

TRABAJO DE GRADO

Opción Seminario-Diplomado.

Implementación de los servicios de AWS

Corporación Universitaria Remington.

Facultad De Ingenierías

Ingeniería de Sistemas

Eduard Alexander Múnera González.

Juan Pablo Berrio López.

Opción de Trabajo de grado Seminario-Diplomado.

2024.

Tabla de Contenidos

Resumen	3
Palabras clave	4
Marco conceptual y contextual	5
Desarrollo e implementación del aprendizaje.....	7
Configuración de un de un contenedor desde una instancia de Linux.....	7
Conexión de una instancia AWS en Linux para que pueda cargar archivos en S3 y conexión a una BD	16
Creación de un servidor con servicio web y funcionamiento	26
Conclusiones.....	50
Referencias	51

Resumen

Este trabajo exploraremos Amazon Web Services (AWS) un referente a nivel mundial en el ámbito de la computación en la nube. Desde su lanzamiento en el 2006, ha sido pionero en la oferta de servicios en la nube, ofreciendo una amplia gama de soluciones. Su enfoque en la innovación constante y la mejora continua de sus servicios ha sido clave para su éxito y su posición dominante en el mercado, por esto compartiremos algunas de sus herramientas o los servicios más relevante de la plataforma AWS, implementando y ampliando el conocimiento visto durante el seminario.

Palabras clave

Amazon Web Services, Computación en la nube, Linux, EC2, RDS.

Marco conceptual y contextual

La virtualización se remota a la década De los 60 y 70, y se le puede agradecer a la empresa IBM quienes desarrollaron el CP-40, el CP-67 y VM/370, estos dos primeros fueron sistemas que permitieron la ejecución de varios o múltiples sistemas operativos en un solo mainframe creando así las máquinas virtuales, y tecnologías de virtualización con sistemas como VM/370, permitieron a los usuarios crear y gestionar múltiples máquinas virtuales. Este sistema establecido los principios básicos de la virtualización moderna.

Con esta tecnología lograron mejorar la utilización del hardware de mainframe, permitiendo a diferentes usuarios y procesos compartir los recursos de manera eficiente.

Otra de las grandes marcas y a las que se les debe este término es VMware Workstation, que en el año 1999 revolucionó la virtualización, especialmente en plataformas x86. VMware permitió a los usuarios ejecutar múltiples sistemas operativos en una sola máquina física, reduciendo costos y aumentando la flexibilidad.

El termino cloud computing o computación en la nube, se originó en la década de los 60 y sus primeros pinos fueron gracias a lo anterior mencionado de la virtualización, sin embargo, fue hasta el 2006 en el que Amazon, lanza Amazon Web Services (AWS), un servicio EC2 (Elastic Compute Cloud), que permitía a las empresas alquilar servidores virtuales y pagar solo por el uso que hacían. Esto marcó un hito en la computación en la nube, popularizando el modelo de "infraestructura como servicio" (IaaS).

Otra de las grandes empresas que incursiona en la computación en la nube fue Google al lanzar Google App Engine en 2008, el cual permitió a los desarrolladores crear aplicaciones que se ejecutaran en la infraestructura de Google.

Más tarde Microsoft ingresa al mercado con Azure en 2010, ampliando aún más las posibilidades y servicios en la nube, se convirtió en la principal competencia de AWS.

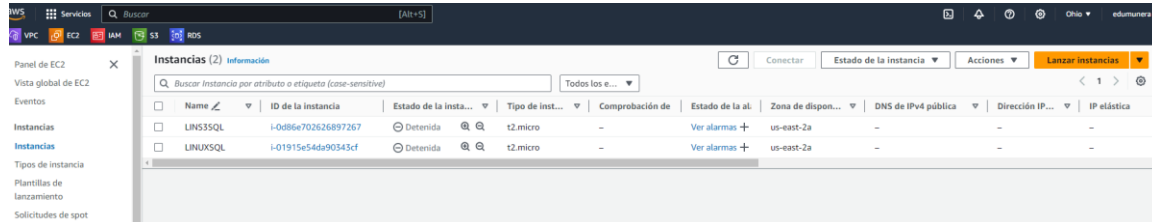
Desarrollo e implementación del aprendizaje

Configuración de un de un contenedor desde una instancia de Linux

Previamente se debe crear una instancia en la plataforma de AWS, esta instalación se realizó con una imagen de Linux Amazon 2 suministrada por la misma plataforma. Visualizar Figura 1

Figura 1

Instancia AWS



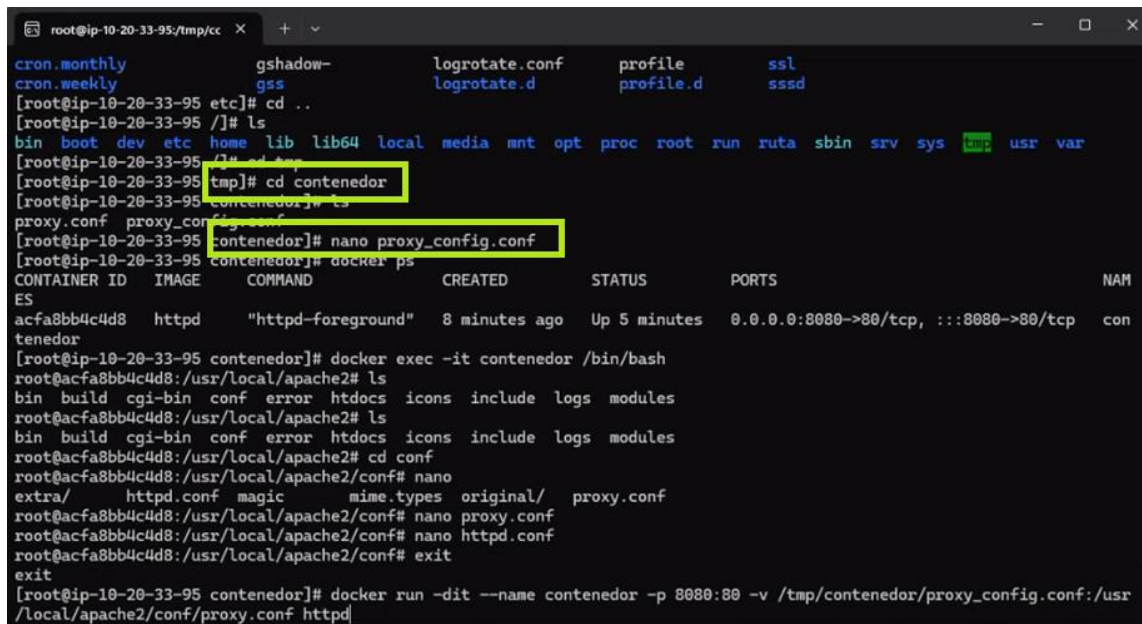
Name	ID de la instancia	Estado de la instancia	Tipo de instancia	Comprobación de	Estado de la alarma	Zona de disponibilidad	DNS de IPv4 pública	Dirección IP	IP elástica
LINS3SQL	i-0d86e702626897267	Detenida	t2.micro	-	Ver alarmas +	us-east-2a	-	-	-
LINUXSQL	i-01915e54da90543cf	Detenida	t2.micro	-	Ver alarmas +	us-east-2a	-	-	-

Con las instancias creadas, y luego de conectarse a través del protocolo SSH desde cualquier herramienta que existen para este procedimiento (en este caso se usó WSL)

crearemos una carpeta en la ubicación tmp y la llamaremos contenedor Figura 2, y dentro de esta carpeta crearemos con el comando nano un archivo llamado proxy_config.conf y en este escribiremos las líneas como se reflejan en la Figura 3, la ejecución del comando Docker run para la instalación de nuestro Docker Figura 4 y la visualización de que si está funcionando Figura 5.

Figura 2

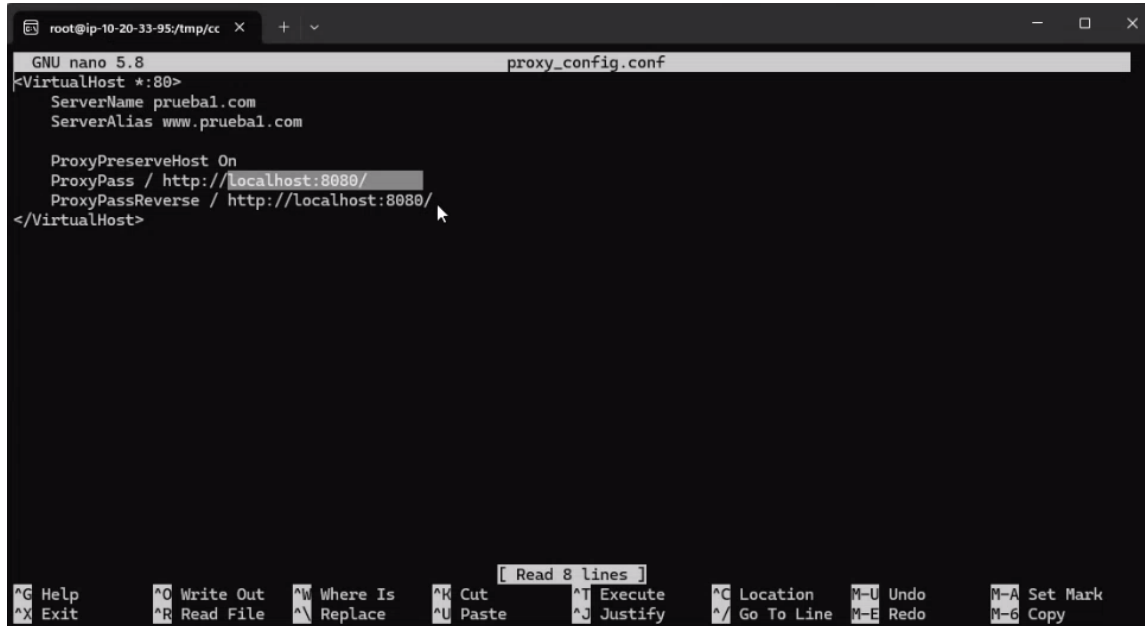
Crear carpeta y crear archivo de configuración proxy_config.conf



```
root@ip-10-20-33-95/tmp/cc x + v
cron.monthly      gshadow-          logrotate.conf    profile           ssl
cron.weekly       gss               logrotate.d       profile.d         sssd
[root@ip-10-20-33-95 etc]# cd ..
[root@ip-10-20-33-95 /]# ls
bin boot dev etc home lib lib64 local media mnt opt proc root run ruta sbin srv sys tmp usr var
[root@ip-10-20-33-95 tmp]# cd contenedor
[root@ip-10-20-33-95 contenedor]# ls
proxy.conf proxy_config.conf
[root@ip-10-20-33-95 contenedor]# nano proxy_config.conf
[root@ip-10-20-33-95 contenedor]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                NAMES
acfa8bb4c4d8  httpd    "httpd-foreground"     8 minutes ago Up 5 minutes   0.0.0.0:8080->80/tcp, :::8080->80/tcp  contenedor
[root@ip-10-20-33-95 contenedor]# docker exec -it contenedor /bin/bash
root@acfa8bb4c4d8:/usr/local/apache2# ls
bin build cgi-bin conf error htdocs icons include logs modules
root@acfa8bb4c4d8:/usr/local/apache2# ls
bin build cgi-bin conf error htdocs icons include logs modules
root@acfa8bb4c4d8:/usr/local/apache2# cd conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano
extra/ httpd.conf magic mime.types original/ proxy.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano proxy.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano httpd.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# exit
exit
[root@ip-10-20-33-95 contenedor]# docker run -dit --name contenedor -p 8080:80 -v /tmp/contenedor/proxy_config.conf:/usr/local/apache2/conf/proxy.conf httpd
```


Figura 3

Líneas a escribir en el archivo proxy_conf.conf



```
root@ip-10-20-33-95/tmp/cc x + v
GNU nano 5.8 proxy_conf.conf
<VirtualHost *:80>
  ServerName prueba1.com
  ServerAlias www.prueba1.com

  ProxyPreserveHost On
  ProxyPass / http://localhost:8080/
  ProxyPassReverse / http://localhost:8080/
</VirtualHost>

[ Read 8 lines ]
^G Help      ^O Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo     M-A Set Mark
^X Exit      ^R Read File  ^N Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo     M-C Copy
```

Nota. Localhost debe reemplazarse por nuestro nombre de sitio.

Figura 4

Instalación del Docker

```

root@ip-10-20-33-95/tmp/cc  x  +  v  -  □  x
cron.monthly      gshadow-      logrotate.conf   profile          ssl
cron.weekly       gss             logrotate.d      profile.d        sssd
[root@ip-10-20-33-95 etc]# cd ..
[root@ip-10-20-33-95 /]# ls
bin boot dev etc home lib lib64 local media mnt opt proc root run ruta sbin srv sys tmp usr var
[root@ip-10-20-33-95 /]# cd tmp
[root@ip-10-20-33-95 tmp]# cd contenedor
[root@ip-10-20-33-95 contenedor]# ls
proxy.conf proxy_config.conf
[root@ip-10-20-33-95 contenedor]# nano proxy_config.conf
[root@ip-10-20-33-95 contenedor]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                    NAM
ES
acfa8bb4c4d8   httpd    "httpd-foreground"     8 minutes ago  Up 5 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp  con
tenedor
[root@ip-10-20-33-95 contenedor]# docker exec -it contenedor /bin/bash
root@acfa8bb4c4d8:/usr/local/apache2# ls
bin build cgi-bin conf error htdocs icons include logs modules
root@acfa8bb4c4d8:/usr/local/apache2# ls
bin build cgi-bin conf error htdocs icons include logs modules
root@acfa8bb4c4d8:/usr/local/apache2# cd conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano
extra/ httpd.conf magic mime.types original/ proxy.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano proxy.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano httpd.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# exit
exit
[root@ip-10-20-33-95 contenedor]# docker run -dit --name contenedor -p 8080:80 -v /tmp/contenedor/proxy_config.conf:/usr
/local/apache2/conf/proxy.conf httpd

```

Nota. El nombre de la instalación se usó como ejemplo contenedor. Imagen propia

Figura 5

Verificación de la instalación del Docker

```

root@ip-10-20-33-95:/tmp/cc x + v
bashrc exports krb5.conf nsswitch.conf rsyslog.d tmpfiles.d
bindresvport.blacklist exports.d krb5.conf.d openldap rwtab.d trusted-key.key
binfmt.d filesystems ld.so.cache opt sasl2 udev
chkconfig.d fstab ld.so.conf os-release screenrc update-motd.d
chrony.conf gcrypt ld.so.conf.d pam.d security vimrc
chrony.d gnupg libaudit.conf passwd selinux virg
chrony.keys groff libbverbs.d passw services wgetrc
cifs-utils group libnl pkcs11 sestatus.conf xattr.conf
cloud group libreport pki pkgconfig shadow xdg
cni grub.d libuser.conf pm shadow- yum.repos.d
containerd grub2-efi.cfg locale.conf popt.d shells
cron.daily grub2.cfg localtime popt.d skel
cron.hourly gshadow login.defs printcap ssh
cron.monthly gshadow- logrotate.conf profile.d ssl
cron.weekly gss logrotate.d profile.d sssd
[root@ip-10-20-33-95 etc]# cd ..
[root@ip-10-20-33-95 /]# ls
bin boot dev etc home lib lib64 local media mnt opt proc root run ruta sbin srv sys usr var
[root@ip-10-20-33-95 /]# cd tmp
[root@ip-10-20-33-95 tmp]# cd contenedor
[root@ip-10-20-33-95 contenedor]# ls
proxy.conf proxy_config.conf
[root@ip-10-20-33-95 contenedor]# nano proxy_config.conf
[root@ip-10-20-33-95 contenedor]# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAME
ES
acfa8bb4c4d8 httpd "httpd-foreground" 8 minutes ago Up 5 minutes 0.0.0.0:8080->80/tcp, :::8080->80/tcp con
tenedor
[root@ip-10-20-33-95 contenedor]#

```

Habilitar los módulos del proxy, ingresado por medio del editor de texto nano al archivo httpd.conf Figura 6 y estando en el archivo buscar los loadModule de proxy y quitarles el carácter #.

Figura 6

Configuración del archivo httpd.conf

```

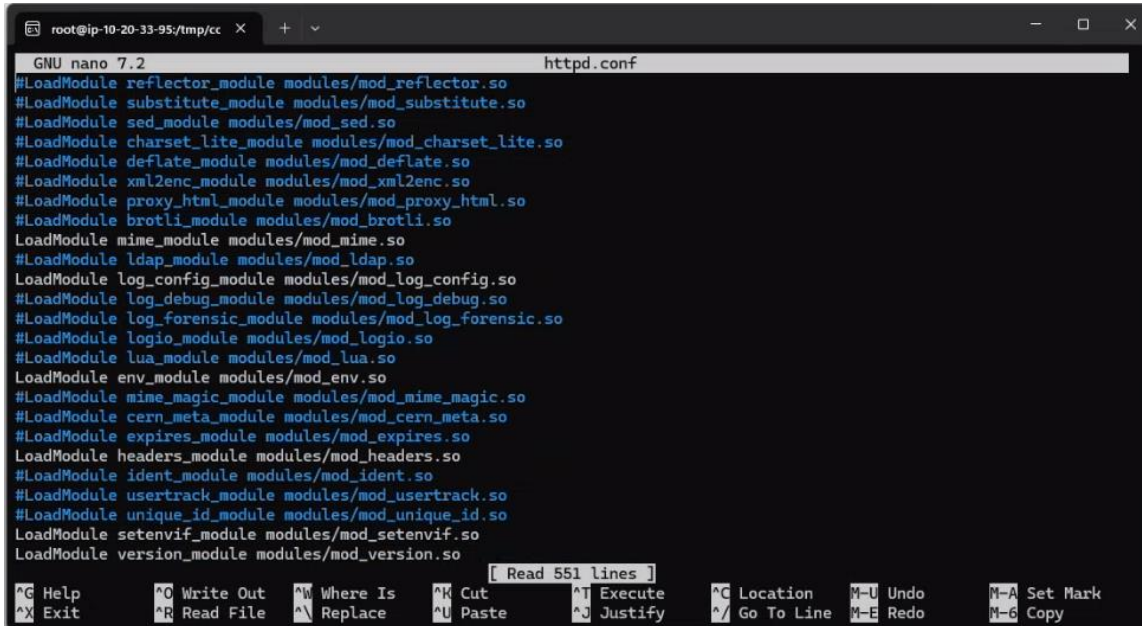
root@ip-10-20-33-95/tmp/cc x + v - □ x
containerd      grub2-efi.cfg  locale.conf    pm             shells
cron.daily      grub2.cfg      localtime     popt.d         skel
cron.hourly     gshadow        login.defs     printcap       ssh
cron.monthly    gshadow-       logrotate.conf profile         ssl
cron.weekly     gss            logrotate.d    profile.d      sssd
[root@ip-10-20-33-95 etc]# cd ..
[root@ip-10-20-33-95 /]# ls
bin boot dev etc home lib lib64 local media mnt opt proc root run ruta/sbin/srv/sys tmp/usr/var
[root@ip-10-20-33-95 /]# cd tmp
[root@ip-10-20-33-95 tmp]# cd contenedor
[root@ip-10-20-33-95 contenedor]# ls
proxy.conf proxy_config.conf
[root@ip-10-20-33-95 contenedor]# nano proxy_config.conf
[root@ip-10-20-33-95 contenedor]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS
acfa8bb4c4d8  httpd    "httpd-foreground"     8 minutes ago Up 5 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp
[root@ip-10-20-33-95 contenedor]# docker exec -it contenedor /bin/bash
root@acfa8bb4c4d8:/usr/local/apache2# ls
bin build cgi-bin conf error htdocs icons include logs modules
root@acfa8bb4c4d8:/usr/local/apache2# ls
bin build cgi-bin conf error htdocs icons include logs modules
root@acfa8bb4c4d8:/usr/local/apache2# cd conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano
extra/ httpd.conf magic mime.types original/ proxy.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf# nano httpd.conf
root@acfa8bb4c4d8:/usr/local/apache2/conf#

```

Nota. Para ingresar al archivo httpd.conf previamente debe estar en la ruta carpeta conf ingresando con el comando Docker exec

Figura 7

Visualización del archivo `httpd.conf`



```
root@ip-10-20-33-95/tmp/cc X + v
GNU nano 7.2 httpd.conf
#LoadModule reflector_module modules/mod_reflector.so
#LoadModule substitute_module modules/mod_substitute.so
#LoadModule sed_module modules/mod_sed.so
#LoadModule charset_lite_module modules/mod_charset_lite.so
#LoadModule deflate_module modules/mod_deflate.so
#LoadModule xml2enc_module modules/mod_xml2enc.so
#LoadModule proxy_html_module modules/mod_proxy_html.so
#LoadModule brotli_module modules/mod_brotli.so
LoadModule mime_module modules/mod_mime.so
#LoadModule ldap_module modules/mod_ldap.so
LoadModule log_config_module modules/mod_log_config.so
#LoadModule log_debug_module modules/mod_log_debug.so
#LoadModule log_forensic_module modules/mod_log_forensic.so
#LoadModule logio_module modules/mod_logio.so
#LoadModule lua_module modules/mod_lua.so
LoadModule env_module modules/mod_env.so
#LoadModule mime_magic_module modules/mod_mime_magic.so
#LoadModule cern_meta_module modules/mod_cern_meta.so
#LoadModule expires_module modules/mod_expires.so
LoadModule headers_module modules/mod_headers.so
#LoadModule ident_module modules/mod_ident.so
#LoadModule usertrack_module modules/mod_usertrack.so
#LoadModule unique_id_module modules/mod_unique_id.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule version_module modules/mod_version.so

[ Read 551 lines ]
^G Help      ^O Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo     M-A Set Mark
^X Exit      ^R Read File  ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line M-E Redo     M-G Copy
```

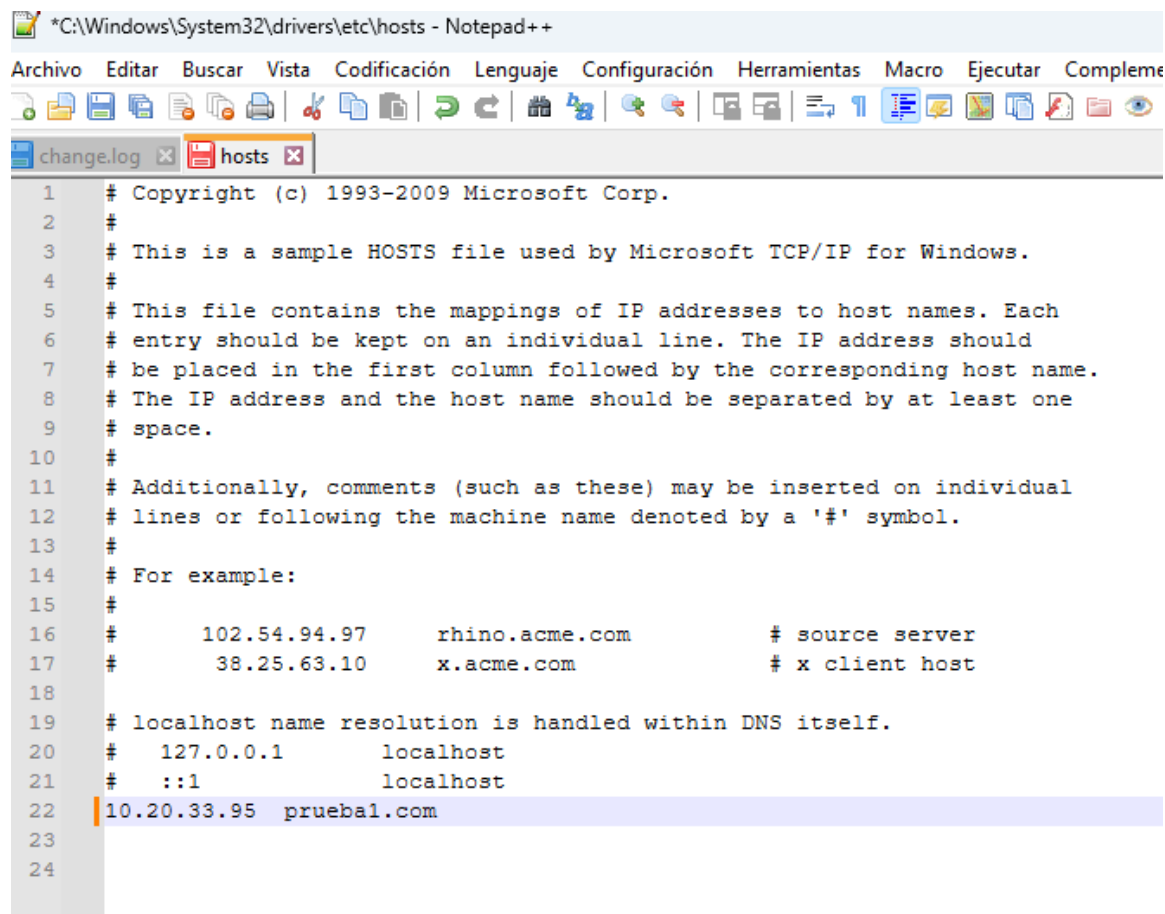
Nota. Se debe buscar las opciones de `LoadModule Proxy`, y se debe eliminar el carácter `#` el cual indica que esa línea es un comentario.

Para finalizar la práctica se debe configurar en nuestra maquina local el archivo host ubicado en C:\Windows\System32\drivers\etc\, aquí agregar la IP publica de la instancia y el nombre que se desea agregar Figura 8 y en el navegador realizar la consulta al sitio no nombre configurado

Figura 9.

Figura 8

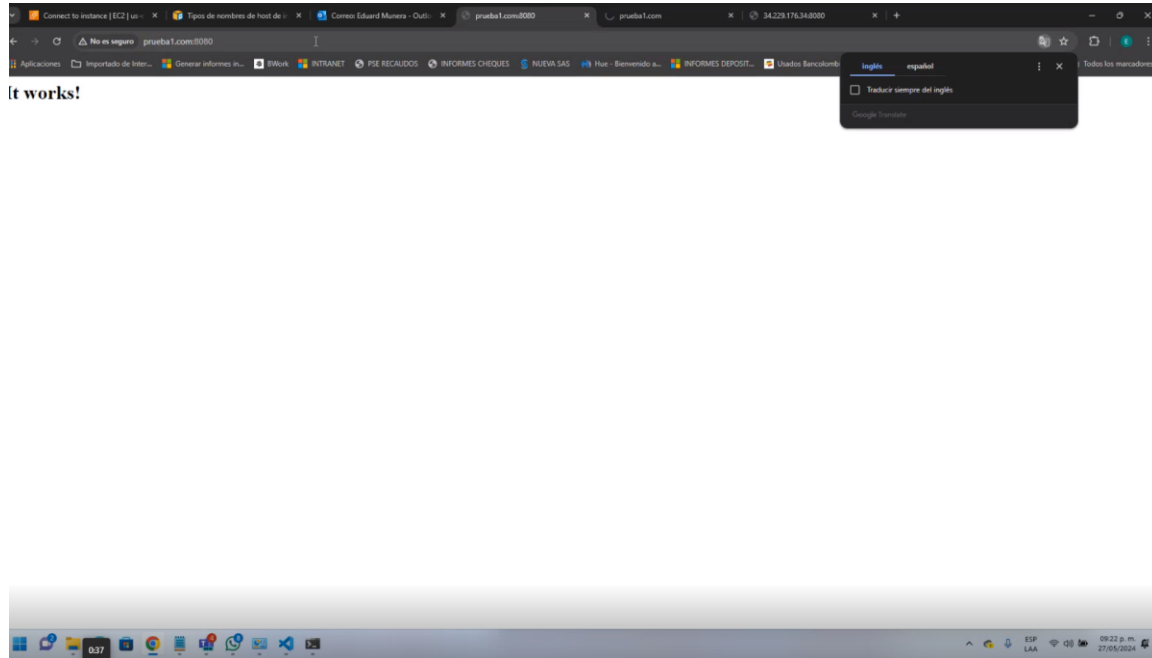
Configuración del archivo host



```
*C:\Windows\System32\drivers\etc\hosts - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro  Ejecutar  Compleme
change.log x hosts x
1  # Copyright (c) 1993-2009 Microsoft Corp.
2  #
3  # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
4  #
5  # This file contains the mappings of IP addresses to host names. Each
6  # entry should be kept on an individual line. The IP address should
7  # be placed in the first column followed by the corresponding host name.
8  # The IP address and the host name should be separated by at least one
9  # space.
10 #
11 # Additionally, comments (such as these) may be inserted on individual
12 # lines or following the machine name denoted by a '#' symbol.
13 #
14 # For example:
15 #
16 #     102.54.94.97      rhino.acme.com          # source server
17 #     38.25.63.10     x.acme.com              # x client host
18
19 # localhost name resolution is handled within DNS itself.
20 #   127.0.0.1        localhost
21 #   ::1              localhost
22 10.20.33.95 prueba1.com
23
24
```

Figura 9

Prueba de funcionamiento



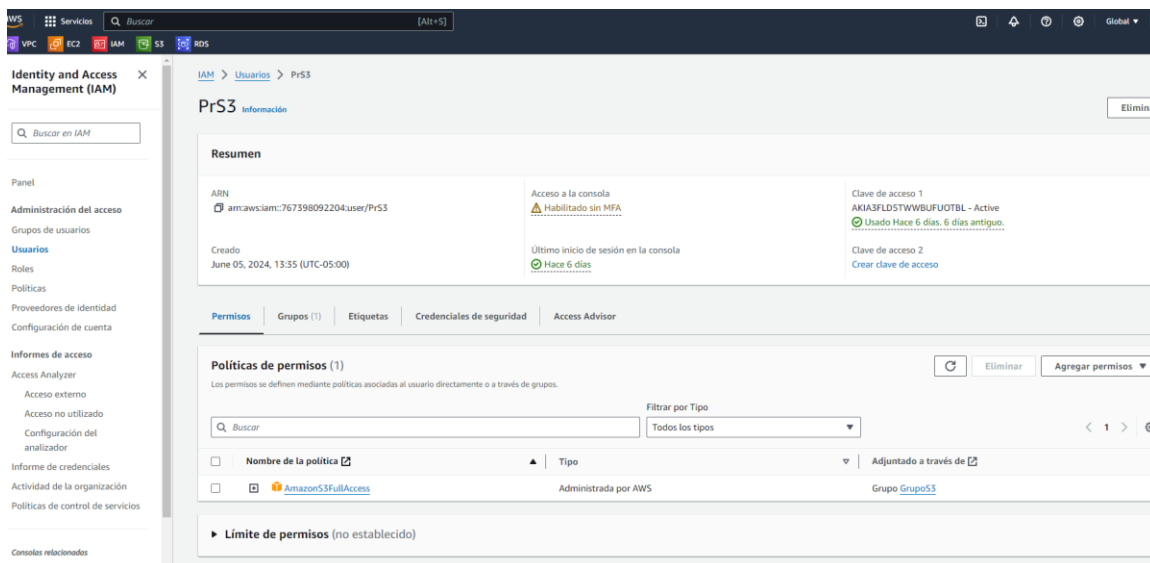
Nota. Esta página es de prueba si desean agregar una pagina mejor elaborada pueden agregar un índice nuevo.

Conexión de una instancia AWS en Linux para que pueda cargar archivos en S3 y conexión a una BD

Para realizar una carga de archivos en S3 se crea en el apartado de IAM un usuario y a este usuario le otorgamos permisos de acceso completo a S3 como se observa en la imagen 10.

Figura 10

Visualización de usuario creado



The screenshot displays the AWS IAM console interface. The main content area shows the details for a user named 'PrS3'. The 'Resumen' (Summary) section includes the ARN 'arn:aws:iam::767398092204:user/PrS3', console access status 'Habilitado sin MFA', and two active access keys. The 'Políticas de permisos (1)' section shows a single policy named 'AmazonS3FullAccess' administered by AWS, attached to the 'Grupo GrupoS3'.

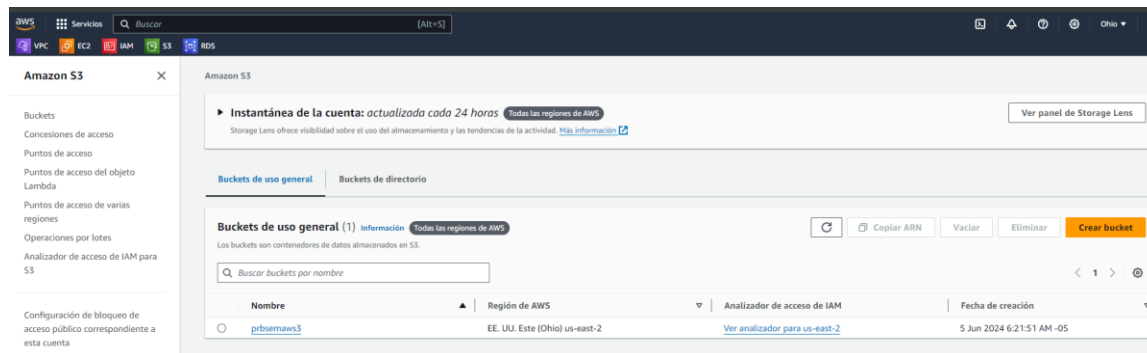
Resumen		
ARN arn:aws:iam::767398092204:user/PrS3	Acceso a la consola Habilitado sin MFA	Clave de acceso 1 AKIA3FLDSTWWBWFUOTBL - Active Usado Hace 6 días, 6 días antiguo.
Creado June 05, 2024, 13:35 (UTC-05:00)	Último inicio de sesión en la consola Hace 6 días	Clave de acceso 2 Crear clave de acceso

Políticas de permisos (1)		
<input type="checkbox"/> Nombre de la política	<input type="checkbox"/> Tipo	Adjuntado a través de
<input type="checkbox"/> AmazonS3FullAccess	Administrada por AWS	Grupo GrupoS3

Creamos por la opción S3 el Bucket

Figura 11

Visualización del Bucket



Nota. Este proceso se realizó en base a la clase 3 del seminario.

Creado el bucket ingresamos a la instancia con el usuario creado previamente

Con la conexión establecida, usamos el comando `aws s3 ls` para validar el bucket y el comando `aws s3 ls s3://prbsemaws3` visualizamos lo que contiene actualmente el bucket

Figura 13

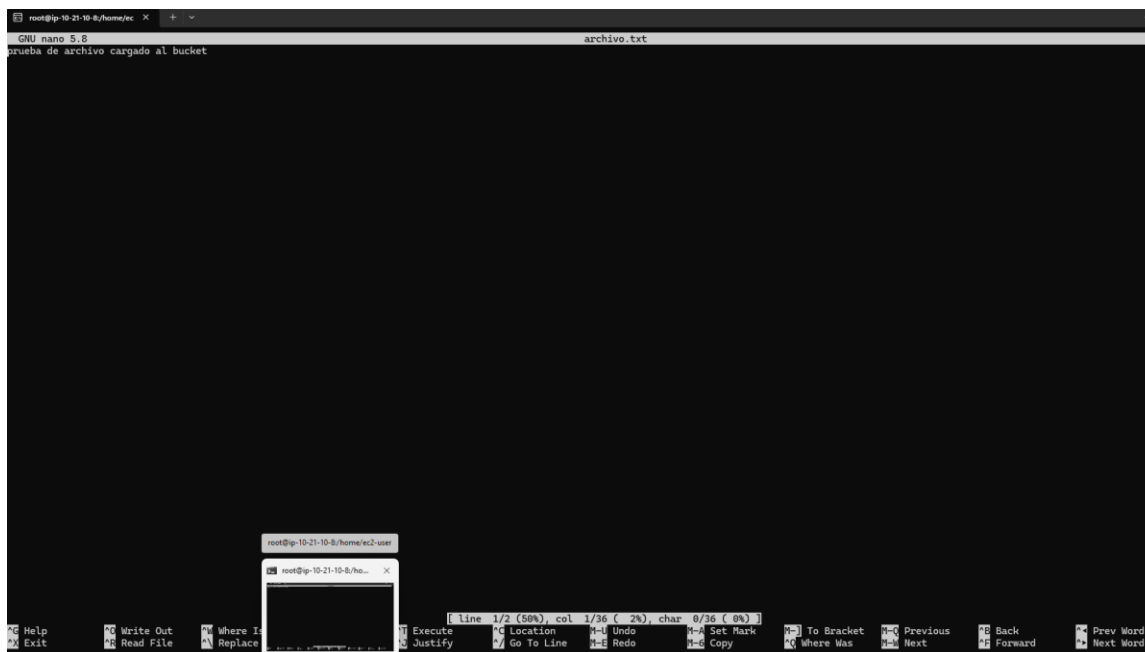
Visualizar el contenido del Bucket

```
[root@ip-10-21-10-8 ec2-user]# aws s3 ls
2024-06-05 11:21:52 prbsemaws3
[root@ip-10-21-10-8 ec2-user]# aws s3 ls s3://prbsemaws3
2024-06-05 17:59:53      110182 LINEA_TIEMPO.jpg
[root@ip-10-21-10-8 ec2-user]# ls
[root@ip-10-21-10-8 ec2-user]# pwd
/home/ec2-user
[root@ip-10-21-10-8 ec2-user]# nano archivo.txt
[root@ip-10-21-10-8 ec2-user]# ls
archivo.txt
[root@ip-10-21-10-8 ec2-user]# |
```

Con el comando `nano` creamos un archivo con el fin de realizar las pruebas.

Figura 14

Creando un archivo de prueba



Con el archivo creado usamos el comando `aws s3 cp /home/ec2-user/archivo.txt`

`s3://prbsemaws3` y visualizamos que carga el documento "archivo.txt" a la ruta indicada en este caso el bucket

Figura 15

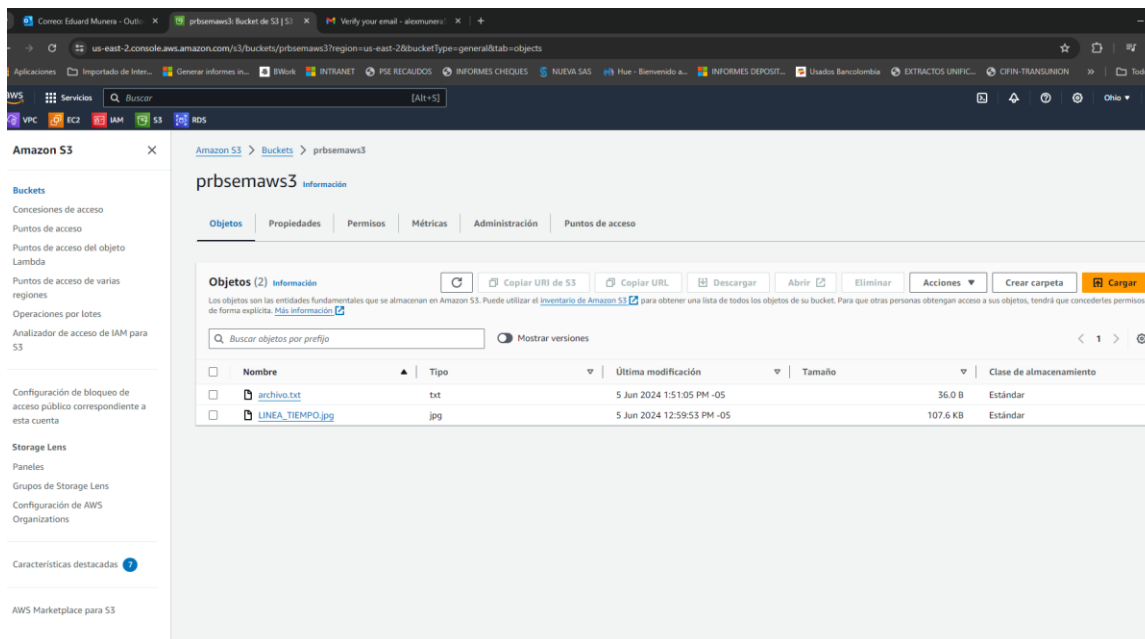
Cargando archivos al Bucket

```
[root@ip-10-21-10-8 ec2-user]# aws s3 cp /home/ec2-user/archivo.txt s3://prbsemaws3
upload: ./archivo.txt to s3://prbsemaws3/archivo.txt
[root@ip-10-21-10-8 ec2-user]# |
```

Visualizamos el bucket desde la consola y se observa el documento archivo.txt cargado

Figura 16

Vista desde la consola AWS



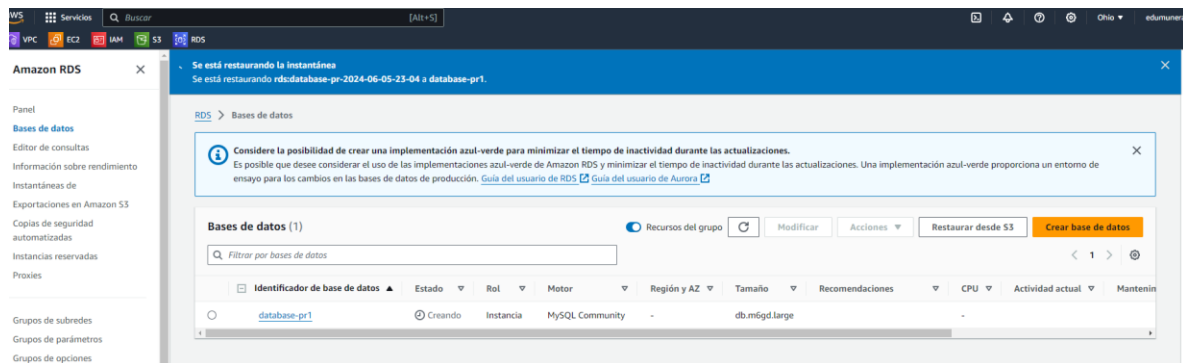
The screenshot displays the AWS Management Console interface for an Amazon S3 bucket named 'prbsemaws3'. The console shows the bucket's details, including its name, region, and type. A table lists the objects stored in the bucket:

Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
archivo.txt	txt	5 Jun 2024 1:51:05 PM -05	36.0 B	Estándar
LINEA_TIEMPO.jpg	jpg	5 Jun 2024 12:59:53 PM -05	107.6 KB	Estándar

Previamente en la plataforma de aws se creó el motor de base de datos con mysql basados en lo explicado en el seminario.

Figura 17

Creación de la Base de Datos



Creado el motor y estando funcionamiento o encendido, iniciamos sesión con la consola (recordemos que para el ejemplo usamos WSL) y estando conectados a la instancia en aws, ejecutamos el comando resaltado en rojo `mysql -h database-pr1.cb0s2agi6d0z.us-east-2.rds.amazonaws.com -u admin -p` el cual nos permite la conexión al motor de base de datos, esta solicitara una contraseña al cual fue previamente asignada al momento de crear el motor, y según la imagen podemos visualizar la conexión, realizamos un `show databases` con el fin de listar las bases de datos.

Figura 18

Conexión por consola a la Base de Datos

```
[root@ip-10-21-10-8 ec2-user]# mysql -h database-pr1.cb0s2agi6d0z.us-east-2.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 55
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.002 sec)

MySQL [(none)]>
```


Con el comando `create database`, creamos una base de datos y con `show databases` visualizamos nuestra nueva base de datos

Figura 19

Creando una Base de Datos por comando

```
MySQL [(none)]> CREATE DATABASE prawsem_db;
Query OK, 1 row affected (0.005 sec)

MySQL [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| prawsem_db |
| sys |
+-----+
5 rows in set (0.002 sec)

MySQL [(none)]> |
```

Creación de un servidor con servicio web y funcionamiento

A través de lo aprendido en clase, creamos unas instancias Windows Server, en la cual previamente le instalamos el servicio IIS que nos permita conectar y hacer pruebas locales de conexión al index por default de Windows

Figura 20

Instancias Windows Sever

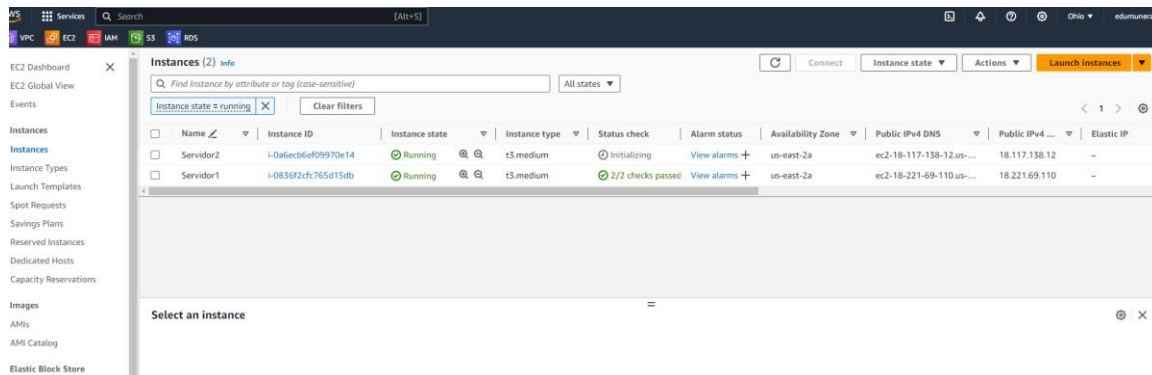
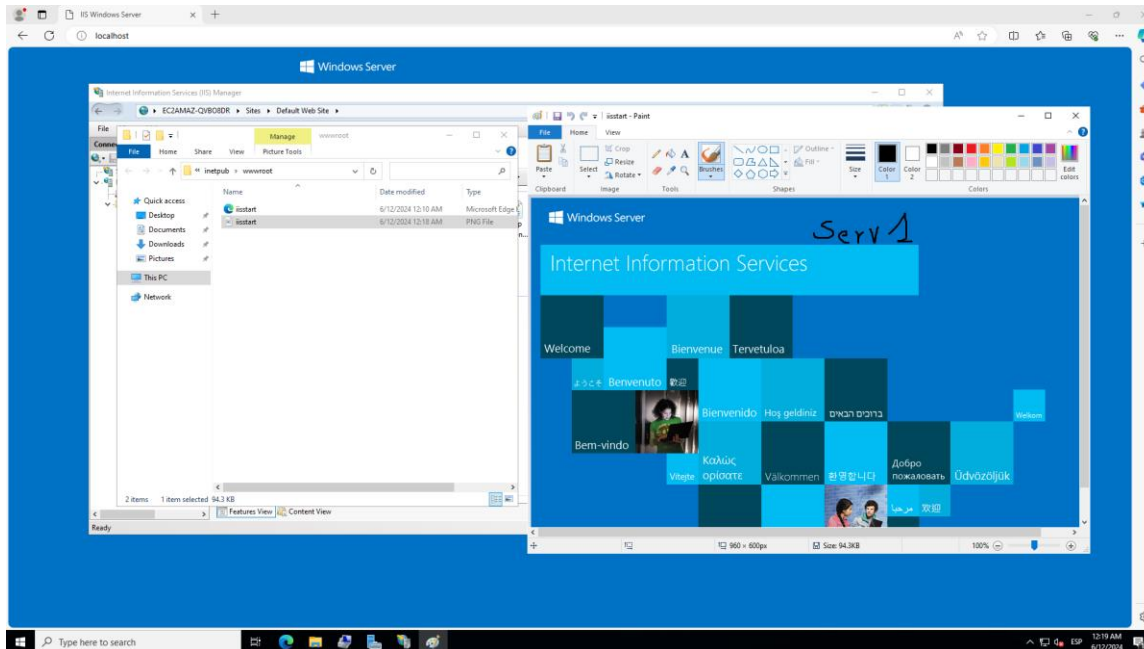


Figura 21

Modificando el index

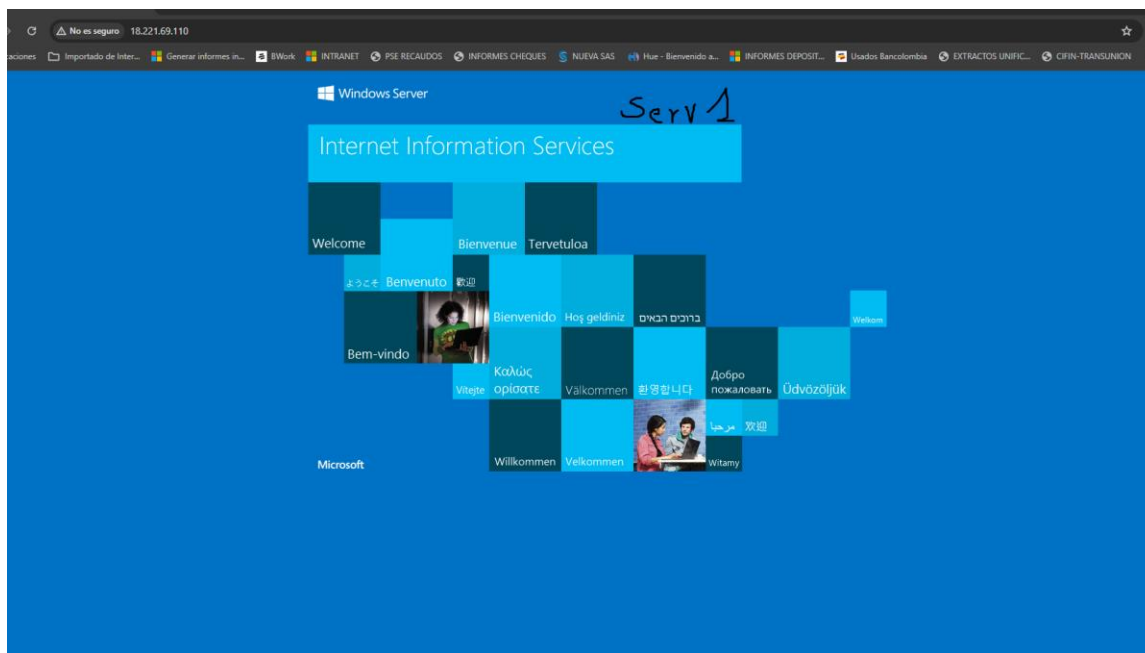


Nota. Ingresando IIS Manager, modificamos el index para diferenciar el servidor 1.

Desde nuestra maquina local accedemos a la IP publica de la instancia y visualizamos la pagina por default del Windows Server

Figura 22

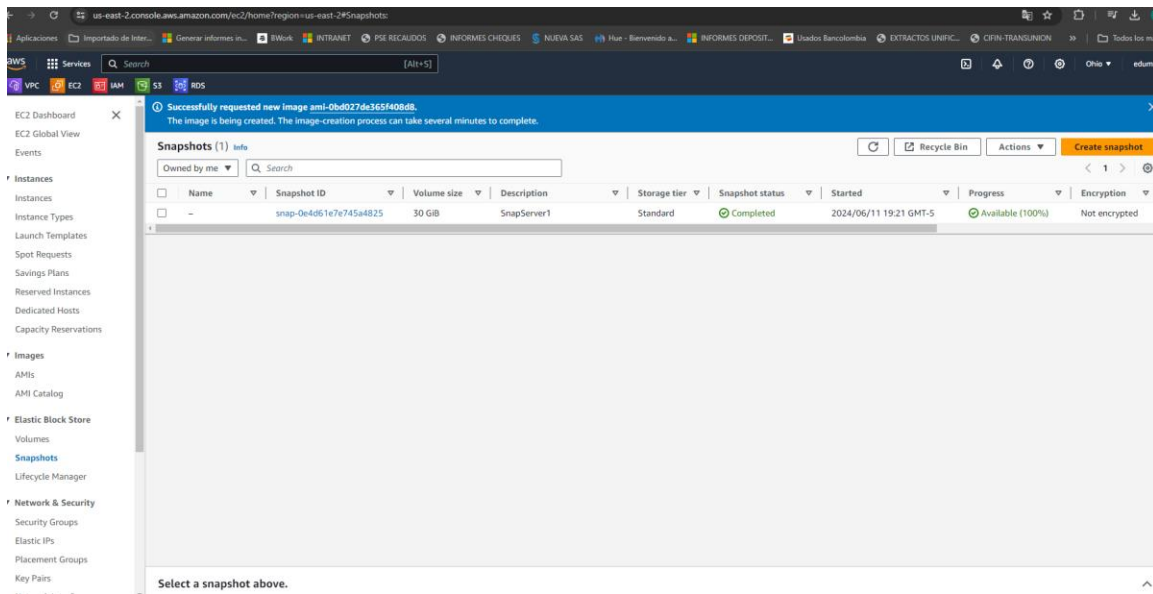
Prueba de funcionamiento



A través del servicio de snapshots creamos uno para nuestra imagen ya con las configuraciones de red, y el servicio IIS.

Figura 23

Visualización del snapshot



Basado en el Snapshot, generamos la AMI la cual la nombramos ImagenServ y montamos el servidor # 2 el cual al conectarnos nos permite visualizar localmente el index por default de Windows (previamente le realizamos una modificación a este index para diferenciar los servidores).

Figura 24
Creación de la AMI

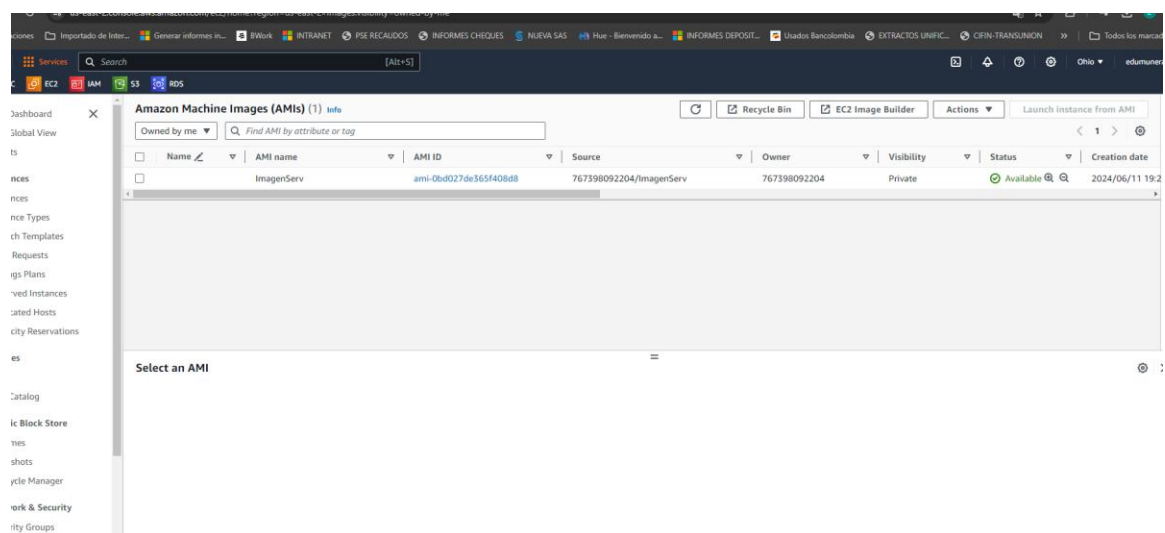


Figura 25

Inicializando el servidor 2

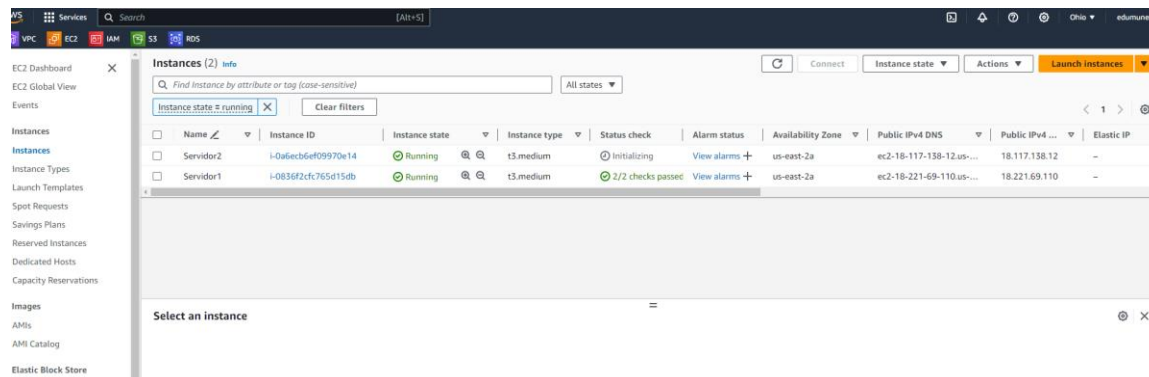
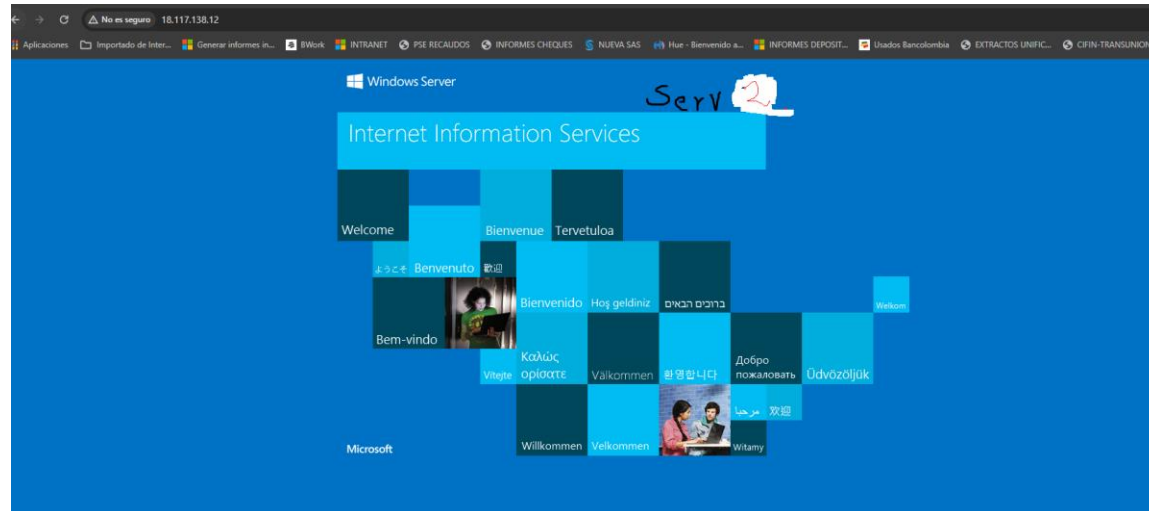


Figura 26

Pruebas del IIS del servidor 2



Nota. Previamente modificamos el index para diferenciar el servidor

Con los dos servidores en funcionamiento, creamos el balanceador de cargas, en el cual le asignamos un nombre, que sea internet-facing, seleccionamos el VPC, y definimos el puerto 80 para que reciba las peticiones entrantes por este puerto.

Figura 27
Visualización del balanceador

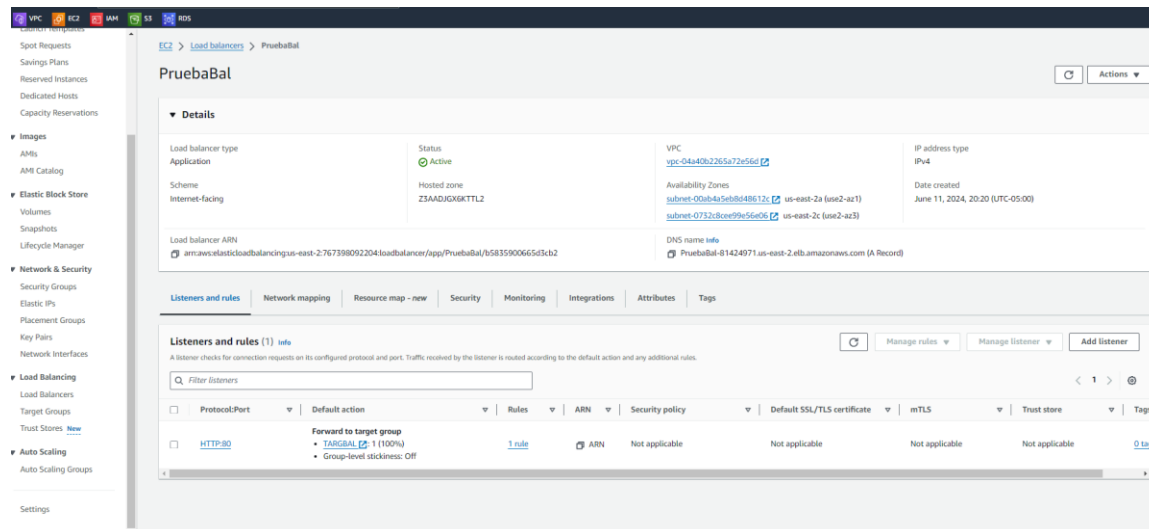


Figura 28

Configuraciones del balanceador

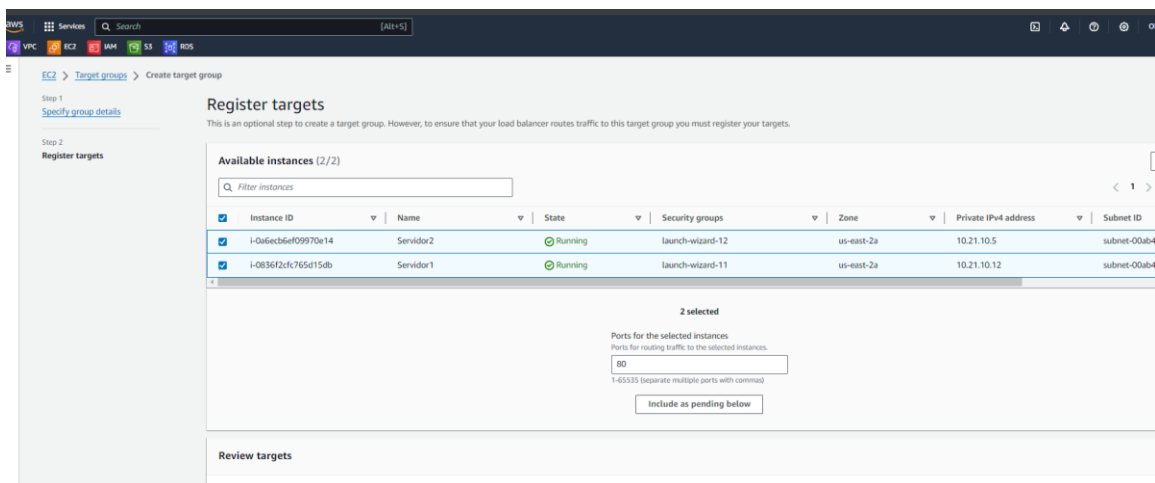
The screenshot displays the AWS Management Console interface for configuring a load balancer. The top navigation bar includes the AWS logo, a search bar, and a list of services: VPC, EC2, IAM, S3, and RDS. The main content area is divided into three sections:

- Subnets:** This section shows two IPv4 addresses assigned by AWS. The first is selected with a checked box and labeled "us-east-2b (use2-az2)". Below it, a "Subnet" dropdown menu is set to "subnet-03df3a0208b3a3a58" with a refresh button. The second IPv4 address is labeled "us-east-2c (use2-az3)" and is not selected.
- Security groups:** This section explains that a security group is a set of firewall rules. A dropdown menu labeled "Security groups" is set to "Select up to 5 security groups" with a refresh button. A search box below shows a result for "default" (sg-033dc11ee3a7c55c3) in VPC vpc-04a40b2265a72e56d.
- Listeners and routing:** This section defines a listener for "HTTP:80". The "Protocol" is set to "HTTP" and the "Port" is "80". The "Default action" is set to "Forward to" with a dropdown menu showing "Select a target group" and a refresh button. A red error message at the bottom states: "You must select at least one target group."

Se seleccionan las instancias requeridas por el balanceador para poder repartir las cargas entre las instancias.

Figura 29

Registrando los Targuets



The screenshot shows the AWS Management Console interface for the 'Create target group' wizard. The current step is 'Register targets'. A table lists two available instances, both of which are selected. Below the table, the number of selected instances is shown as '2 selected', and a text input field contains the port number '80'. The 'Include as pending below' button is visible.

Instance ID	Name	State	Security groups	Zone	Private IPv4 address	Subnet ID
<input checked="" type="checkbox"/> i-0a6ecb6ef09970e14	Servidor 2	Running	launch-wizard-12	us-east-2a	10.21.10.5	subnet-00ab4...
<input checked="" type="checkbox"/> i-0836f2cfc765d15db	Servidor 1	Running	launch-wizard-11	us-east-2a	10.21.10.12	subnet-00ab4...

2 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.

80

1-65535 (separate multiple ports with comma)

Include as pending below

Con la configuración del targuet y el balanceador nos indica que están en funcionamiento y con “buena salud” para su funcionamiento.

Figura 30

Funcionamiento de las instancias en el target

The screenshot displays the AWS Management Console for a Target Group named TARGBAL. The console shows the following details:

- Target type:** Instance
- Protocol:** HTTP, **Port:** 80
- Protocol version:** HTTP1
- VPC:** vpc-04a42b2265a73e56d
- IP address type:** IPv4
- Load balancer:** Prunbalbal

The health status summary indicates:

- 2 Total targets
- 2 Healthy
- 0 Unhealthy
- 0 Unused
- 0 Initial
- 0 Draining
- 0 Anomalous

The **Distribution of targets by Availability Zone (AZ)** section shows the following registered targets:

Instance ID	Name	Port	Zone	Health status	Health status details	Launch...	Anomaly detection result
i-0a6eb6ef09970e14	Server2	80	us-east-2a	Healthy	-	June 11, 2...	Normal
i-0836f2cfc765d15db	Server1	80	us-east-2a	Healthy	-	June 11, 2...	Normal

Realizamos pruebas en nuestro windows local con el nombre suministrado por la plataforma y se visualizan el balanceo de cargas

Figura 31

Prueba por el navegador Edge

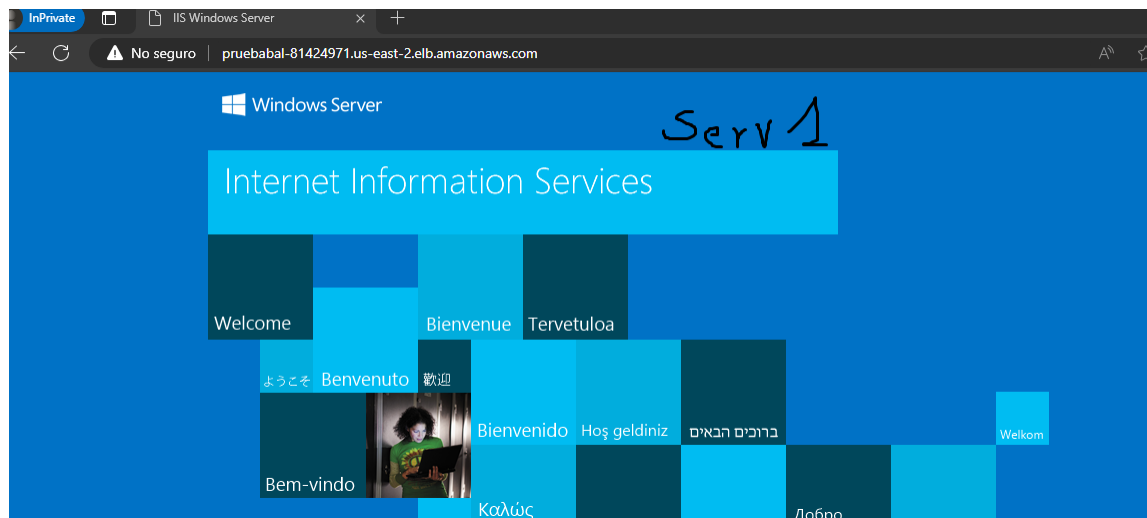
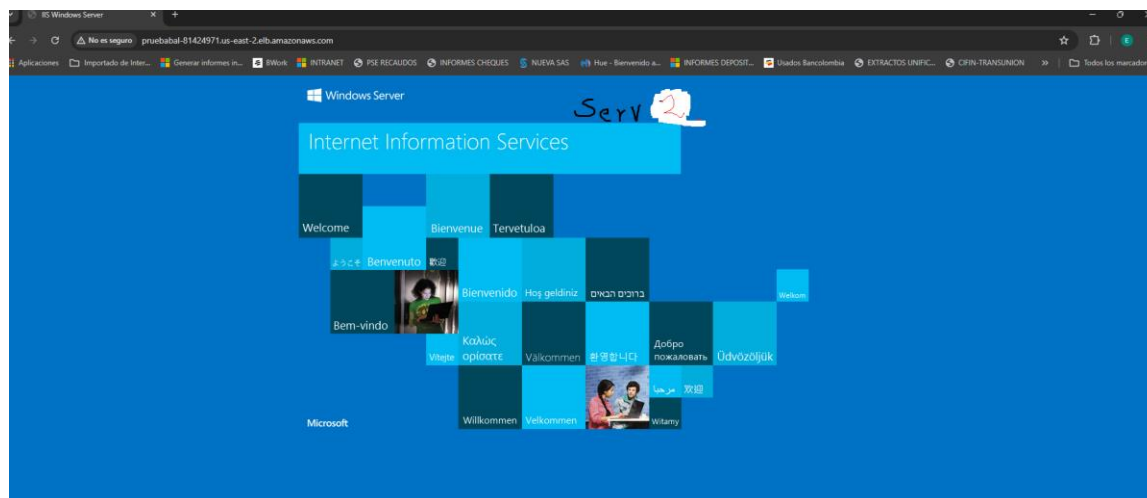


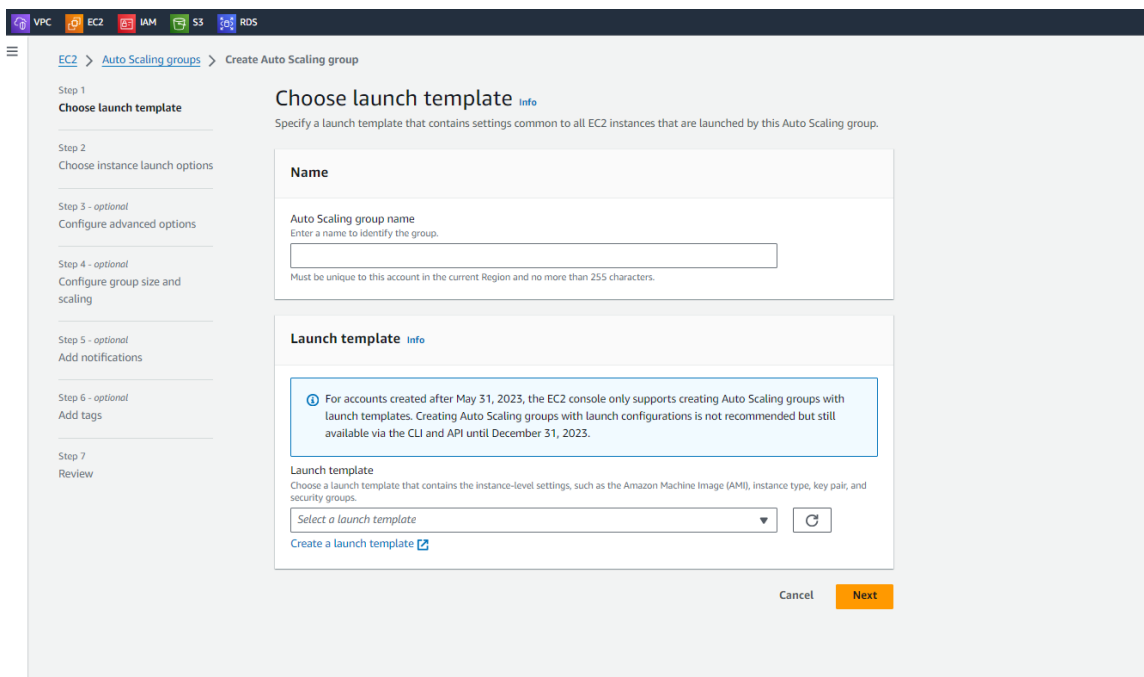
Figura 32

Prueba por el navegador Chrome



Para crear el auto Scaling Group, ingresamos por la opción con este mismo nombre, recuerda ubicarte en EC2 para visualizarla, nos solicitara un nombre un template, en este caso si no se ha creado previamente, se da clic en “create a launch template”.

Figura 33
Creación de Scaling Group



En el templete, asignamos un nombre, seleccionamos la imagen del sistema operativo, que en este caso seleccionaremos la opción my aMIs la cual creamos previamente, nos solicitara que tipo de maquina deseamos montar esta AMI, un grupo de seguridad (seleccionar uno que tenga habilitado el puerto 80 para la salida a internet, puede ser el mismo configurado al inicio de la instancia del windows server).

Figura 34

Selección de las opciones del template

▼ **Application and OS Images (Amazon Machine Image) - required** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

🔍 Search our full catalog including 1000s of application and OS images

Recents | **My AMIs** | Quick Start

Owned by me Shared with me

[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Imagenserv2
ami-029fc3c4753a5efe1
2024-06-12T22:59:42.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

Description
-

Architecture	AMI ID
x86_64	ami-029fc3c4753a5efe1

Creado el template, lo seleccionamos en la opción del launch template y continuamos con el proceso.

Figura 35

Continuación del Scaling Group

The screenshot shows the AWS Management Console interface for creating an Auto Scaling group. The page title is "Choose launch template" with an "Info" link. Below the title, there is a sub-header "Name" and a text input field for the "Auto Scaling group name" containing the text "PRSCALING". A note below the field states: "Must be unique to this account in the current Region and no more than 255 characters." Below this is the "Launch template" section, which includes a blue information box with a warning icon and text: "For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023." Underneath, there is a "Launch template" dropdown menu set to "PRTEMPLATE" and a refresh button. Below that is a "Version" dropdown menu set to "Default (1)" with another refresh button. At the bottom, there is a table of details for the selected launch template:

Description	Launch template PRTEMPLATE lt-0166858391d79f234	Instance type t2.medium
AMI ID ami-0bd027de365f408d8	Security groups -	Request Spot Instances No

El siguiente paso es escoger el VPC y las zonas donde se crearán las instancias

Figura 36

Continuación de la configuración de Scaling Group

Choose instance launch options [Info](#)

Choose the VPC network environment that your instances are launched into, and customize the instance types and purchase options.

Instance type requirements [Info](#)

You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

[Override launch template](#)

Launch template	Version	Description
PRTEMPLATE ↗ lt-0166858391d79f234	Default	-

Instance type
t2.medium

Network [Info](#)

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-04a40b2265a72e56d (SemAWSed-vpc) [↕](#) [↻](#)
10.21.10.0/24

[Create a VPC \[↗\]\(#\)](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets [↕](#) [↻](#)

[Create a subnet \[↗\]\(#\)](#)

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

Seleccionamos la opción de adjuntar un balanceador, y en select target groups, escogemos el balanceador previamente creado.

Figura 37

Seleccionando un Balanceador

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer
 Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer
 Choose from your existing load balancers.

Attach to a new load balancer
 Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups
 This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups ▼ ↻

VPC Lattice integration options [info](#)

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

Select VPC Lattice service to attach

No VPC Lattice service
 VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

Attach to VPC Lattice service
 Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

[Create new VPC Lattice service](#)

Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks
 Always enabled

Additional health check types - *optional* | [info](#)

Turn on Elastic Load Balancing health checks Recommended

Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Seleccionamos la capacidad deseada de instancias el mínimo y máximo de instancias.

Figura 38

Continuación de las configuraciones

Tipo de capacidad deseado
Elija la unidad de medida para el valor de capacidad deseado. Las vCPU y la memoria (GiB) solo son compatibles con grupos de instancias mixtos configurados con un conjunto de atributos de instancia.

Unidades (número de instancias) ▼

Capacidad deseada
Especifique el tamaño de su grupo.

3

Escalado [Info](#)
Puede cambiar el tamaño de su grupo de escalamiento automático de forma manual o automática para cumplir con los cambios en la demanda.

Límites de escalamiento
Establezca límites sobre cuánto puede aumentarse o disminuirse la capacidad deseada.

Capacidad deseada mínima	Capacidad deseada máxima
2	6
Capacidad igual o inferior a la deseada	Capacidad igual o superior a la deseada

Escalamiento automático - *opcional*
Elija si desea utilizar una política de seguimiento de destino [Info](#)
Puede configurar otras políticas de escalado basadas en métricas y un escalado programado después de crear su grupo de escalamiento automático.

Sin políticas de escalamiento
Su grupo de escalamiento automático mantendrá su tamaño inicial y no se redimensionará de forma dinámica para satisfacer la demanda.

Política de escalado de seguimiento de destino
Elija una métrica y un valor objetivo de CloudWatch y deje que la política de escalamiento ajuste la capacidad deseada en proporción al valor de la métrica.

Nombre de la política de escalado

SCALINGPOL

Tipo de métrica [Info](#)
Métrica supervisada que determina si la utilización de recursos es demasiado baja o alta. Si utiliza métricas de EC2, considere la posibilidad de habilitar la supervisión detallada para obtener un mejor rendimiento de escalado.

Creado el grupo de auto Scaling, ingresamos por la opción de automatic scaling, y crearemos dos políticas dinámicas, en esta opción creamos las alertas una para aumentar la cantidad de instancias cuando supere en nuestro ejemplo el 60 % y termine instancias cuando este por debajo del 25 %

Figura 39

Creando políticas del Auto Scaling

The screenshot displays the AWS Auto Scaling console for a group named 'PRSCALING'. The interface includes a navigation bar with tabs for 'Details', 'Activity', 'Automatic scaling', 'Instance management', 'Monitoring', and 'Instance refresh'. A blue informational banner at the top explains that scaling policies help meet demand changes by tracking CloudWatch metrics and taking action when alarms are triggered. Below this, the 'Dynamic scaling policies' section is active, showing two policies: 'CREAR' and 'QUIT'. The 'CREAR' policy is a step scaling policy that triggers when CPU utilization reaches 60% for 1 consecutive period of 300 seconds, adding 1 capacity unit. The 'QUIT' policy is a step scaling policy that triggers when CPU utilization reaches 25% for 1 consecutive period of 300 seconds, removing 1 capacity unit. At the bottom, the 'Predictive scaling policies' section is visible, showing an evaluation period of 2 days and a table with columns for Name, Metric pair, Forecast and scale, Recommendation, Chart, Availability impact, and Cost impact.

Figura 40

Creando las alarmas con cloud Watch

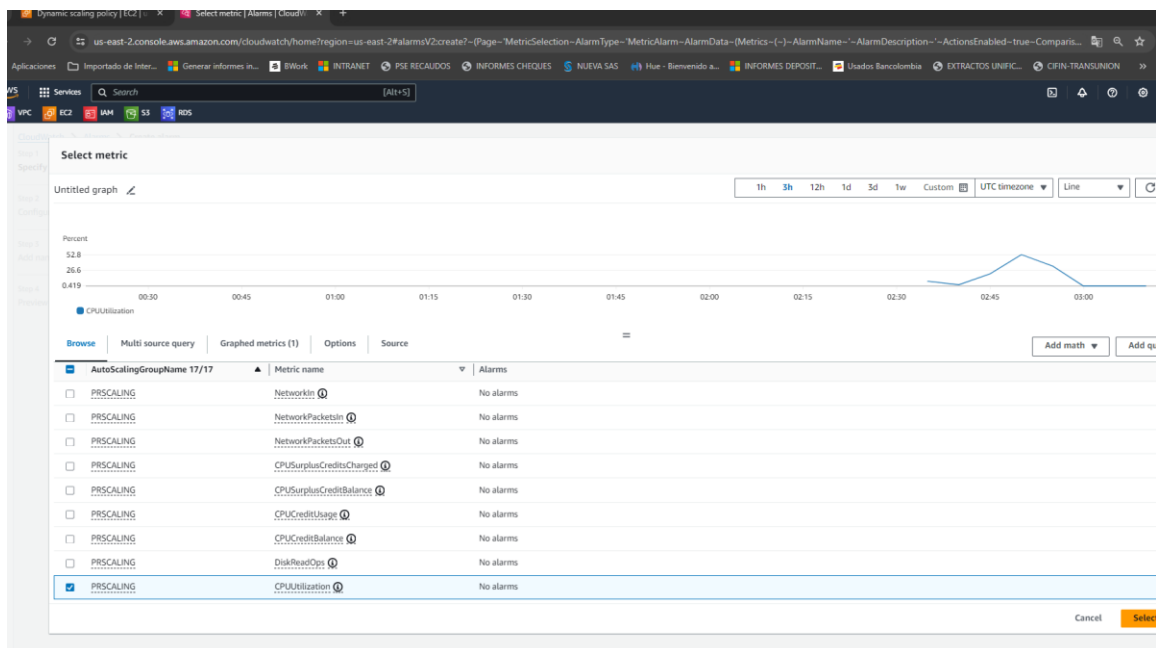
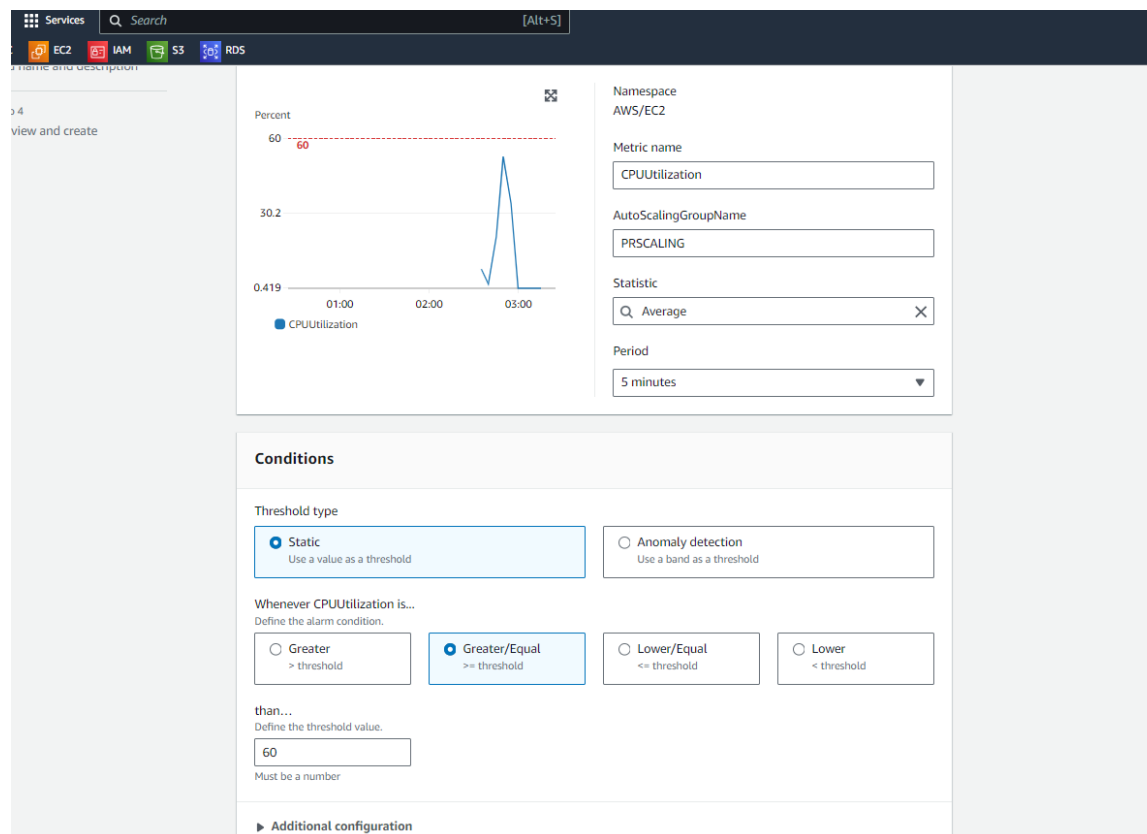
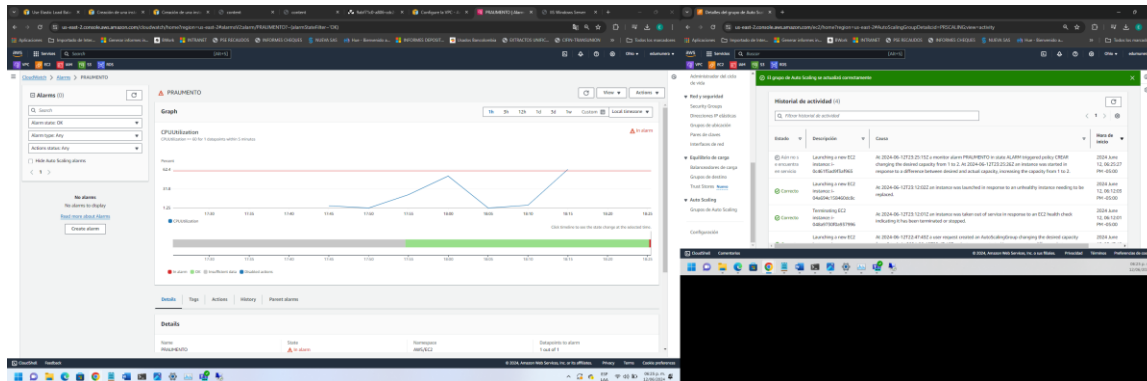


Figura 41*Configuración de la Alarma*

Al realizar prueba de estrés a una instancia visualizamos en la imagen izquierda que al empezar a sobrepasar el 60 % inicia la creación de una instancia visualizándose en el lado derecho de la imagen.

Figura 42

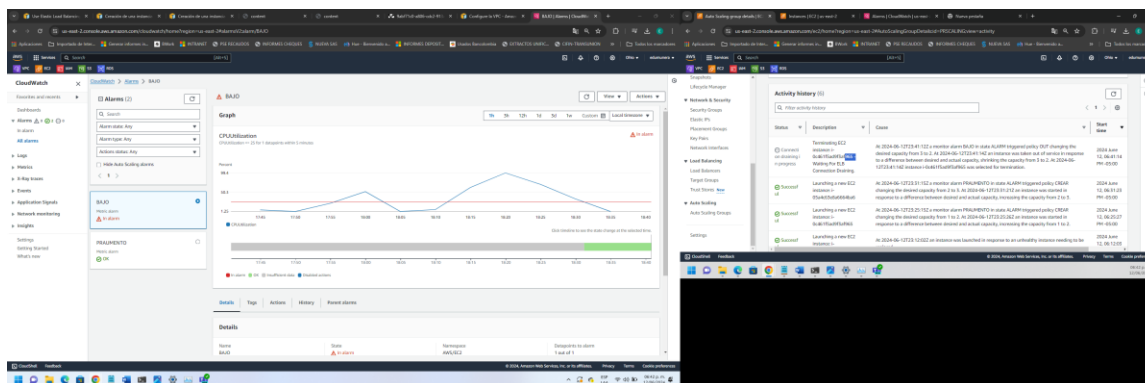
Funcionamiento de creación de instancias



Y cuando la instancia no esta en uso y disminuye su uso sobre el 25% se visualiza que elimina instancias.

Figura 43

Funcionamiento de eliminación de instancias



Conclusiones

En conclusión, con el trabajo realizado se ha demostrado cómo se pueden aplicar los conceptos básicos aprendidos durante el seminario de Amazon Web Services. La implementación de servicios en la nube, tales como la configuración de instancias EC2, creación de Snapshots y AMIs, la creación y gestión de bases de datos con RDS, establecer políticas de escalado automático, permite visualizar la flexibilidad y eficiencia ofrecidas para las infraestructuras empresariales.

Con estos ejercicios se puede evidenciar la importancia de la computación en la nube, sus ventajas y proporcionan un conocimiento de la plataforma y así mejorar la comprensión de las capacidades ofrecidas por AWS

Referencias

Amazon Web Services. (2023). Amazon Web Services. Recuperado el 19 de mayo de 2024, de <https://aws.amazon.com/es/>

Amazon Web Services. (s.f.). Use an Elastic Load Balancer. Amazon Web Services. Recuperado el 19 de mayo de 2024, de <https://docs.aws.amazon.com/autoscaling/ec2/userguide/autoscaling-load-balancer.html>

Amazon Web Services. (s.f.). Amazon RDS: Guía del usuario. Recuperado el 12 de junio de 2024, de https://docs.aws.amazon.com/es_es/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.CreatingConnecting.MySQL.html

Amazon Web Services. (s.f.). Amazon RDS: Guía del usuario. Recuperado el 12 de junio de 2024, de https://docs.aws.amazon.com/es_es/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.CreatingConnecting.MySQL.html#CHAP_GettingStarted.Connecting.MySQL

Amazon Web Services. (s.f.). Amazon Virtual Private Cloud (VPC): Guía del usuario.

Recuperado el 12 de junio de 2024, de

https://docs.aws.amazon.com/es_es/vpc/latest/userguide/modify-vpcs.html

Google. (n.d.). Productos de Google Cloud. Google Cloud. Recuperado el 19 de mayo de 2024,

de https://cloud.google.com/products?hl=es_419

Google Cloud. (2023). Comparación de servicios entre AWS, Azure y GCP. Recuperado el 19 de

mayo de 2024, de <https://cloud.google.com/docs/get-started/aws-azure-gcp-service-comparison?hl=es-419>

IBM. (2012). CP-40 and the origins of VM/370. IBM. Recuperado el 19 de mayo de 2024, de

<https://www.vm.ibm.com/history/vm40hist.pdf>

IBM. (2024). Virtual machines. IBM. Recuperado el 19 de mayo de 2024, de

<https://www.ibm.com/topics/virtual-machines>

IBM. (n.d.). IBM System/370. IBM. Recuperado el 19 de mayo de 2024, de

<https://www.ibm.com/history/system-370>

Mainframe Cloud. (2024). Origen de la computación en la nube. Recuperado el 19 de mayo de 2024, de <https://mainframecloud.co/origen-computacion-en-la-nube>

Microsoft. (2008, octubre 27). Microsoft unveils Windows Azure at Professional Developers Conference. Microsoft News. Recuperado el 19 de mayo de 2024, de <https://news.microsoft.com/2008/10/27/microsoft-unveils-windows-azure-at-professional-developers-conference/>

Microsoft. (2023). Servicios profesionales de AWS en Azure. Microsoft Learn. Recuperado el 19 de mayo de 2024, de <https://learn.microsoft.com/es-es/azure/architecture/aws-professional/services>

Nuvalab. (2021, 8 de junio). Auto Scaling Group AWS (En Español). [Video de YouTube]. Recuperado el 19 de mayo de 2024, de <https://www.youtube.com/watch?v=4z7z3z7z3z7>

Tezén, D. (2022, mayo 19). Transferir archivos entre EC2 y S3 en AWS. Dennys Tezén. [Video de YouTube]. Recuperado el 19 de mayo de 2024, de https://www.youtube.com/watch?v=z_232022