus aplicaciones.

Servicios adicionales pertinentes:

Amazon Auto Scaling: Esta característica modifica automáticamente la capacidad de sus instancias en respuesta a la demanda.

Amazon CloudWatch: Realiza un seguimiento del rendimiento de tus instancias y te permite establecer alertas.

AWS Systems Manager: Puede administrar sus instancias a escala.

Marco Contextual

En el entorno actual, en el que las empresas buscan soluciones tecnológicas flexibles para satisfacer la creciente demanda de servicios digitales, el despliegue de arquitecturas en la nube escalables y seguras es crucial. Para enmarcar este proyecto de grado se utilizan los siguientes contextos:

1. Relevancia académica: Este estudio se realiza dentro del programa de Ingeniería de Sistemas con enfoque en seguridad de la información, lo cual está en línea con los objetivos de desarrollar profesionales competentes en soluciones en la nube y tecnologías emergentes.
2. Sector tecnológico: AWS domina el mercado de servicios en la nube y es una plataforma muy utilizada en comercio electrónico, fintech y educación. Los estudiantes que aprenden estas herramientas a través de la experiencia práctica están mejor equipados para hacer frente a los obstáculos en el lugar de trabajo.
3. Beneficio práctico: Al simular circunstancias de implementación reales, el uso de una arquitectura con balanceadores de carga y contenedores proporciona a los estudiantes experiencia en la creación de soluciones escalables y fiables.
4. Seguridad: Como componentes cruciales de la gestión de TI contemporánea, el proyecto hace hincapié en la importancia de la seguridad de los datos y la configuración segura del entorno de nube.

Entrega 1

Implemente un servidor web en Amazon Linux, y explique toda la configuración necesaria que hizo en aws y dentro de la instancia.

Previamente nos damos de alta en Amazon Web services <https://aws.amazon.com/>

Luego de darnos de alta procedemos a crear todo lo necesario para implementar nuestro servidor web, además de un video explicando todo lo realizado:

https://drive.google.com/file/d/1rVTTVpeztj2ncX1MeH60aIrnUzh-1r0O/view?usp=drive_link

Figura 1.1: Creamos una nube virtual, en este caso llamada “Remington-vpc”

The screenshot displays the AWS Management Console interface for configuring a Virtual Private Cloud (VPC). The main panel shows the 'Sus VPC (1/2) Información' section, which includes a table of VPCs and a detailed configuration view for the selected 'Remington-vpc'.

Name	ID de la VPC	Estado	Bloquear el...	CIDR IPv4	CIDR IPv6
Remington-vpc	vpc-0874eb3a8da8e8292	Available	Desactivado	10.0.0.0/16	-
-	vpc-0a8d93abf57723a6	Available	Desactivado	172.31.0.0/16	-

The configuration view for 'Remington-vpc' shows the following settings:

- Resolución de DNS: Habilitado
- Tenencia: Default
- Conjunto de opciones de DHCP: dopt-04109b251f2b50327
- Tabla de enrutamiento principal: rtb-09e8a69c0962ff416
- ACL de red principal: acl-080e8f9733ce03ece
- VPC predeterminada: No
- CIDR IPv4: 10.0.0.0/16
- Grupo IPv6: -
- CIDR IPv6 (grupo de bordes de red): -
- Métricas de uso de direcciones de red: Desactivado
- Grupos de reglas del firewall de DNS de Route 53 Resolver: -
- ID de propietario: 156041402286

Figura 1.2: Información de nuestro VPC

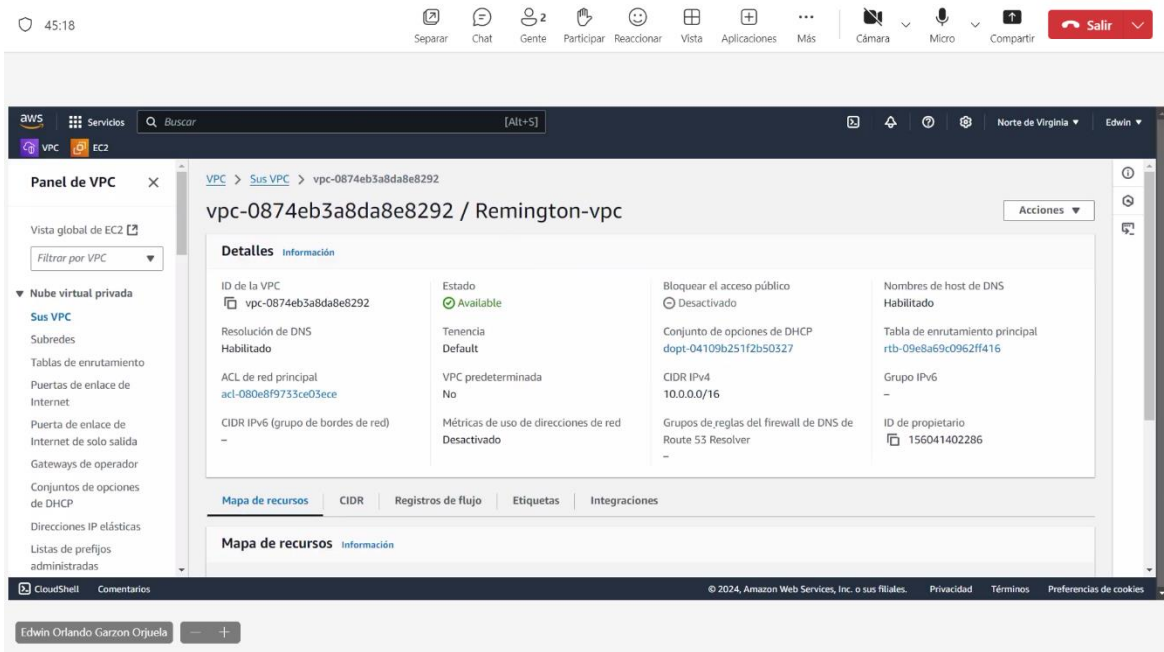


Figura 1.3: Información de nuestro mapa de recursos

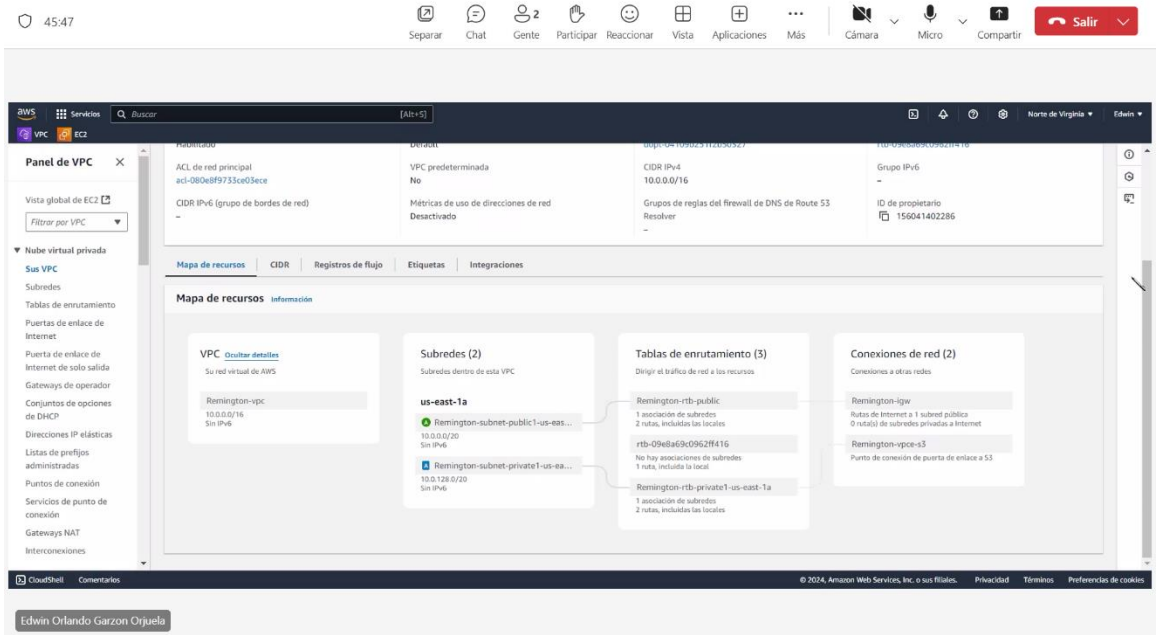


Figura 1.4: Se crea una instancia Amazon-linux llamada “ServerU1”

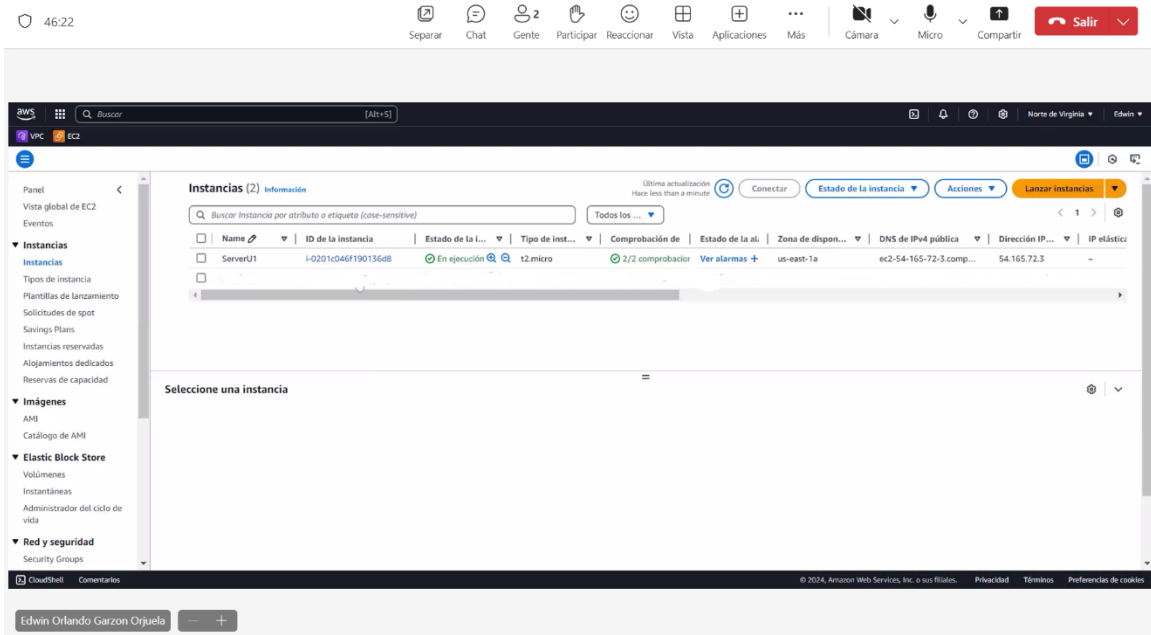


Figura 1.5: Aquí vemos un resumen de nuestra instancia (IPv4 pública y privada, tipo de instancia, entre otra información)

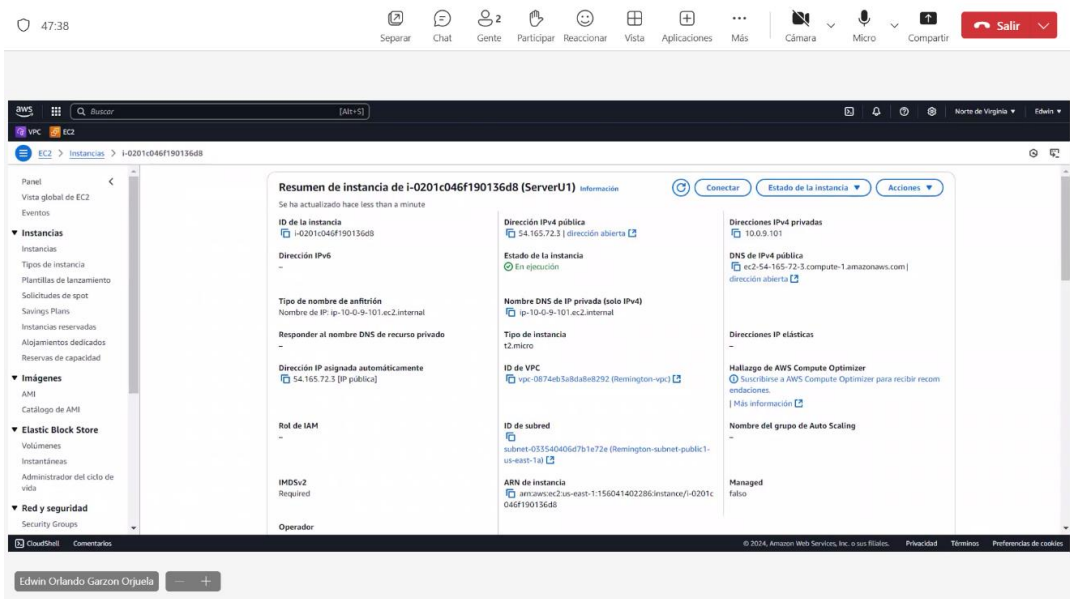


Figura 1.6: Aquí se logra ver los Security Groups y las reglas de entrada y salida

Panel
Vista global de EC2
Eventos

▼ Instancias
Instancias
Tipos de instancia
Plantillas de lanzamiento
Solicitudes de spot
Savings Plans
Instancias reservadas
Alojamientos dedicados
Reservas de capacidad

▼ Imágenes
AMI
Catálogo de AMI

▼ Elastic Block Store
Volúmenes
Instantáneas
Administrador del ciclo de vida

▼ Red y seguridad
Security Groups

CloudShell Comentarios

© 2024, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

Edwin Orlando Garzon Orjuela

Figura 1.7: Una vista más detallada de las reglas de entrada, utilizaremos los protocolos SSH con su puerto 22 y HTTP con su respectivo puerto 80 y dejaremos el segmento 0.0.0.0/0 para que se pueda acceder desde cualquier lugar.

48:40

Separar Chat Gente Participar Reaccionar Vista Aplicaciones Más Cámara Micro Compartir Salir

SWPS [Alt+S]

vpc ec2

EC2 > Grupos de seguridad > sg-00e914e52205e2e47 - SG_ServerU1 > Editar reglas de entrada

Editar reglas de entrada Información

Las reglas de entrada controlan el tráfico entrante que puede llegar a la instancia.

Reglas de entrada Información

ID de la regla del grupo de seguridad	Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional
sg-09189eade658fe4ae	HTTP	TCP	80	Pe... 0.0.0.0/0	
sg-06d510c4674cb163d	SSH	TCP	22	Pe... 0.0.0.0/0	

Agregar regla

Las reglas cuyo origen es 0.0.0.0/0 o :::/0 permiten a todas las direcciones IP acceder a la instancia. Recomendamos configurar reglas de grupo de seguridad para permitir el acceso únicamente desde direcciones IP conocidas.

Cancelar Previsualizar los cambios Guardar reglas

CloudShell Comentarios

© 2024, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

Edwin Orlando Garzon Orjuela

Figura 1.8: Para conectarnos a la instancia creada, seleccionamos la instancia y damos clic en la opción “Conectar”.

The screenshot shows the AWS Management Console interface. At the top, there is a navigation bar with various icons and a 'Salir' button. Below this, the console header shows 'AWS' and a search bar. The main content area is titled 'Instancias (1/2) Información'. A table lists instances, with one instance 'ServerU1' (ID: i-0201c046f190136d8) in the 'En ejecución' state. The 'Conectar' button is highlighted. Below the table, the details for instance 'i-0201c046f190136d8 (ServerU1)' are shown, including security groups and launch time.

Figura 1.9: Escogemos el tipo de conexión que va a ser por SSH, previamente al momento de crear la instancia tenemos descargado nuestra llave en extensión “.ppk”

The screenshot shows the 'Conectarse a la instancia' page in the AWS Management Console. The 'Cliente SSH' tab is selected. The page provides instructions for connecting to the instance 'i-0201c046f190136d8 (ServerU1)'. The instructions include:

1. Abra un cliente SSH.
2. Localice el archivo de clave privada. La clave utilizada para lanzar esta instancia es ServerU1.pem
3. Ejecute este comando, si es necesario, para garantizar que la clave no se pueda ver públicamente.


```
chmod 400 "ServerU1.pem"
```
4. Conéctese a la instancia mediante su DNS público:


```
ec2-54-165-72-3.compute-1.amazonaws.com
```

 An example command is provided:


```
ssh -i "ServerU1.pem" ec2-user@ec2-54-165-72-3.compute-1.amazonaws.com
```

 A note at the bottom states: 'Nota: En la mayoría de los casos, el nombre de usuario adivinado es correcto. Sin embargo, lea las instrucciones de uso de la AMI para comprobar si el propietario de la AMI ha cambiado el nombre de usuario predeterminado de la AMI.'

Figura 1.10: Por medio del software Putty nos conectamos a nuestra instancia...

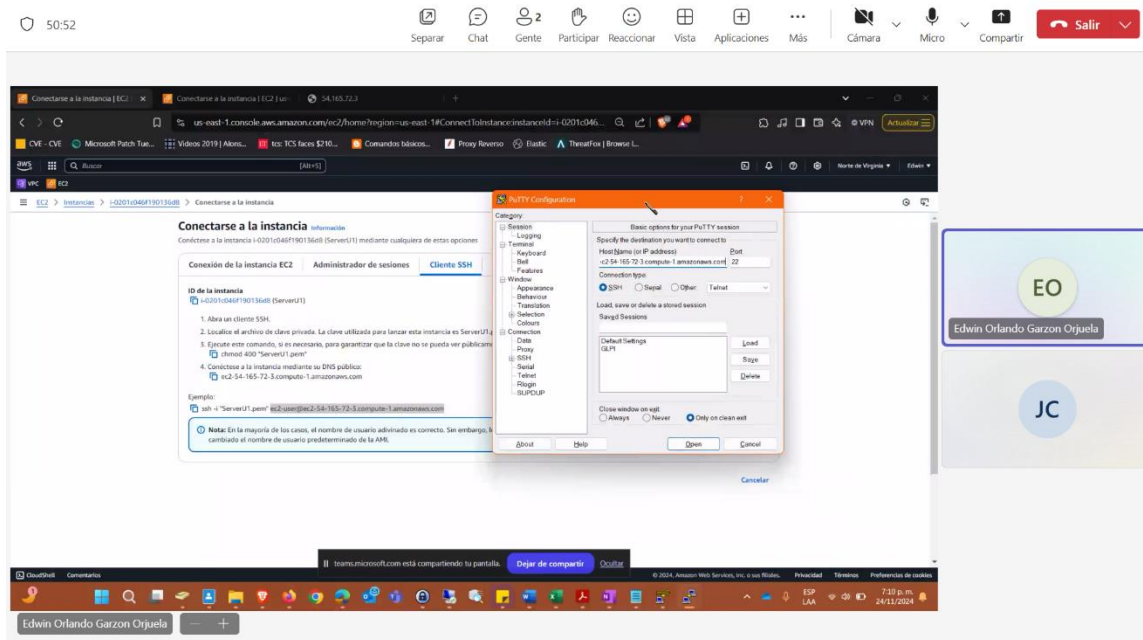


Figura 1.11: Nos vamos a la pestaña Connection/SSH/Auth/Credentials

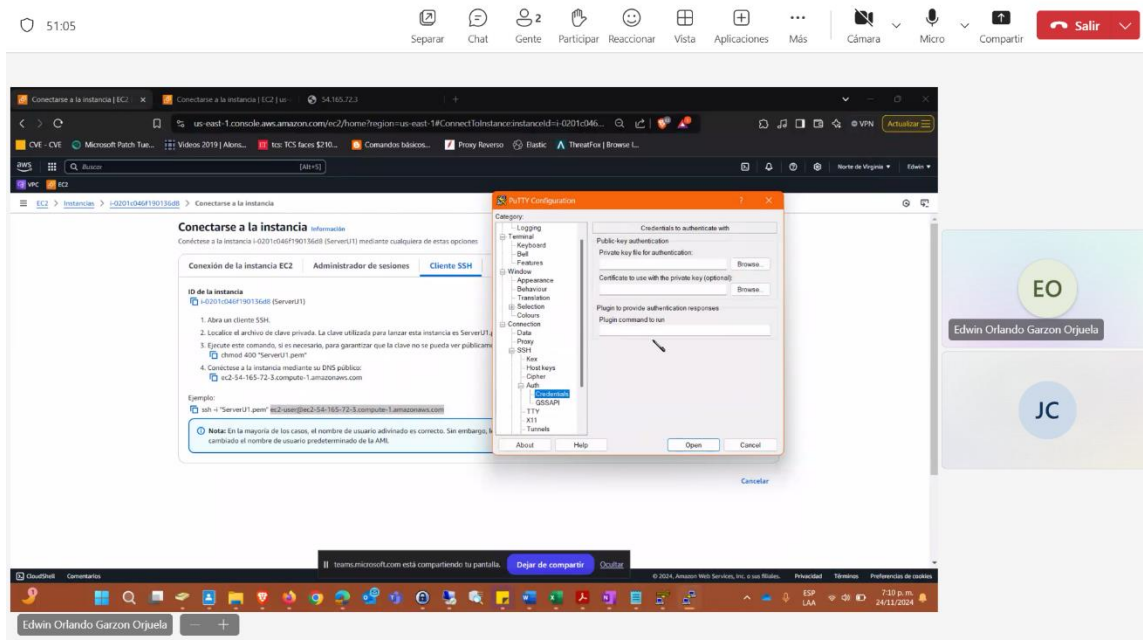


Figura 1.12: Escogemos la llave “ppk” que descargamos al momento de crear nuestra instancia.

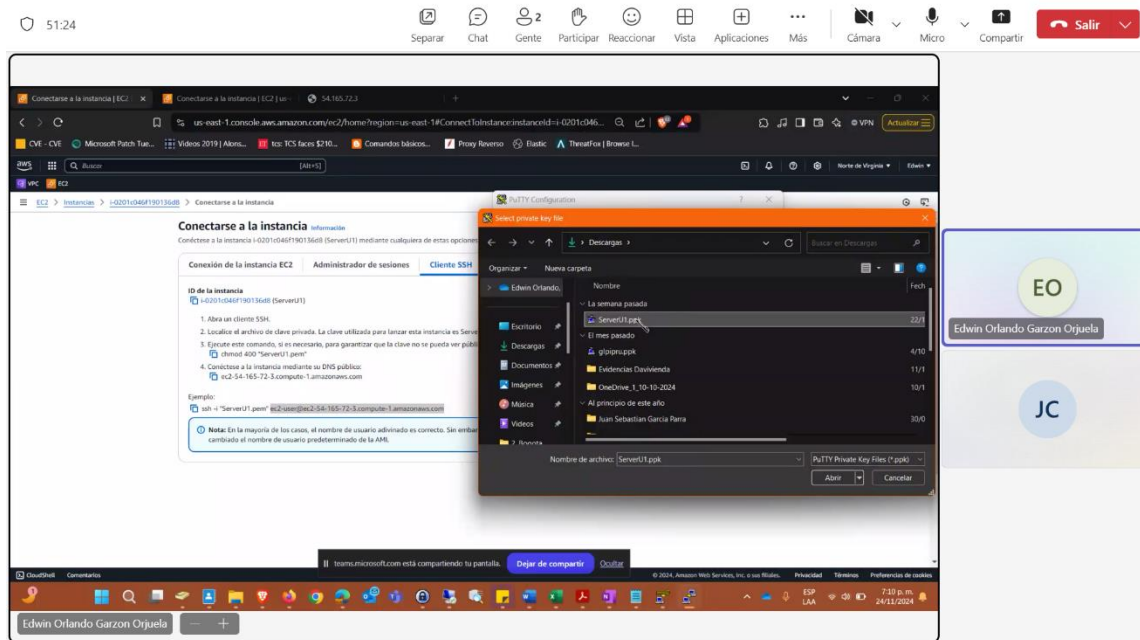


Figura 1.13: Logramos ingresar a nuestra instancia.

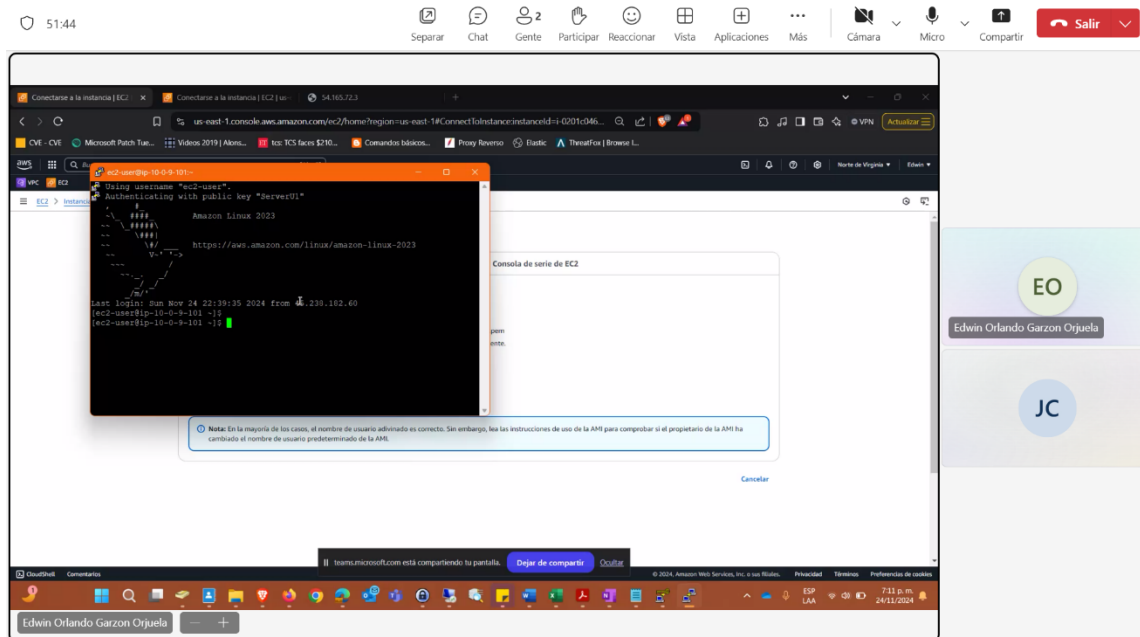


Figura 1.14: Le instalamos el servidor de protocolo de transferencia de hipertexto de apache por medio del comando “yum install httpd”

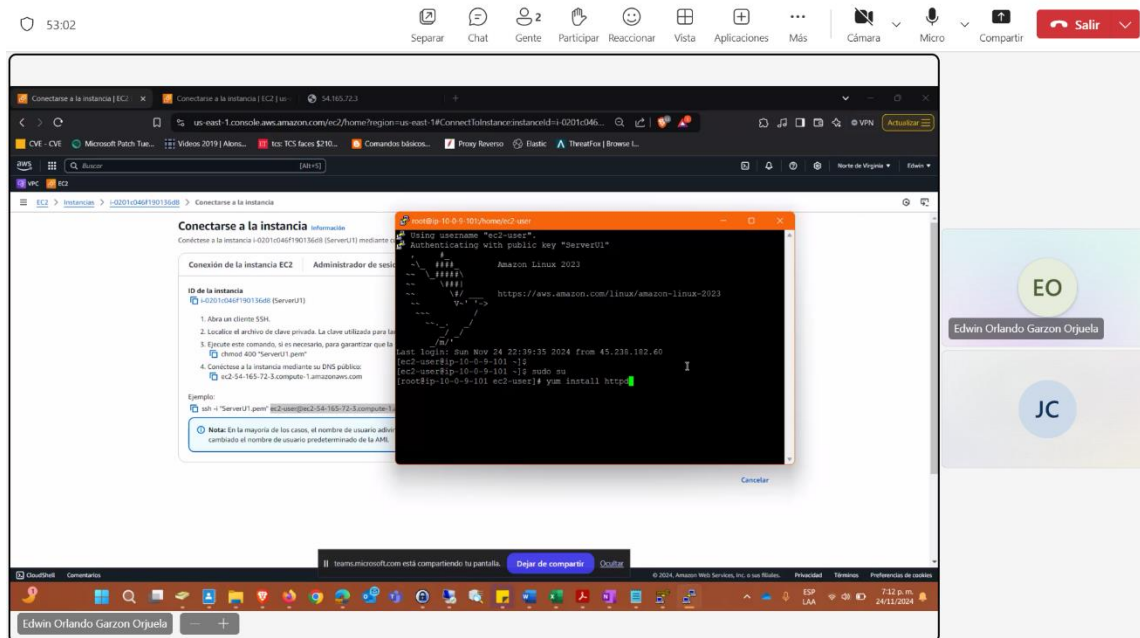


Figura 1.15: utilizamos el comando “systemctl status httpd” Para verificar estado de la instalación httpd

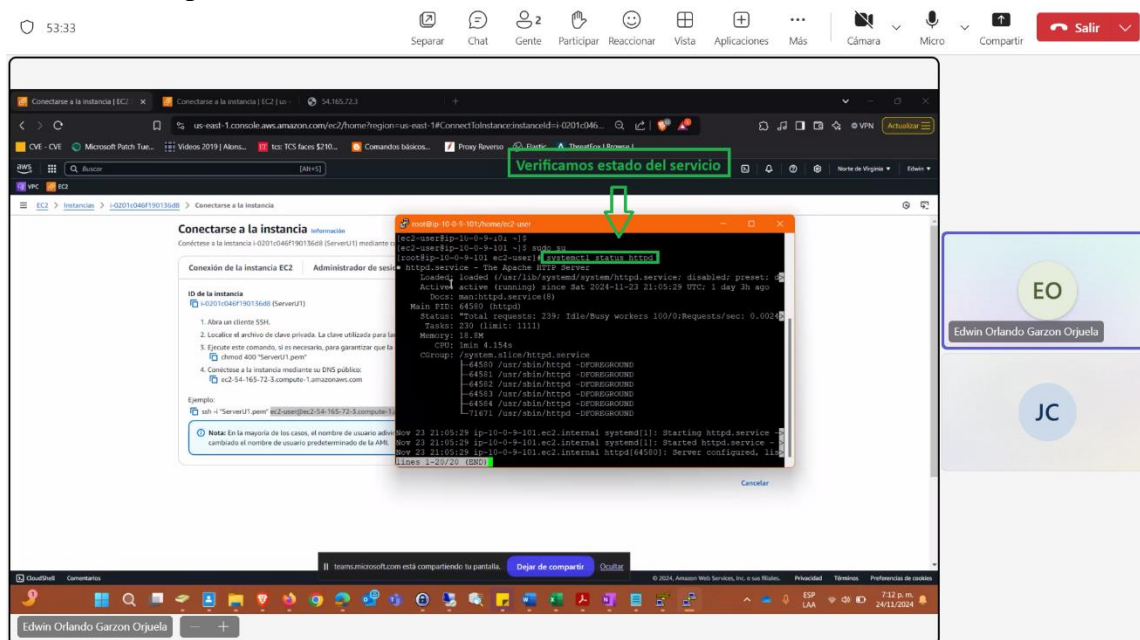
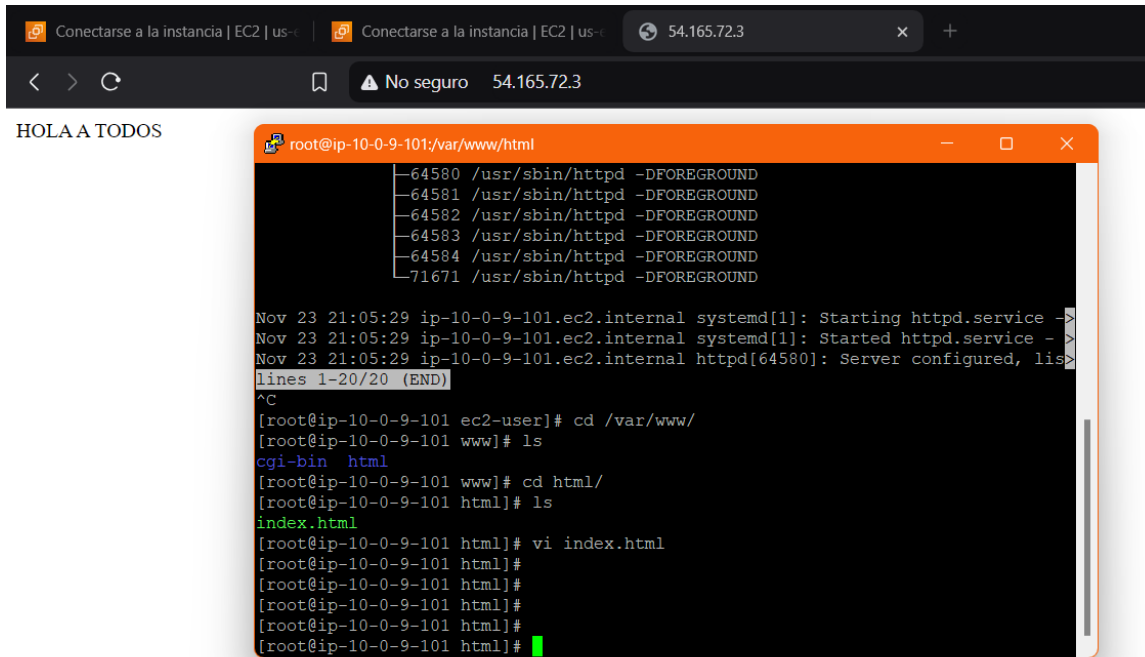


Figura 1.16: Le agregamos contenido al directorio web ubicado en este caso en `/var/www/html/` y un clásico “HOLA A TODOS”



```

root@ip-10-0-9-101:/var/www/html# vi index.html
^C
[root@ip-10-0-9-101 ec2-user]# cd /var/www/
[root@ip-10-0-9-101 www]# ls
cgi-bin  html
[root@ip-10-0-9-101 www]# cd html/
[root@ip-10-0-9-101 html]# ls
index.html
[root@ip-10-0-9-101 html]# vi index.html
[root@ip-10-0-9-101 html]#

```

Figura 1.17: con el comando “vi” podemos ver el contenido del archivo index.html” en este caso una línea

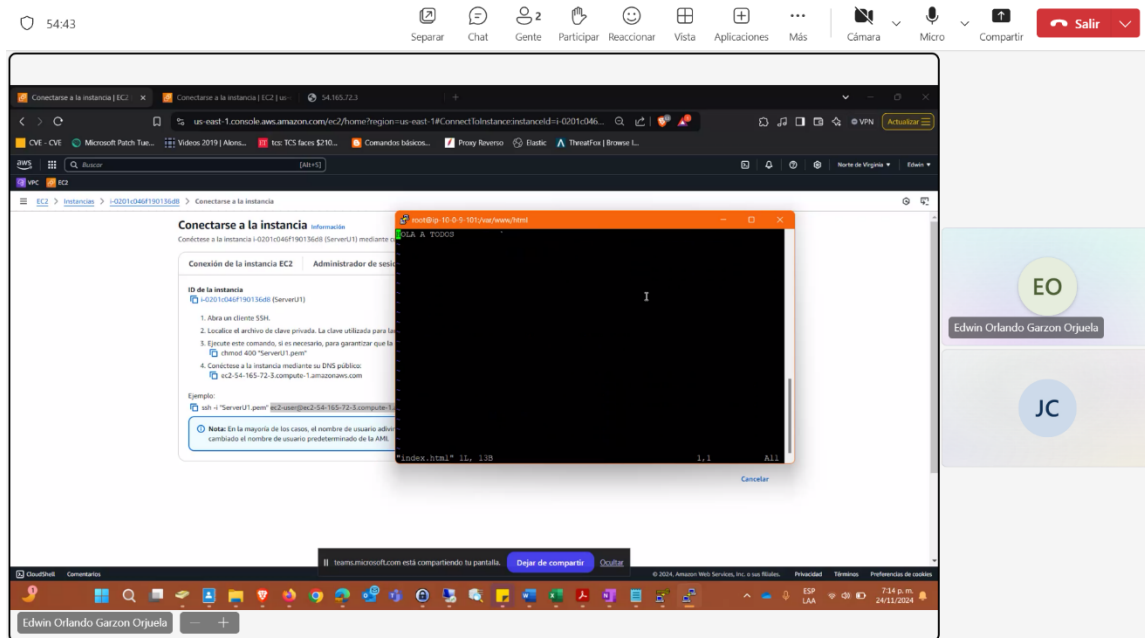


Figura 1.18: Demostración de nuestra página desde la ip: 54.165.72.3

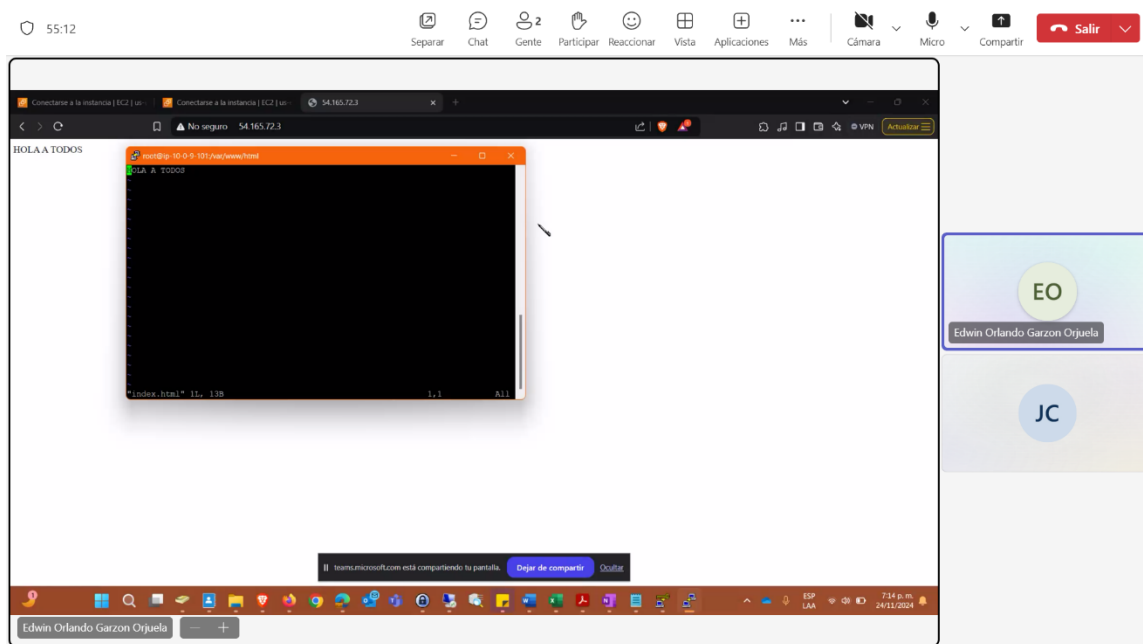
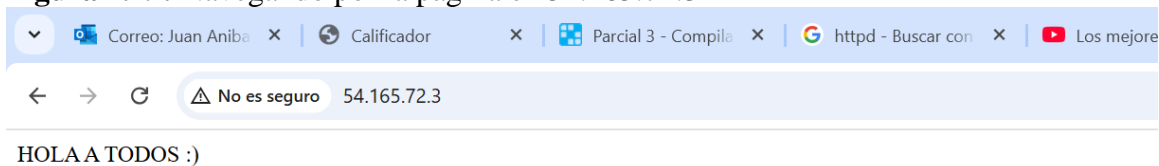


Figura 1.19: Navegando por la página en 54.165.72.3



Entrega 2

Implemente un servidor web en Amazon Linux, que ejecute una aplicación en una instancia, haga pruebas mostrando como las instancias se crean automáticamente y la aplicación queda funcionando correctamente.

Para hacer que la instancia suba el uso de CPU se ejecutan las dos siguientes sentencias en el terminal:

```
“while ;; do ;; done &
for i in {1..4}; do while ;; do ;; done & done”
```

Se muestra video de lo realizado:

https://drive.google.com/file/d/1rVTTVpeztj2ncX1MeH60aIrnUzh-1r0O/view?usp=drive_link

Figura 2.1: Aquí nos muestra los balanceadores de carga, en nuestro caso un balanceador que se llama “ELB-PRD”

The screenshot displays the AWS Management Console interface for the 'Balanceadores de carga' (Load Balancers) section. A notification banner at the top indicates a change in the ARS zone for application load balancers. Below this, a table lists the load balancers, with 'ELB-PRD' selected. The details for 'ELB-PRD' are shown below the table, including its status (Active), VPC ID, zones, and creation date.

Nombre	Nombre de DNS	Estado	ID de VPC	Zonas de disponibilidad	Tipo	Fecha creada
ELB-PRD	ELB-PRD-1690893324.us-e...	Activo	vpc-0874eb3a8da8e8292	2 Zonas de disponibilidad	application	30 de noviembre de 2024, 17:09 (UTC-05:00)

Equilibrador de carga: ELB-PRD

Detalles

Tipo de equilibrador de carga Aplicación	Estado Activo	VPC vpc-0874eb3a8da8e8292	Tipo de dirección IP del equilibrador de carga IPv4
Esquema Internet-facing	Zona hospedada Z35SXDOTRQ7X7K	Zonas de disponibilidad subnet-0334a0406d7b1e72e us-east-1a (use1-a24) subnet-0204926124856c4bf us-east-1b (use1-a26)	Fecha creada 30 de noviembre de 2024, 17:09 (UTC-05:00)
ARN del equilibrador de carga arn:aws:elasticloadbalancing:us-east-1:156041402286:loadbalancer/app/ELB-PRD/383f1bc22f8994ff		Nombre de DNS ELB-PRD-1690893324.us-east-1.elb.amazonaws.com (Registro A)	

Figura 2.2: Observamos los detalles de nuestro target group, a que VPC está apuntando, versión y tipo de protocolo, balanceador, tipo de destino y tipo de direccionamiento IP.

The screenshot shows the 'TargetGroup-PRD' details page in the AWS Management Console. The 'Detalles' section includes:

- Identificación:** arn:aws:elasticloadbalancing:us-east-1:156041402286:targetgroup/TargetGroup-PRD/5144814717e1350be
- Tipo de destino:** Instancia
- Protocolo:** HTTP, **Puerto:** 80
- Versión del protocolo:** HTTP1
- VPC:** vpc-08746b7ab64b292
- Tipo de dirección IP:** IPv4
- Balanceador de carga:** ELB-PRD

Summary statistics:

- Destinos totales: 5
- En buen estado: 3
- En mal estado: 1
- Sin utilizar: 1
- Inicial: 0
- Vaciado: 0

Below the statistics is a table of registered targets:

ID de instancia	Nombre	Puerto	Zona	Estado	Detalles del estado	Sustituir	Detalle...
i-0201d46f190135608	ServerU1	80	us-east-1a (us-...)	Healthy	-	No override	No overri...
i-07d9f95f5ub7b7e7e	ServerU2	80	us-east-1a (us-...)	Unread	Target is in the stoppe...	No override	No overri...
i-0cc19591879b6c06d		80	us-east-1b (us-...)	Healthy	-	No override	No overri...
i-055a3ad3b06b2ac8e		80	us-east-1a (us-...)	Healthy	-	No override	No overri...

Figura 2.3: Aquí observamos más en detalle a nuestro balanceador y sus respectivas reglas

The screenshot shows the 'ELB-PRD' details page in the AWS Management Console. The 'Detalles' section includes:

- Tipo de equilibrador de carga:** Aplicación
- Estado:** Activo
- VPC:** vpc-08746b7ab64b292
- Esquema:** Internet-facing
- Zona hospedada:** us-east-1a (us-east-1)
- Zonas de disponibilidad:** us-east-1a (us-east-1), us-east-1b (us-east-1)
- Tipo de dirección IP del equilibrador de carga:** IPv4
- Fecha creada:** 30 de noviembre de 2024, 17:09 (UTC-05:00)
- ARN del equilibrador de carga:** arn:aws:elasticloadbalancing:us-east-1:156041402286:loadbalancer/app/ELB-PRD/5183f1bc22f8994ff
- Nombre de DNS:** ELB-PRD-1690893324.us-east-1.elb.amazonaws.com (Registro A)

Below the details is a table of listening rules:

Protocolo/Port	Acción predeterminada	Reglas	ARN	Política de seguridad	Certificado SSL/TLS predet
HTTP:80	Reenviar al grupo de destino	1 regla	ARN	No aplicable	No aplicable

Figura 2.4: Aquí observamos más en detalle a nuestro balanceador y sus respectivos grupos de seguridad

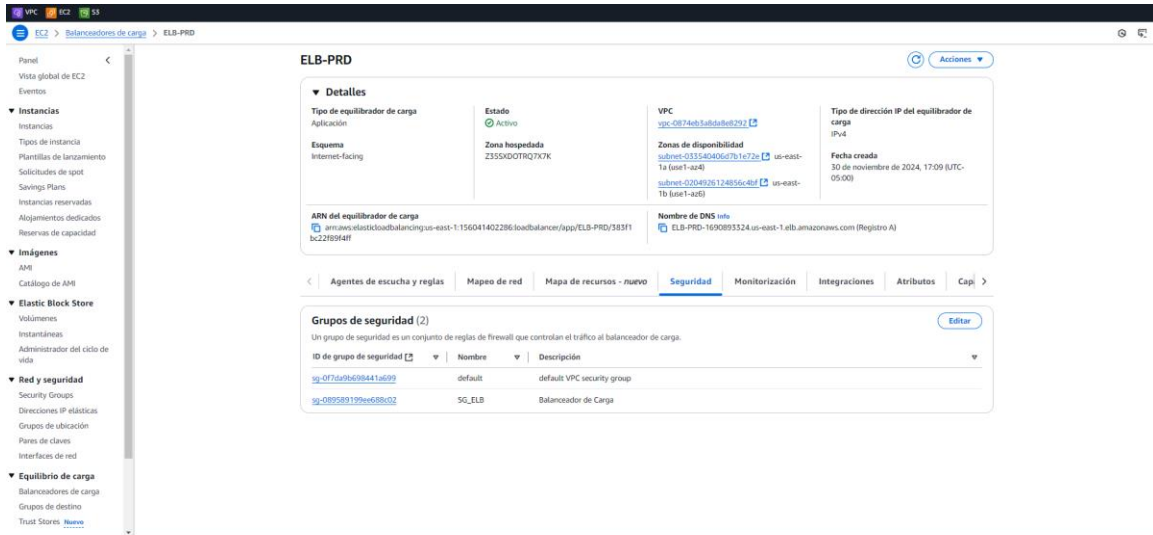


Figura 2.5: Aquí se observan los grupos de autoescalado

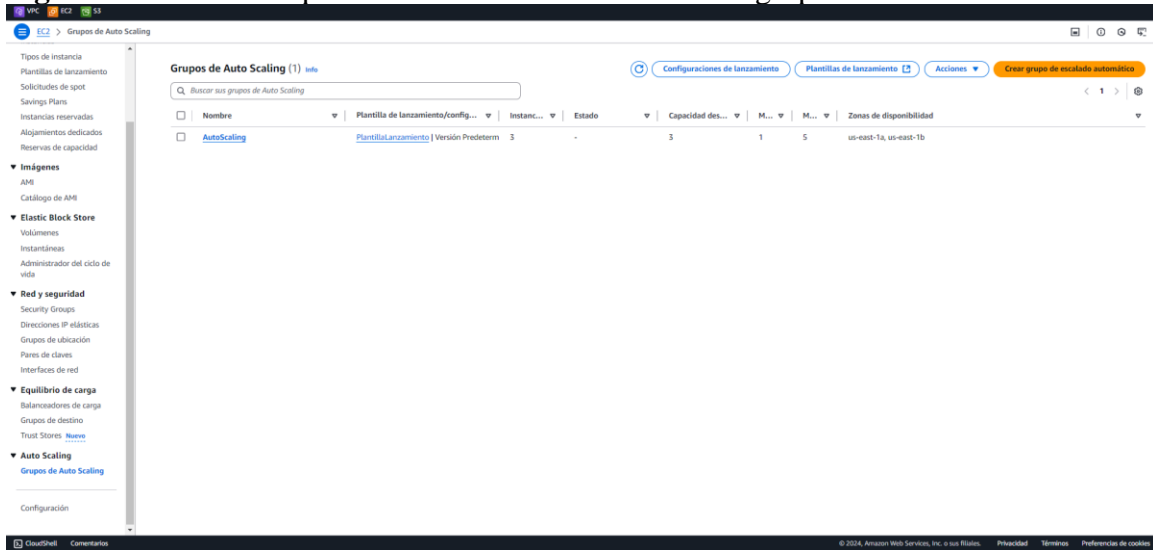


Figura 2.6: Aquí se detalla el tipo de capacidad, el tamaño de nuestro grupo y la capacidad mínima y máxima deseada

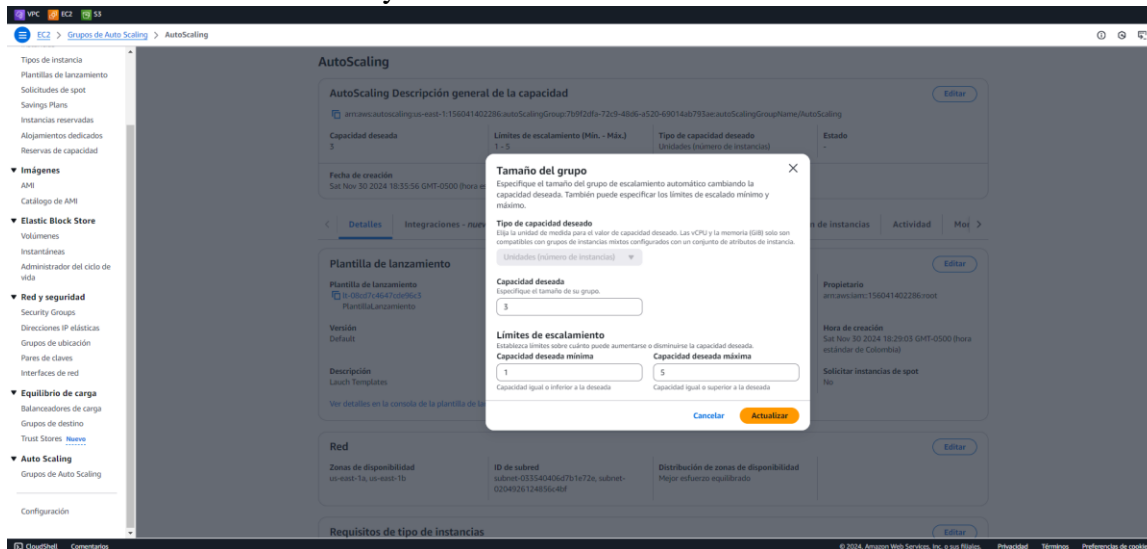


Figura 2.7: Continuamos viendo los detalles de nuestro grupo, como lo son los requisitos de tipos de instancias y zonas de disponibilidad y subredes.

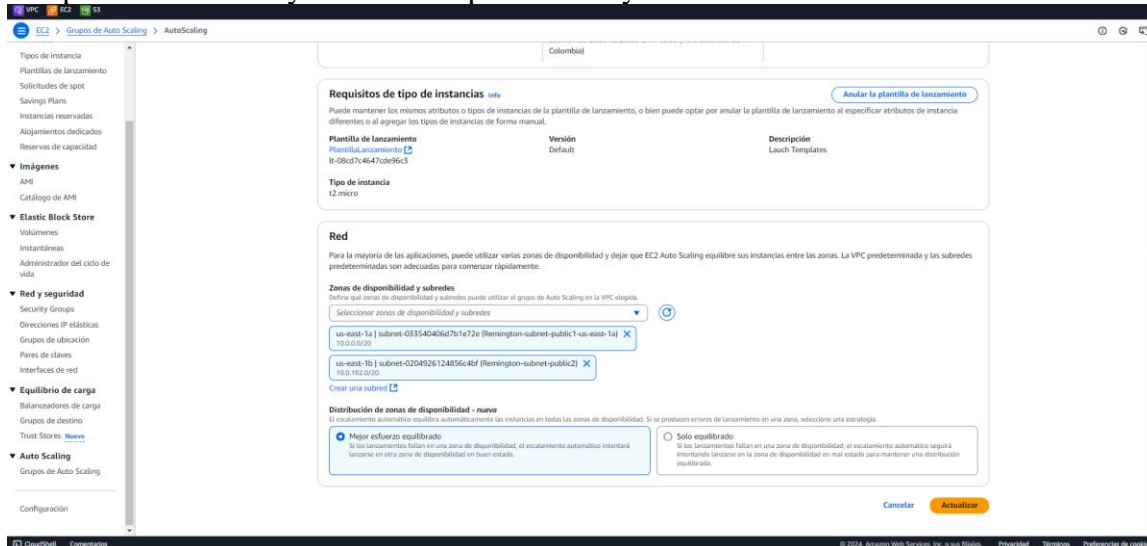


Figura 2.8: En esta vemos los detalles de la plantilla de lanzamiento con su respectiva versión

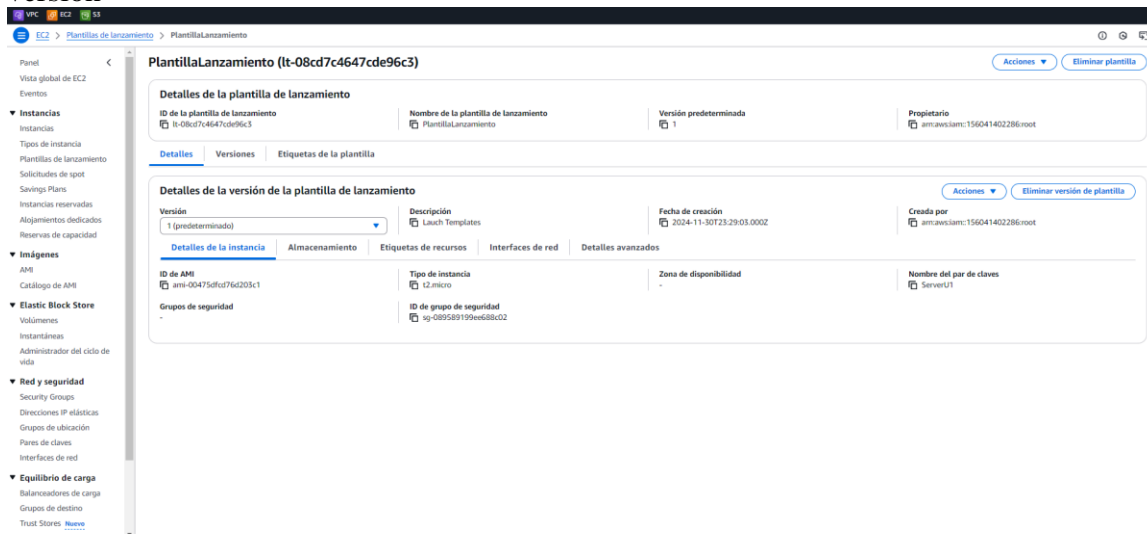


Figura 2.9: Una descripción general de la capacidad deseada y los límites de escalamiento.

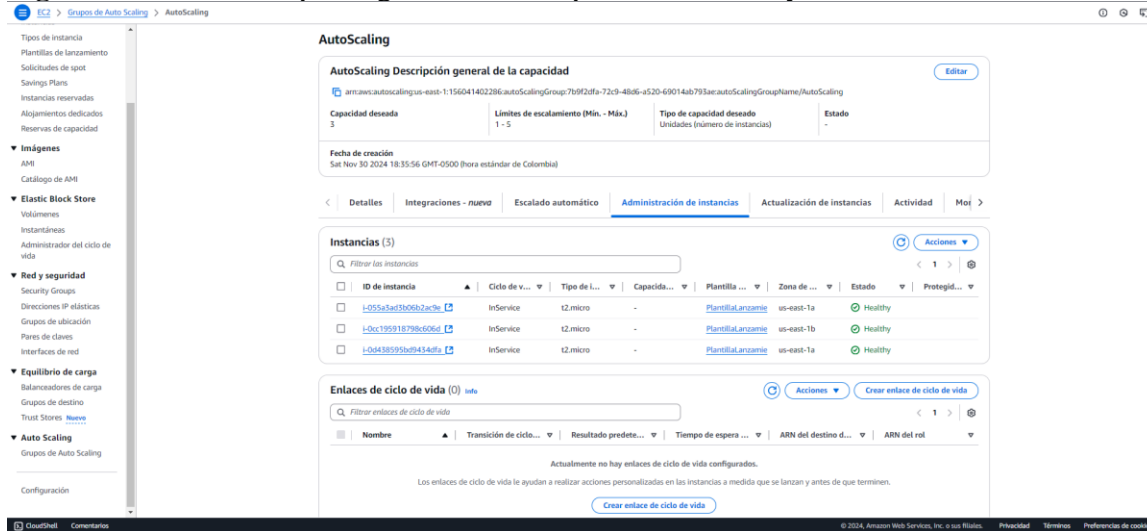


Figura 2.10: Las instancias que tenemos en nuestro balanceador

Nombre	ID de la instancia	Estado de la instancia	Tipo de inst...	Comprobación de estado	Estado de la apl...	Zona de dispon...	DNS de IPv4 pública	Dirección IP...	IP elasti
ServerU1	i-0201c046f190136d8	En ejecución	t2.micro	2/2 comprobaciones superadas	Ver alarmas +	us-east-1a	ec2-3-87-92-143.comp...	3.87.92.143	-
Windows	i-050424dfb4965a75a	Detenida	t2.micro	-	Ver alarmas +	us-east-1a	-	-	-
	i-04485959d94344fa	En ejecución	t2.micro	2/2 comprobaciones superadas	Ver alarmas +	us-east-1a	-	-	-
	i-055a5ad5b06b2ac9e	En ejecución	t2.micro	2/2 comprobaciones superadas	Ver alarmas +	us-east-1a	-	-	-
	i-0cc199f18796d06d	En ejecución	t2.micro	2/2 comprobaciones superadas	Ver alarmas +	us-east-1b	-	-	-
ServerU2	i-07d895f5sub7bd7fe	Detenida	t2.micro	-	Ver alarmas +	us-east-1a	-	-	-

Figura 2.11: Accediendo a una de nuestras instancias (“ServerU1”) a través de PUTTY

Conectarse a la instancia Información

Conéctese a la instancia i-0201c046f190136d8 (ServerU1) mediante cualquiera de estas opciones

Conexión de la instancia EC2 | Administrador de sesiones | **Cliente SSH** | Consola de serie de EC2

ID de la instancia: i-0201c046f190136d8 (ServerU1)

1. Abra un cliente SSH.
2. Localice el archivo de clave privada. La clave utilizada para lanzar esta instancia es ServerU1.pem
3. Ejecute este comando, si es necesario, para garantizar que la clave no se pueda ver públicamente. `chmod 400 'ServerU1.pem'`
4. Conéctese a la instancia mediante su DNS público: `ec2-3-87-92-143.compute-1.amazonaws.com`

Ejemplo: `ssh -i 'ServerU1.pem' ec2-user@ec2-3-87-92-143.compute-1.amazonaws.com`

Nota: En la mayoría de los casos, el nombre de usuario adivinado es correcto. Sin embargo, lea las instrucciones de uso de la AMI usuario predeterminado de la AMI.

PuTTY Configuration

Category: Session

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): `ec2-user@ec2-3-87-92-143.compute-1.a...`

Port: `22`

Connection type: SSH Serial Other: Telnet

Load, save or delete a stored session

Saved Sessions

Default Settings

GLPFI

Load Save Delete

Close window on exit: Always Never Only on clean exit

About Help Open Cancel

Figura 2.12: La SubRed que estamos utilizando es “Remington-subnet-public2”

Name	ID de subred	Estado	VPC	Bloquear el...	CIDR IPv4	CIDR IPv6	ID de asociación de CIDR
-	subnet-0d83b17d85cc0dad	Available	vpc-0a8d93abf977723a6	Desactivado	172.31.64.0/20	-	-
Remington-subnet-private1-us-east-1a	subnet-055140406c7b1c72a	Available	vpc-0874eb3ab0a8c2921 Remington-vpc	Desactivado	10.0.0.0/20	-	-
-	subnet-0c486c7a2b79940b	Available	vpc-0a8d93abf977723a6	Desactivado	172.31.80.0/20	-	-
-	subnet-02a4b10e6144d4	Available	vpc-0a8d93abf977723a6	Desactivado	172.31.32.0/20	-	-
-	subnet-02541ae11e45c0ee	Available	vpc-0a8d93abf977723a6	Desactivado	172.31.48.0/20	-	-
-	subnet-08f4663d099d1af	Available	vpc-0a8d93abf977723a6	Desactivado	172.31.0.0/20	-	-
Remington-subnet-public2	subnet-02d402612463c6df	Available	vpc-0874eb3ab0a8c2921 Remington-vpc	Desactivado	10.0.192.0/20	-	-
Remington-subnet-private1-us-east-1a	subnet-08f4663d099d1af	Available	vpc-0874eb3ab0a8c2921 Remington-vpc	Desactivado	10.0.128.0/20	-	-
-	subnet-0890c52ad8130200	Available	vpc-0a8d93abf977723a6	Desactivado	172.31.16.0/20	-	-

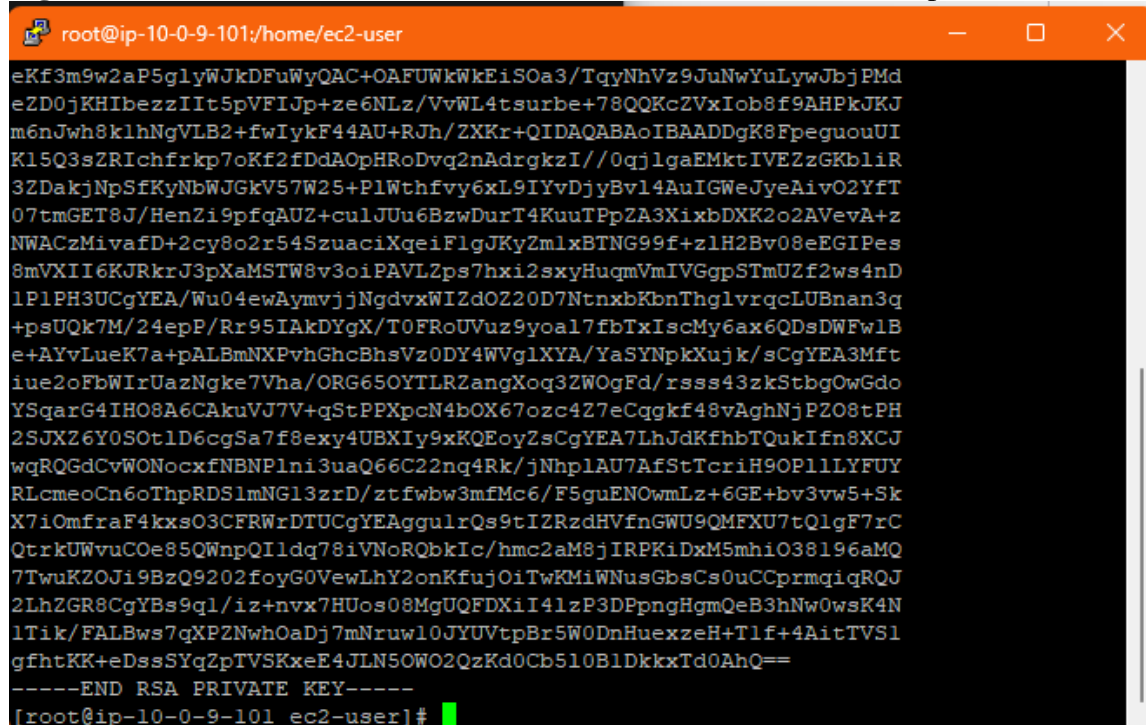
Figura 2.13: Parte de nuestra clave privada RSA

```

root@ip-10-0-9-101/home/ec2-user
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA2o55UfpuXVvBBSdvNwtHwUsCzRg4FAf7Egtg9m+W+GSrpYwi
nt/AJRaUyfdCALSyXL/nYA3uDP7BZfclWHetQWLG+FnipltzypAacVtVTWt0bW48
eFFoHtTff3QfBJHlg+MoV+PQ18ddiwUY2K4FrcAzc8VFtV/Tc3n9r+dVBSaOZCB1
eKf3m9w2aP5g1yWJkDFuWYQAC+OAFUWkKwEiSOa3/TqyNhVz9JuNwYuLywJbJpMd
eZD0jKHibezzIIt5pVFIJp+ze6NLz/VvWL4t surbe+78QQKc2VxIob8f9AHPkJKJ
m6nJwh8klhNgVLB2+fwIykF44AU+RjH/ZXKr+QIDAQABAoIBAADDgK8FpeguouUI
K15Q3sZRIchfrkp7oKf2fDdAOpHRoDvq2nAdrgkzI//0qjlgaEMktIVEZzGKbliR
3ZDakjNpSfKyNbWJGkV57W25+PlWthfvy6xL9IYvDjyBv14AuIGWeJyeAivO2YfT
07tmGET8J/HenZi9pfgAUZ+culJUu6BzwDurT4KuuTPpZA3XiXbDXK2o2AVevA+z
NWAczMivafd+2cy8o2r54SzuaciXqeiFlgJKyZmlxBTNG99f+z1H2Bv08eEGIPes
8mVXIIE6KJRkrJ3pXaMSTW8v3oiPAVLZps7hxi2sxyHuqmVmIVGgpSTmUZf2ws4nD
1P1PH3UCgYEA/Wu04ewAymvjNgdVxWIZdOZ20D7NtnxbKbnThglvrqcLUBnan3q
+psUqk7M/24epP/Rr95IAkDYgX/T0FRoUVuz9yoal7fbTxIscMy6ax6QDsDWFw1B
e+AyvLueK7a+pALBmNXpvhGhcBhsVz0DY4WVglXYA/YaSYNpkXujk/sCgYEA3Mft
iue2oFbWIRUazNgke7Vha/ORG65OYTLRZangXoq3ZWOgFd/rsss43zkStbgOwGdo
YSqarG4IH08A6CAkuVJ7V+qStPPXpcN4bOX67ozc427eCqgkf48vAghNjPZ08tPH
2SjXZ6Y0Sot1D6cgSa7f8exy4UBXIy9xKQEoyZsCgYEA7LhJdKfhtQukIfn8XCJ
wqRQgdcvWONocxfNBNPlni3uaQ66C22nq4Rk/jNhplAU7AfStTcriH9OP11LYFUY
RLcmeoCn6oThpRDSlmNG13zrD/ztfwbw3mFMc6/F5guENowmLz+6GE+bv3vw5+S
kX7iOmfraF4kx03CFRWRDTUCgYEA9gguLrQs9tI2RzdHVfnGWU9QMFXU7tQ1gF7rC
QtrkUvWuCOe85QWnpQIldq78iVnORQbkIc/hmc2aM8jIRPKiDxM5mhi038196aMQ
7TtuKZOJi9BzQ9202foyG0VewLhY2onKfujOitwKMiWNusGbsCs0uCCprmqiqRQJ
"cert.PEM" [readonly] 27L, 1679B
1,1 Top

```

Figura 2.14: Continuación de nuestra clave privada RSA.



```

root@ip-10-0-9-101:/home/ec2-user
eKf3m9w2aP5glyWJkDFuWyQAC+OAFUWkWkEiSOa3/TqyNhVz9JuNwYuLywJbjPMd
eZD0jKHibezzIIt5pVFIJp+ze6NLz/VvWL4tsurbe+78QQKcZVxIob8f9AHPkJKJ
m6nJwh8klhNgVLB2+fwIykF44AU+Rjh/ZXKr+QIDAQABAoIBAADDgK8FpeguouUI
K15Q3sZRIchfrkp7oKf2fDdAOpHRoDvq2nAdrgkzI//0qjlgEMktIVEZzGKbliR
3ZDakjNpSfKyNbWJGkV57W25+PlWthfvy6xL9IYvDjyBv14AuIGWeJyeAivO2YfT
07tmGET8J/Henzi9pfqAUZ+culJUu6BzwDurT4KuuTPpZA3XixbDXK2o2AVevA+z
NWACzMivafD+2cy8o2r54SzuaciXqeiFlgJKyZmlxBTNG99f+z1H2Bv08eEGIPes
8mVXIIE6KJRkrJ3pXaMSTW8v3oiPAVLZps7hxi2sxyHuqmVmIVGgpStmUZf2ws4nD
lPlPH3UCgYEA/Wu04ewAymvjNgdvxWIZdO220D7NtnxbKbnThglvrqcLUBnan3q
+psUQk7M/24epP/Rr95IAkDYgX/T0FRoUVuz9yoal7fbTxIscMy6ax6QDsDFw1B
e+AYvLueK7a+pALBmNXPvhGhcBhsVz0DY4WVglXYA/YaSYNpkXujk/sCgYEA3Mft
iue2oFbWIrUazNgke7Vha/ORG65OYTLRZangXoq3ZWogFd/rs43zkStbgOwGdo
YSqarG4IH08A6CAkuVJ7V+qStPPXpcN4bOX67ozc4Z7eCqgkf48vAghNjPZ08tPH
2SjXZ6Y0SotlD6cgSa7f8exy4UBXIy9xKQEoyZsCgYEA7LhJdKfhtQukIfn8XCJ
wQRQgdCvWONocxfNBNPlni3uaQ66C22nq4Rk/jNhplAU7AfStTcriH90P11LYFUY
RLcmeoCn6oThpRDSlmNG13zrD/ztfwbw3mFMc6/F5guEN0wmLz+6GE+bv3vw5+S
kX7iOmfraF4kxsO3CFRWrDTUCgYEAaggulrQs9tIZRzdHVfnGWU9QMFxu7tQlqF7rC
QtrkUWvuCOe85QWnpQIldq78iVNoRQbkIc/hmc2aM8jIRPKiDxM5mhi038196aMQ
7TwuKZOJi9BzQ9202foYG0VewLhY2onKfujOiTwKMiWNusGbsCs0uCCprmqiqRQJ
2LhZGR8CgYBs9q1/iz+nvx7HUos08MgUQFDXiI41zP3DPpngHgmQeB3hNw0wsK4N
lTik/FALBws7qXP2NwhOaDj7mNruw10JYUVtpBr5W0DnHuezzeH+T1f+4AitTVS1
gfhtKK+eDssSYqZpTVSKxeE4JLN5OW02QzKd0Cb510B1DkKxTd0AhQ==
-----END RSA PRIVATE KEY-----
[root@ip-10-0-9-101 ec2-user]#

```

Figura 2.15: cargamos nuestro archivo de llave

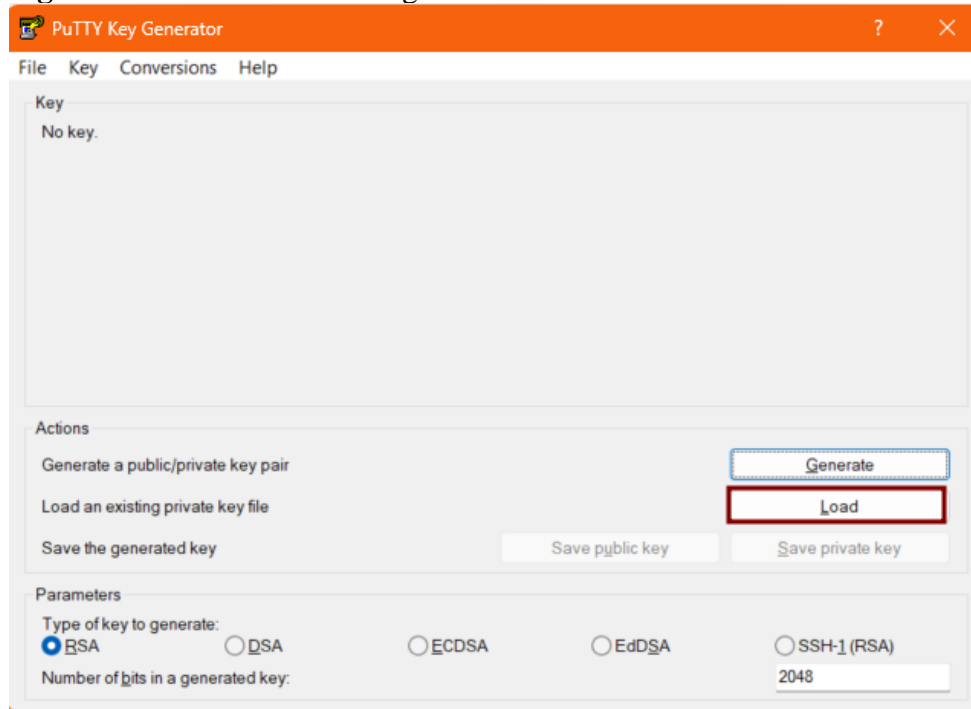


Figura 2.16: El archivo se obtiene al momento de crear nuestra instancia.

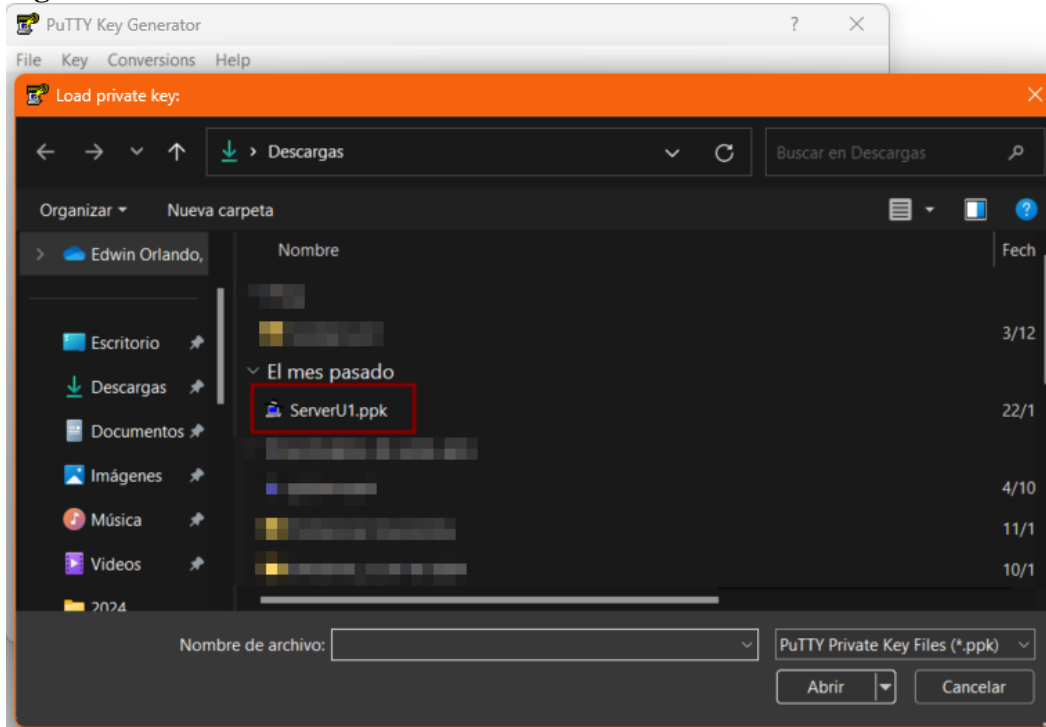


Figura 2.17: Un vistazo a la ventana del gestor de llaves de PUTTY

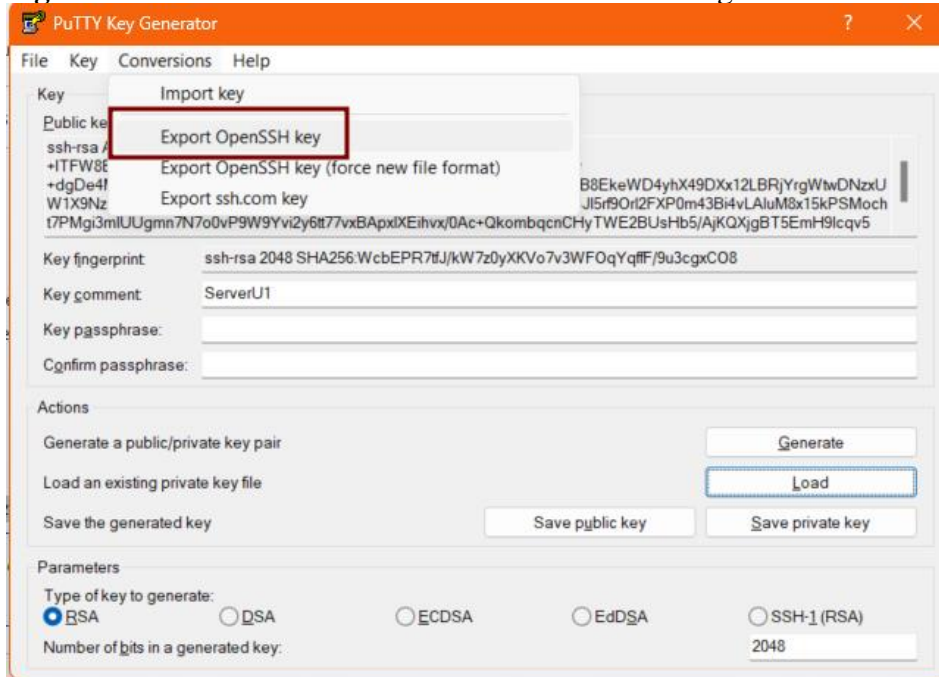


Figura 2.18: Observando nuestra llave desde el block de notas de windows

```

Windows.pem
Archivo  Editar  Ver

|-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA2o55UfPuxVvBBsdvNwtHwUsCzRg4FAf7Egtg9m+W+GSrpYwi
nt/AJRaUyfdCALSyXL/nYA3uDP7BZfc1WHetQWLG+Fnip1tzyPAacVtVTwT0bW48
eFFoHtTFF3QfBJH1g+MoV+PQ18ddiwUY2K4FrcAzC8VftV/Tc3n9r+dVBSa0ZCB1
eKf3m9w2aP5g1yWkDFuWYQAC+0AFUWkKwEiS0a3/TqyNhVz9JuNwYyLywJbJpMd
eZD0jKHIBEzIIt5pFIJp+ze6NLz/VvWL4tsurbe+78QQKcZVxIob8f9AHPkJKJ
m6nJwh8k1hNgVLB2+fWYkF44AU+Rjh/ZXKr+QIDAQABAoIBAADDgK8FpeguouUI
K15Q3sZRIchfrkp7oKf2fDdAOpHRoDvq2nAdrgkzI//0qj1gaEMktIVEZzGkb1iR
3ZDakjNpSfkyNbwJGkV57W25+P1Wthfvy6xL9IYvDjyBv14AuIGWeJyeAivO2YFT
07tmGET8J/HenZi9pfqAUZ+cu1JUu6BzWdurT4KuuTPpZA3XiXbDXK2o2AVevA+z
NWAczMiVafD+2cy8o2r54SzuaciXqeiF1gJKyZm1xBTNG99f+z1H2Bv08eEGIPes
8mVXIIE6KJRkrJ3pXaMSTW8v3oiPAVLZps7hxi2sxyHuqmVmIVGgpSTmUZf2ws4nD
1P1PH3UCGYEA/wu04ewAymvjjNgdVxwIZdOZ20D7NtnxbKbnThglvrqcLUBnan3q
+psUQk7M/24epP/Rr95IAkDYgX/T0FRoUVuz9yoa17fbTxIscMy6ax6QDsDwFw1B
e+AYvLueK7a+pALBmNXPvhGhcBhsVz0DY4Wg1XYA/YaSYNpkXujk/sCgYEA3Mft
iue2oFbWIrUazNlge7Vha/ORG65OYTLRZangXoq3Zw0Gfd/rss43zkStbg0wGdo
YSqarG4IH08A6CAkuVJ7V+qStPPXpcN4b0XG67ozc4Z7eCqgkF48vAghNjPZ08tPH
2S7XZ6Y0S0t1D6cgSa7f8exy4UBXIy9xKQEoyZsCgYEA7LhJdKfhbTQukIfn8XCJ
wqRQdCvWONocxfNBNP1n13uaQ66C22nq4Rk/jNhp1AU7AfStTcrIH90P11LYFUY
RLcmeoCn6oThpRDS1mNG13zrD/ztfwbw3mfMc6/F5euENOWmLz+6GE+bv3vw5+Sk

Ln 1, Col 1 | 1,679 caracteres. | 100%

```

Figura 2.19: Las direcciones IPv4 privadas de nuestra instancia...

The screenshot displays the AWS Management Console for an EC2 instance. The main content area shows the 'Resumen de instancia' (Instance Summary) for instance ID `i-055a3ad3b06b2ac9e`. A red box highlights the 'Direcciones IPv4 privadas' (Private IPv4 addresses) field, which lists the address `10.0.5.120`. Other visible fields include 'Dirección IPv4 pública', 'Estado de la instancia' (En ejecución), 'Nombre DNS de IP privada (solo IPv4)', 'Tipo de instancia' (t2.micro), 'ID de VPC', 'ID de subred', and 'ARN de instancia'. The console interface includes a left-hand navigation menu with categories like 'Instancias', 'Imágenes', 'Elastic Block Store', 'Red y seguridad', and 'Equilibrio de carga'. The top navigation bar shows the instance ID and various action buttons like 'Conectar', 'Estado de la instancia', and 'Acciones'.

Figura 2.20: Nos conectamos a esa instancia por SSH

```

root@ip-10-0-9-101:/home/ec2-user
K15Q3sZRIchfrkp7oKf2fDdAOpHRoDvq2nAdrgkzI//0qj1gaEMktIVEZzGKbliR
3ZDakjNpSfKyNbWJGkV57W25+PlWthfvy6xL9IYvDjyBv14AuIGWeJyeAivO2YfT
07tmGET8J/HenZi9pfqAUZ+culJUu6BzwDurT4KuuTPp2A3XixbDXK2o2AVevA+z
NWACzMivafD+2cy8o2r54SzuaciXqeiFlgJKyZmlxBTNG99f+z1H2Bv08eEGIPes
8mVXIi6KJRkrJ3pXaMSTW8v3oiPAVLZps7hxi2sxyHuqmVmIVGgpSTmUZf2ws4nD
1P1PH3UCgYEA/Wu04ewAymvjjNgdvxWIZdO220D7NtnxbKbnThglvrqcLUBnan3q
+psUQk7M/24epP/Rr95IAkDYgX/T0FRoUVuz9yoal7fbTxIscMy6ax6QDsDWFw1B
e+AYvLueK7a+pALBmNXpvhGhcBhsVz0DY4WVglXYA/YaSYNpkXujk/sCgYEA3Mft
iue2oFbWirUazNgke7Vha/ORG65OYTLRZangXoq3ZWOgFd/rsss43zkStbgOwGdo
YSqarG4IHO8A6CAkuVJ7V+qStPPXpcN4bOX67ozc4Z7eCqgkf48vAghNjPZ08tPH
2SjXZ6Y0Sot1D6cgSa7f8exy4UBXIy9xKQEoyZsCgYEA7LhJdKfhtTQuKifn8XCJ
wgRQgdcvWONocxfNBNPlni3uaQ66C22nq4Rk/jNhplAU7AfStTcriH9OPl1LYFUy
RLcmeoCn6oThpRDSlmNG13zrD/ztfwbw3mFMc6/F5guENOWmLz+6GE+bv3vw5+Sk
X7iOmFraF4kxsO3CFRwrDTUCgYEAaggulrQs9tIZRzdHVfnGWU9QMFxu7tQlgF7rC
QtrkUWvuCOe85QWnpQIldq78iVNoRQbkIc/hmc2aM8jIRPKiDxM5mhi038196aMQ
7TwuKZOJi9BzQ9202foyg0VewLhY2onKfujOiTwKMiWNusGbsCs0uCCprmqiqRQJ
2LhZGR8CgYBs9ql/iz+nvx7HUos08MgUQFDXiI41zP3DPpngHgmQeB3hNw0wsK4N
1Tik/FALBws7qXP2NwhOaDj7mNruw10JYUVtpBr5W0DnHuexezH+T1f+4AitTVS1
gfhtKK+eDssSYqZpTVSKxeE4JLN5OWO2QzKd0Cb510B1DkxTtd0AhQ==
-----END RSA PRIVATE KEY-----
[root@ip-10-0-9-101 ec2-user]#
[root@ip-10-0-9-101 ec2-user]#
[root@ip-10-0-9-101 ec2-user]#
[root@ip-10-0-9-101 ec2-user]# ssh -i cert.PEM ec2-user@10.0.5.120

```

Figura 2.21: Accedemos a la instancia.

The figure consists of two parts. The top part is a screenshot of the AWS Management Console showing the details of an EC2 instance. The 'Connect' button is highlighted, and a tooltip displays the private IPv4 address '10.0.5.120'. The 'Actions' dropdown menu is also visible. The bottom part is a screenshot of a terminal window titled 'ec2-user@ip-10-0-5-120~'. The terminal shows the execution of the command 'ssh -i cert.PEM ec2-user@10.0.5.120', which successfully connects to the instance. The terminal output displays the Amazon Linux 2023 logo and the URL 'https://aws.amazon.com/linux/amazon-linux-2023'. The prompt changes from '[root@ip-10-0-9-101 ec2-user]#' to '[ec2-user@ip-10-0-5-120 ~]#', indicating a successful login.

Figura 2.22: Autenticándonos con nuestra llave pública.

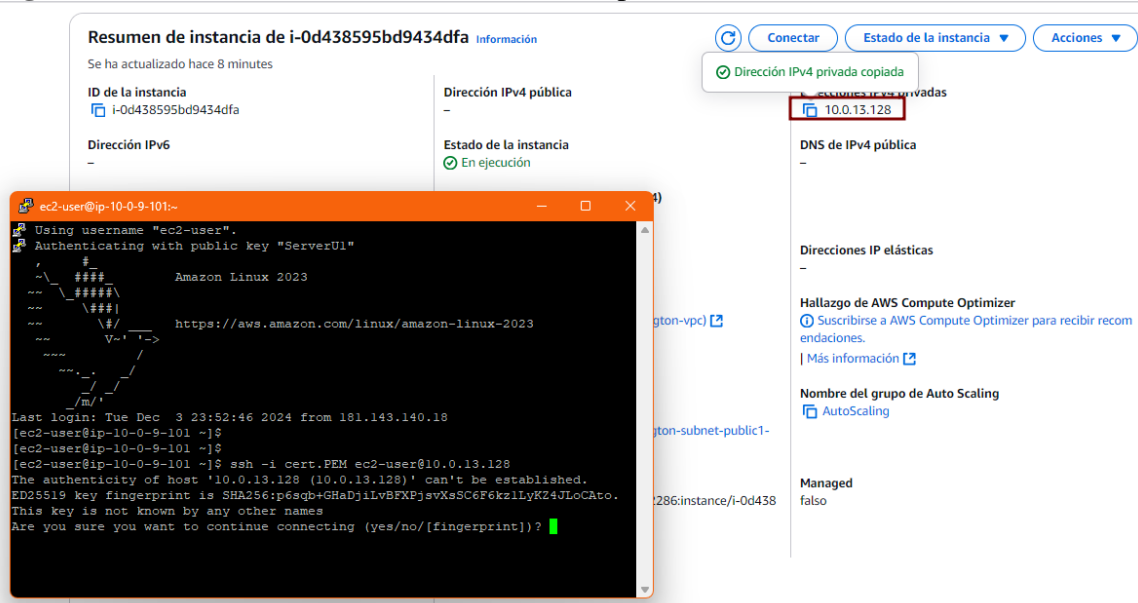


Figura 2.23: Detenemos el demonio HTTPD

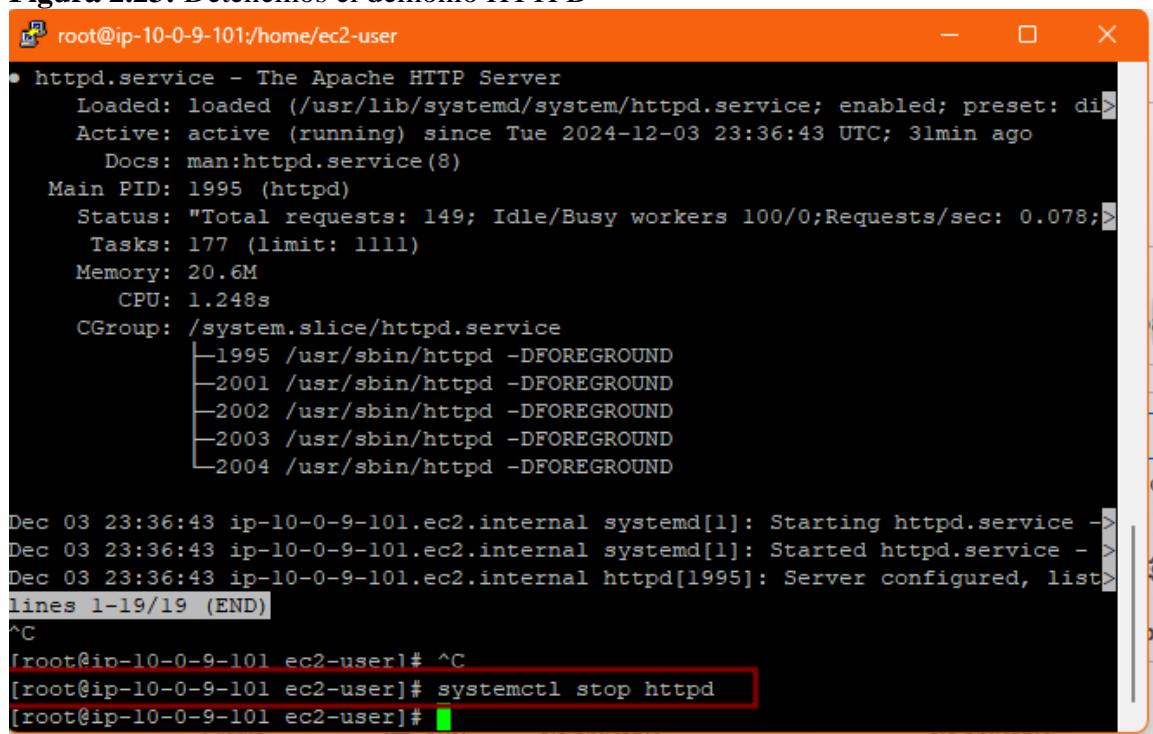


Figura 2.24: Provocando la caída

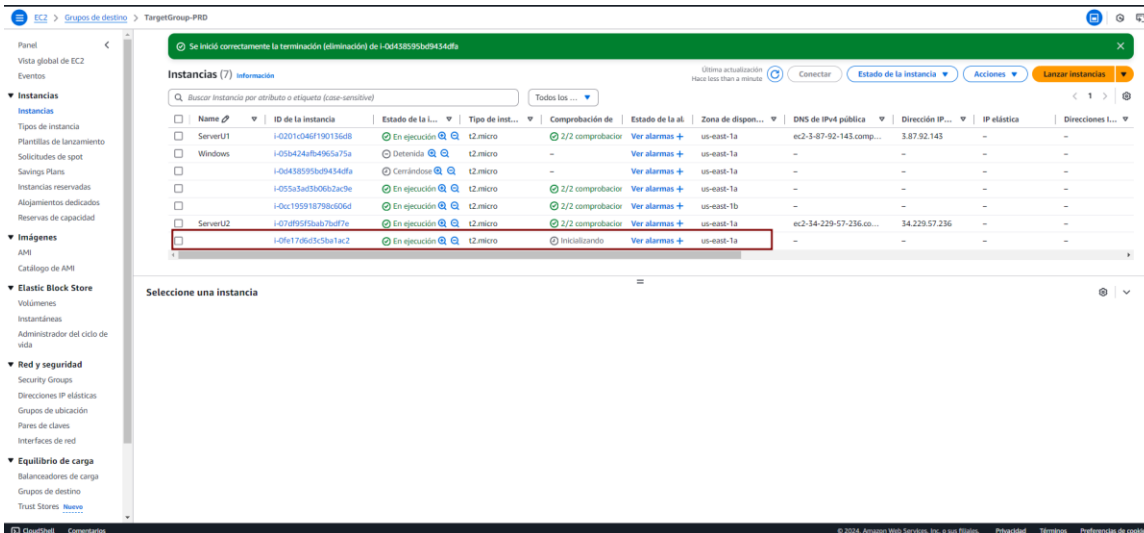


Figura 2.25: Aquí observamos los estados de nuestras instancias.

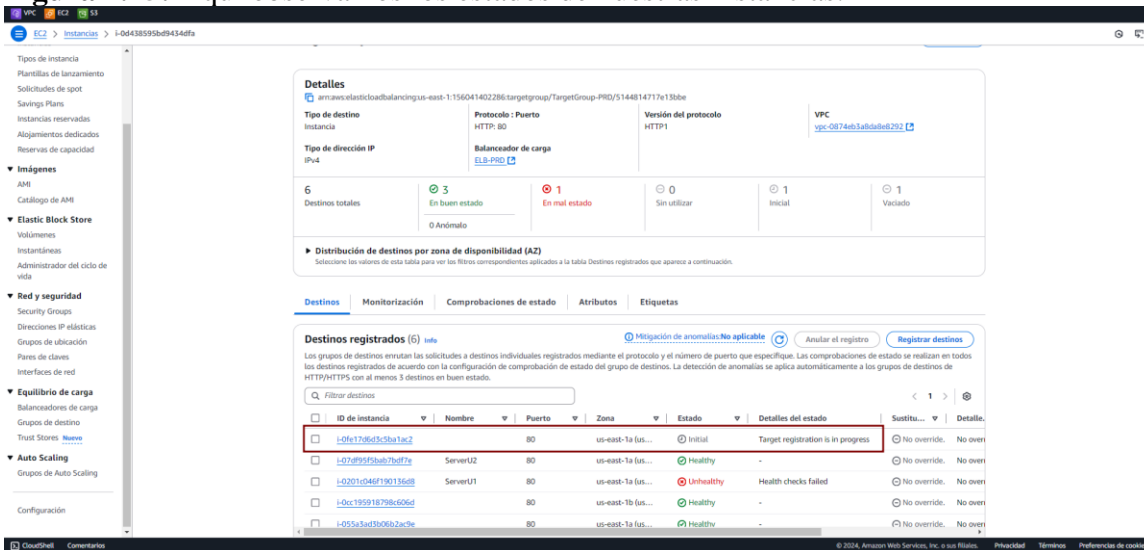


Figura 2.26: Nuevamente el servicio activado.

```

ec2-user@ip-10-0-5-120:~
• httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: di
  Active: active (running) since Sat 2024-11-30 23:36:21 UTC; 3 days ago
  Docs: man:httpd.service(8)
  Process: 3689 ExecReload=/usr/sbin/httpd $OPTIONS -k graceful (code=exited,
  Main PID: 2009 (httpd)
  Status: "Total requests: 21339; Idle/Busy workers 100/0;Requests/sec: 0.08
  Tasks: 230 (limit: 1111)
  Memory: 27.1M
  CPU: 2min 55.852s
  CGroup: /system.slice/httpd.service
    └─ 2009 /usr/sbin/httpd -DFOREGROUND
       3692 /usr/sbin/httpd -DFOREGROUND
       3693 /usr/sbin/httpd -DFOREGROUND
       3694 /usr/sbin/httpd -DFOREGROUND
       3695 /usr/sbin/httpd -DFOREGROUND
       33022 /usr/sbin/httpd -DFOREGROUND
Nov 30 23:36:20 ip-10-0-9-101.ec2.internal systemd[1]: Starting httpd.service ->
Nov 30 23:36:21 ip-10-0-9-101.ec2.internal systemd[1]: Started httpd.service ->
Nov 30 23:36:21 ip-10-0-9-101.ec2.internal httpd[2009]: Server configured, list
Dec 01 00:00:02 ip-10-0-5-120.ec2.internal systemd[1]: Reloading httpd.service ->
Dec 01 00:00:02 ip-10-0-5-120.ec2.internal systemd[1]: Reloaded httpd.service ->
lines 1-23

```

Entrega 3

Se implementa una arquitectura en AWS que cumpla con todos los requisitos principales:

- Balanceador de carga.
- Instancias EC2.
- Proxy reverso.
- Auto escalado
- Seguridad.

En el siguiente link se hace evidencia sobre todo lo realizado:

https://drive.google.com/file/d/1kgtU2mS1Hij4uXTr9QmryrR1WW03cX6a/view?usp=drive_link

Figura 3.1: Evidencia de la creación de las App en carpetas por separado.

```

[root@ip-10-0-9-101 /]# ls
APP1  APP2  APP3  bin  boot

```

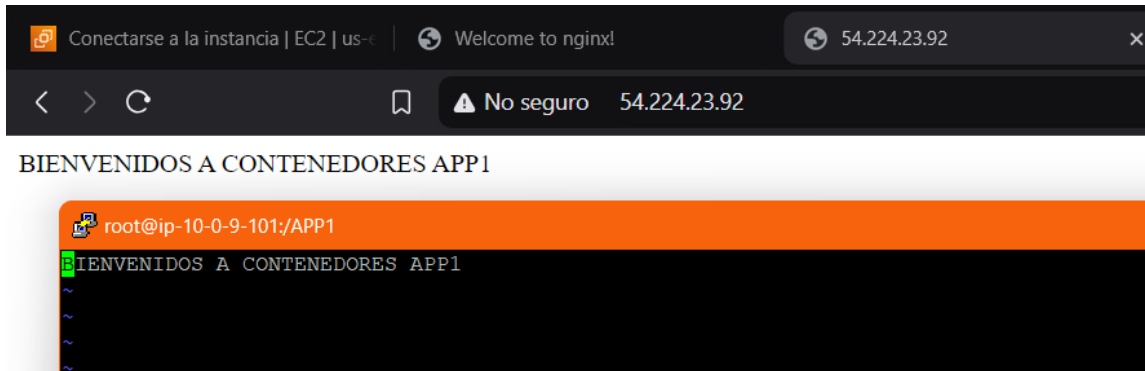


Figura 3.2: Creación de los contenedores.

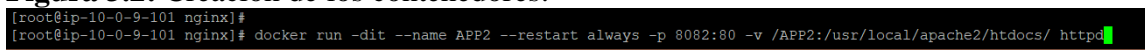


Figura 3.3: Lista de los contenedores ejecutándose con los puertos y nombres.

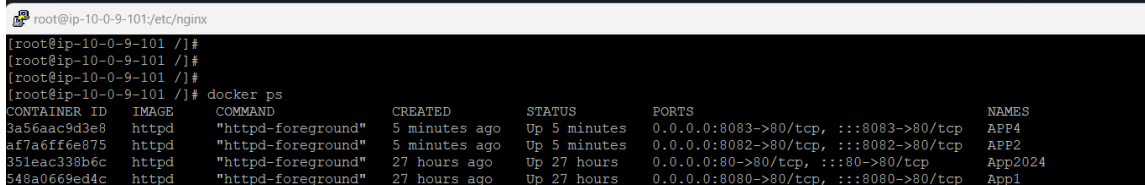


Figura 3.4: Ingreso a la una de las APP's por medio del puerto 8082.

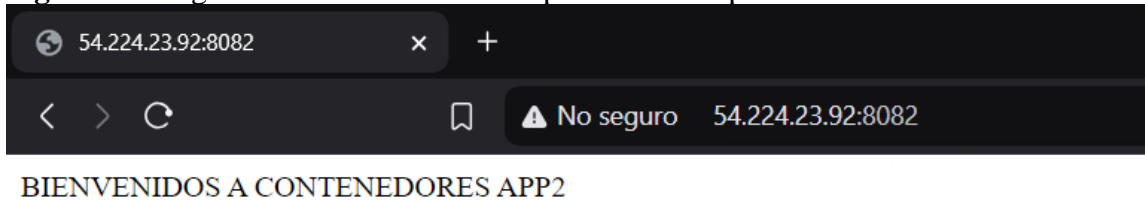


Figura 3.5: Como ingresamos a la configuración del nginx.

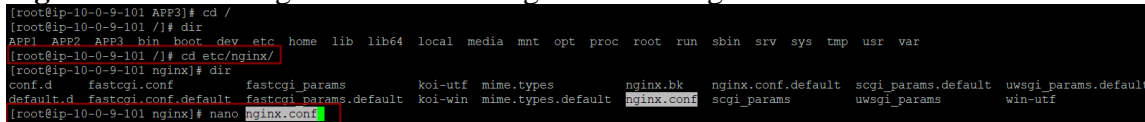
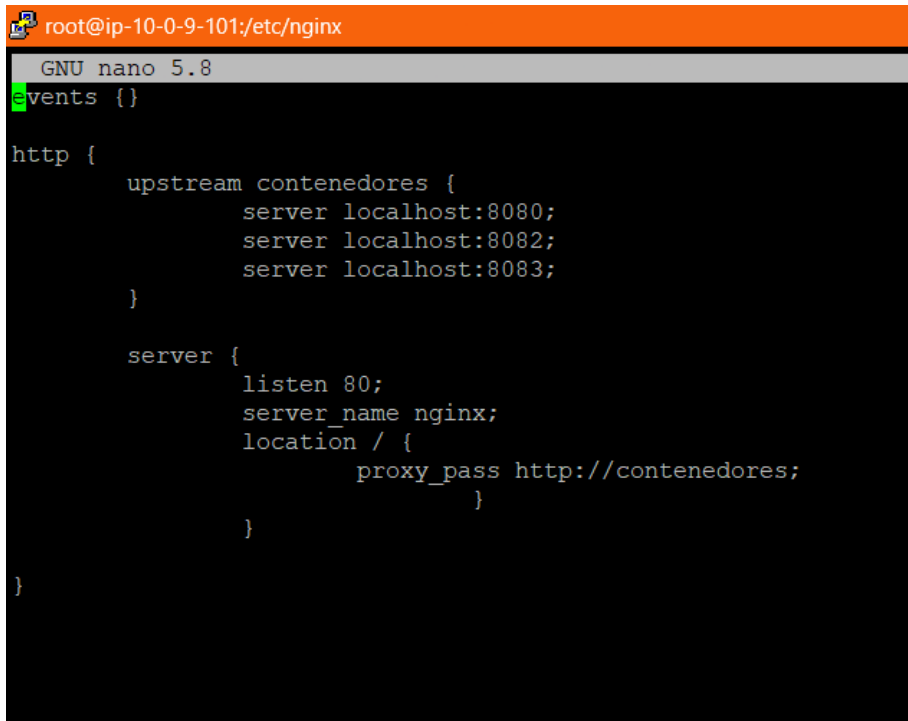


Figura 3.6: Configuración de Nginx.

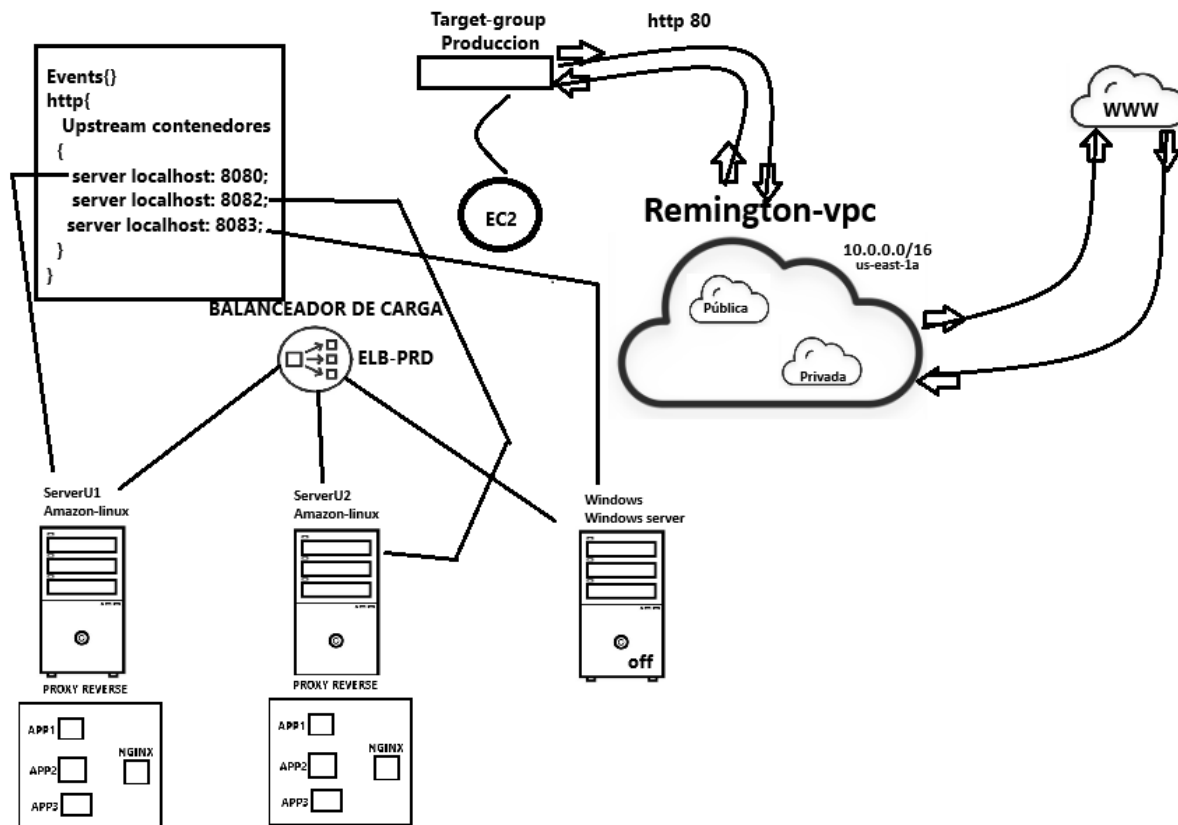
A screenshot of a terminal window showing the configuration of an nginx server in nano editor. The terminal title is "root@ip-10-0-9-101:/etc/nginx". The editor shows the following configuration:

```
GNU nano 5.8
events {}

http {
    upstream contenedores {
        server localhost:8080;
        server localhost:8082;
        server localhost:8083;
    }

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

Figura 4: Diagrama de recursos.



Conclusiones

Como grupo y colegas estamos bastante satisfecho con los resultados obtenidos. Implementar una aplicación en AWS es un proceso que requiere planificación y

conocimientos técnicos, pero es muy gratificante ver cómo una idea se convierte en una aplicación en funcionamiento en la nube.

Esta implementación de arquitectura en AWS nos aportó un conocimiento importante en cómo construir arquitecturas escalables utilizando balanceadores de carga para distribuir el tráfico entre varias instancias, además una ventaja relevante de emplear contenedores en Docker para empaquetar aplicaciones, facilitando de esta manera despliegues más rápidos.

!

Referencias

Mathew, S., & Varia, J. (2014). Overview of amazon web services. Amazon Whitepapers, 105(1), 22.

- Cloud, A. E. C. (2011). Amazon web services. Retrieved November, 9(2011), 2011.
- Wittig, A., & Wittig, M. (2023). Amazon Web Services in Action: An in-depth guide to AWS. Simon and Schuster.
- Hashemipour, S., & Ali, M. (2020, August). Amazon web services (aws)—an overview of the on-demand cloud computing platform. In International Conference for Emerging Technologies in Computing (pp. 40-47). Cham: Springer International Publishing.
- Talha, M., Sohail, M., & Hajji, H. (2020). Analysis of research on amazon AWS cloud computing seller data security. International Journal of Research in Engineering Innovation, 4(3), 131-136.
- Fusaro, V. A., Patil, P., Gafni, E., Wall, D. P., & Tonellato, P. J. (2011). Biomedical cloud computing with amazon web services. PLoS computational biology, 7(8), e1002147.
- Murty, J. (2008). Programming amazon web services: S3, EC2, SQS, FPS, and SimpleDB. " O'Reilly Media, Inc."