

<b>Nombre de la práctica</b>	Apache (httpd) en aws			<b>No.</b>	<b>5</b>
<b>Asignatura:</b>	<b>Administración de Redes</b>	<b>Carrera:</b>	<b>INGENIERÍA EN SISTEMAS COMPUTACIONALES</b>	<b>Duración de la práctica (Hrs)</b>	<b>2 horas</b>

**GRUPO: 3601**

**NOMBRE: Vanesa Hernández Martínez**

## Encuadre con CACEI

No. atributo	Atributos de egreso del PE que impactan en la asignatura	Criterio de desempeño	Indicadores	
A2	El estudiante diseñará esquemas de trabajo y procesos, usando metodologías congruentes en la resolución de problemas de ingeniería en sistemas computacionales	CD1. IDENTIFICA METODOLOGÍAS Y PROCESOS EMPLEADOS EN LA RESOLUCIÓN DE PROBLEMAS	I1	IDENTIFICACION Y RECONOCIMIENTO DE DISTINTAS METODOLOGIAS PARA LA RESOLUCION DE PROBLEMAS
			I2	MANEJO DE PROCESOS ESPECIFICOS EN LA SOLUCION DE PROBLEMAS Y/O DETECCION DE NECESIDADES
		CD2 DISEÑA SOLUCIONES A PROBLEMAS, EMPLEANDO METODOLOGÍAS APROPIADAS AL AREA	I1	USO DE METODOLOGIAS PARA EL MODELADO DE LA SOLUCION DE SISTEMAS Y APLICACIONES
A7	El estudiante desarrolla proyectos y trabajos en equipo basándose en metodologías preestablecidas para lograr mayor calidad y eficiencia.	CD2. ASUME SU RESPONSABILIDAD EN EL DESARROLLO DE TRABAJOS Y/O PROYECTOS EN EQUIPO Y EN LA ENTREGA DE RESULTADOS	I1	PARTICIPACIÓN ACTIVA EN EL DESARROLLO DE TRABAJOS Y PROYECTOS EN EQUIPO
			I2	DIRIGIR Y ORGANIZAR TRABAJO EN EQUIPO
			I3	PRESENTACION Y/O EXPOSICION DE TRABAJOS Y PROYECTOS EN EQUIPO

1. Una vez dentro de la terminal de nuestra instancia, instalamos nmap con el comando:

```
sudo yum -y install nmap
```

2. Revisamos los puertos lógicos abiertos con el comando:

```
sudo nmap localhost
```

```
[ec2-user@ip-172-31-12-60 ~]$ sudo nmap localhost
Starting Nmap 7.93 ( https://nmap.org ) at 2025-03-25 19:28 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000020s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
```

3. Instalamos httpd mediante el comando:

```
sudo yum -y install httpd
```

```
[ec2-user@ip-172-31-12-60 ~]$ sudo yum -y install httpd
Last metadata expiration check: 0:24:04 ago on Tue Mar 25 19:10:37 2025.
Dependencies resolved.
=====
Package                               Architecture      Version           Repository        Size
=====
Installing:
httpd                                  x86_64            2.4.62-1.amzn2023    amazonlinux        48 k
Installing dependencies:
apr                                    x86_64            1.7.5-1.amzn2023.0.4    amazonlinux        129 k
apr-util                              x86_64            1.6.3-1.amzn2023.0.1    amazonlinux        98 k
generic-logos-httpd                  noarch            18.0.0-12.amzn2023.0.3    amazonlinux        19 k
httpd-core                            x86_64            2.4.62-1.amzn2023    amazonlinux        1.4 M
httpd-filesystem                     noarch            2.4.62-1.amzn2023    amazonlinux        14 k
httpd-tools                           x86_64            2.4.62-1.amzn2023    amazonlinux        81 k
libbrotli                             x86_64            1.0.9-4.amzn2023.0.2    amazonlinux        315 k
mailcap                               noarch            2.1.49-3.amzn2023.0.3    amazonlinux        33 k
Installing weak dependencies:
apr-util-openssl                     x86_64            1.6.3-1.amzn2023.0.1    amazonlinux        17 k
mod_http2                             x86_64            2.0.27-1.amzn2023.0.3    amazonlinux        166 k
mod_lua                               x86_64            2.4.62-1.amzn2023    amazonlinux        61 k
=====
Transaction Summary
=====
Install 12 Packages

Total download size: 2.3 M
Installed size: 6.9 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm           433 kB/s | 17 kB  00:00
(2/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm                 1.9 MB/s | 98 kB  00:00
(3/12): apr-1.7.5-1.amzn2023.0.4.x86_64.rpm                       2.2 MB/s | 129 kB  00:00
(4/12): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch.rpm     887 kB/s | 19 kB  00:00
(5/12): httpd-2.4.62-1.amzn2023.x86_64.rpm                       2.4 MB/s | 48 kB  00:00
(6/12): httpd-filesystem-2.4.62-1.amzn2023.noarch.rpm             462 kB/s | 14 kB  00:00
(7/12): httpd-core-2.4.62-1.amzn2023.x86_64.rpm                  31 MB/s | 1.4 MB  00:00
(8/12): httpd-tools-2.4.62-1.amzn2023.x86_64.rpm                 2.2 MB/s | 81 kB  00:00
(9/12): mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm                  1.5 MB/s | 33 kB  00:00
(10/12): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm               6.0 MB/s | 315 kB  00:00
(11/12): mod_http2-2.0.27-1.amzn2023.0.3.x86_64.rpm              2.9 MB/s | 166 kB  00:00
(12/12): mod_lua-2.4.62-1.amzn2023.x86_64.rpm                     1.4 MB/s | 61 kB  00:00
=====
Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64      apr-util-1.6.3-1.amzn2023.0.1.x86_64      apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch      httpd-2.4.62-1.amzn2023.x86_64      httpd-core-2.4.62-1.amzn2023.x86_64
httpd-filesystem-2.4.62-1.amzn2023.noarch      httpd-tools-2.4.62-1.amzn2023.x86_64      libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch      mod_http2-2.0.27-1.amzn2023.0.3.x86_64      mod_lua-2.4.62-1.amzn2023.x86_64

Complete!
[ec2-user@ip-172-31-12-60 ~]$
```

4. Una vez completada la instalación, inicializaremos el servicio con el comando:

**sudo systemctl start httpd**

```
[ec2-user@ip-172-31-12-60 ~]$ sudo systemctl start httpd
```

5. El comando anterior no mostrara nada, pero verificaremos que el estado del servicio este corriendo mediante el comando:

**sudo systemctl status httpd**

```
[ec2-user@ip-172-31-12-60 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Tue 2025-03-25 19:42:52 UTC; 29s ago
     Docs: man:httpd.service(8)
  Main PID: 26681 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
     Tasks: 177 (limit: 1111)
    Memory: 12.9M
       CPU: 67ms
    CGroup: /system.slice/httpd.service
            └─26681 /usr/sbin/httpd -DFOREGROUND
              └─26703 /usr/sbin/httpd -DFOREGROUND
                └─26704 /usr/sbin/httpd -DFOREGROUND
                  └─26705 /usr/sbin/httpd -DFOREGROUND
                    └─26706 /usr/sbin/httpd -DFOREGROUND

Mar 25 19:42:52 ip-172-31-12-60.us-east-2.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Mar 25 19:42:52 ip-172-31-12-60.us-east-2.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Mar 25 19:42:52 ip-172-31-12-60.us-east-2.compute.internal httpd[26681]: Server configured, listening on: port 80
[ec2-user@ip-172-31-12-60 ~]$
```

Nota: Para reiniciar se utiliza: **sudo systemctl restart httpd** y para detener: **sudo systemctl stop httpd**.

6. Verificamos que se haya abierto el puerto 80 que le corresponde a httpd

```
[ec2-user@ip-172-31-12-60 ~]$ sudo nmap localhost
Starting Nmap 7.93 ( https://nmap.org ) at 2025-03-25 19:48 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000020s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.13 seconds
[ec2-user@ip-172-31-12-60 ~]$
```

7. Revisamos cual es la ruta en a cual nos encontramos mediante el comando pwd:

```
[ec2-user@ip-172-31-12-60 ~]$ pwd
/home/ec2-user
```

8. Nos cambiamos a la ruta donde se guardan las paginas mediante el comando: `cd /var/www/html`  
Y verificamos con el comando `pwd` que si nos hayamos movido a esa ruta.

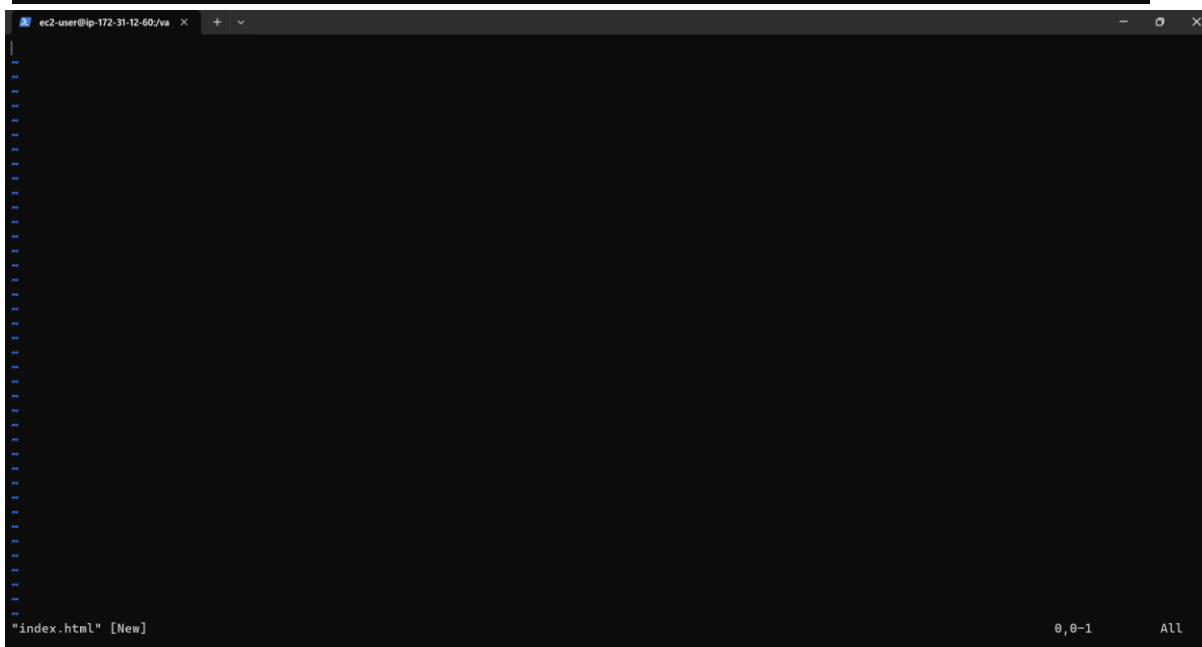
```
[ec2-user@ip-172-31-12-60 ~]$ cd /var/www/html  
[ec2-user@ip-172-31-12-60 html]$ pwd  
/var/www/html  
[ec2-user@ip-172-31-12-60 html]$ |
```

9. Verificamos que no haya nada dentro con el comando `ls`

```
[ec2-user@ip-172-31-12-60 html]$ ls
```

10. Posteriormente abrimos un archivo `index.html` con `vi` mejorado mediante el comando:

```
[ec2-user@ip-172-31-12-60 html]$ sudo vi index.html
```



11. Para comenzar a escribir se pulsa la tecla `i` y escribimos lo siguiente :

```
<html>  
    <head>  
    </head>  
  
    <body>  
        <h1>TECNOLOGICO DE ESTUDIOS SUPERIORES DE JILOTEPEC </h1>  
    </body>  
</html>
```

12. Verificamos que el archivo fue creado correctamente con el siguiente comando: **ls -la** y que no se encuentre ningún archivo .soap que pueda provocar algún error, en dado caso de que existiera se elimina con el comando **sudo rm index.html.swp**

```
[ec2-user@ip-172-31-12-60 html]$ ls -la
total 4
drwxr-xr-x. 2 root root 24 Mar 25 20:03 .
drwxr-xr-x. 4 root root 33 Mar 25 19:34 ..
-rw-r--r--. 1 root root 109 Mar 25 20:03 index.html
[ec2-user@ip-172-31-12-60 html]$
```

## ¿Cómo acceder a la página desde el navegador?

13. Entre a mi panel de aws, en donde se encuentra la instancia creada, en la parte inferior vienen algunos apartados, nos dirigimos la apartado de seguridad

The screenshot shows the AWS Management Console interface. On the left, the navigation pane includes sections like 'EC2', 'Imágenes', 'Elastic Block Store', and 'Red y seguridad'. The main content area displays the 'Instancias (1/1)' page. The 'Seguridad' tab is selected and highlighted with a red box. Under 'Detalles de seguridad', the 'Grupos de seguridad' section is also highlighted with a red box, showing a single security group: 'sg-0e31affc49eb1b730 (launch-wizard-1)'. Other details visible include the IAM role, owner ID, and launch time.

14. Encontramos el grupo de seguridad de nuestra instancia y damos doble clic sobre el nombre de este:

This is a close-up of the 'Seguridad' tab from the previous screenshot. It shows the 'Detalles de seguridad' section. The 'Grupos de seguridad' subsection is highlighted with a red box, displaying the security group 'sg-0e31affc49eb1b730 (launch-wizard-1)'. To the right, the 'ID del propietario' (248189928099) and 'Hora de lanzamiento' (Tue Mar 25 2025 13:22:01 GMT-0600) are visible.

15. Una vez dentro se podrán visualizar todas las reglas que tenemos de momento, la cual solo es la de ssh, y damos clic en el botón de “editar reglas de entrada”.

16. Nos abrirá el siguiente apartado en donde podremos agregar una nueva regla dando clic en el botón de “Agregar regla”.

Editar reglas de entrada Información

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

ID de la regla del grupo de seguridad	Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional	
sg-090468a2ca6c4c24e	SSH	TCP	22	Person...	0.0.0.0/0	Eliminar

**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

17. Dentro de esta nueva regla configuraremos:

- Tipo: **HTTP**
- Protocolo: **TCP**
- Puerto: **80**
- Origen: **0.0.0.0/0** (para acceso desde cualquier IP)

Una vez listas estas especificaciones daremos clic en guardar regla

**Editar reglas de entrada** Información

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

ID de la regla del grupo de seguridad	Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional	Acciones
sgr-090468a2ca6c4c24e	SSH	TCP	22	Person...		Eliminar
-	HTTP	TCP	80	Anywh...	0.0.0.0/0	Eliminar

[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](#)

18. Si la regla se creo correctamente en esta parte en este apartado ya nos aparecerán todas las reglas que tenemos.

**Reglas de entrada (2)**

<input type="checkbox"/>	Name	ID de la regla del gr...	Versión de IP	Tipo	Protocolo	Intervalo de puertos	Origen
<input type="checkbox"/>	-	sgr-0c93d15e7444b78a4	IPv4	HTTP	TCP	80	0.0.0.0/0
<input type="checkbox"/>	-	sgr-090468a2ca6c4c24e	IPv4	SSH	TCP	22	0.0.0.0/0

19. En nuestro navegador agregamos: [http://TU\\_IP\\_PUBLICA](http://TU_IP_PUBLICA)

En mi caso es: <http://3.15.5.134/> Y listo ya podremos visualizar la página.



TECNOLOGIC DE ESTUDIOS SUPERIORES DE JILOTEPEC

20. Si deseáramos cambiar el contenido de dicha página, regresamos a nuestra terminal y volvemos a abrir el archivo:

```
[ec2-user@ip-172-31-12-60 html]$ sudo vi index.html|
```

21. Procedemos a ingresar más código al respecto

```
margin: 20px auto;
background: white;
padding: 20px;
border-radius: 10px;
box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
}
img {
width: 100%;
max-width: 600px;
border-radius: 10px;
}
a {
display: inline-block;
margin-top: 10px;
padding: 10px 20px;
background-color: #007bff;
color: white;
text-decoration: none;
border-radius: 5px;
}
a:hover {
background-color: #0056b3;
}
</style>
</head>
<body>
<header>Ingeniería en Sistemas Computacionales</header>
<div class="container">
<p>La Ingeniería en Sistemas Computacionales es una disciplina que combina el desarrollo de software, administración de redes y gestión de bases de datos para crear soluciones tecnológicas eficientes.</p>
<p>Los ingenieros en sistemas computacionales pueden desempeñarse en áreas como el desarrollo de aplicaciones, ciberseguridad, inteligencia artificial, análisis de datos y administración de infraestructura tecnológica.</p>
<p>Además, esta carrera brinda conocimientos en lenguajes de programación, arquitectura de hardware, optimización de sistemas y gestión de proyectos tecnológicos.</p>

</div>
</body>
</html>
"index.html" 59L, 2156B
```

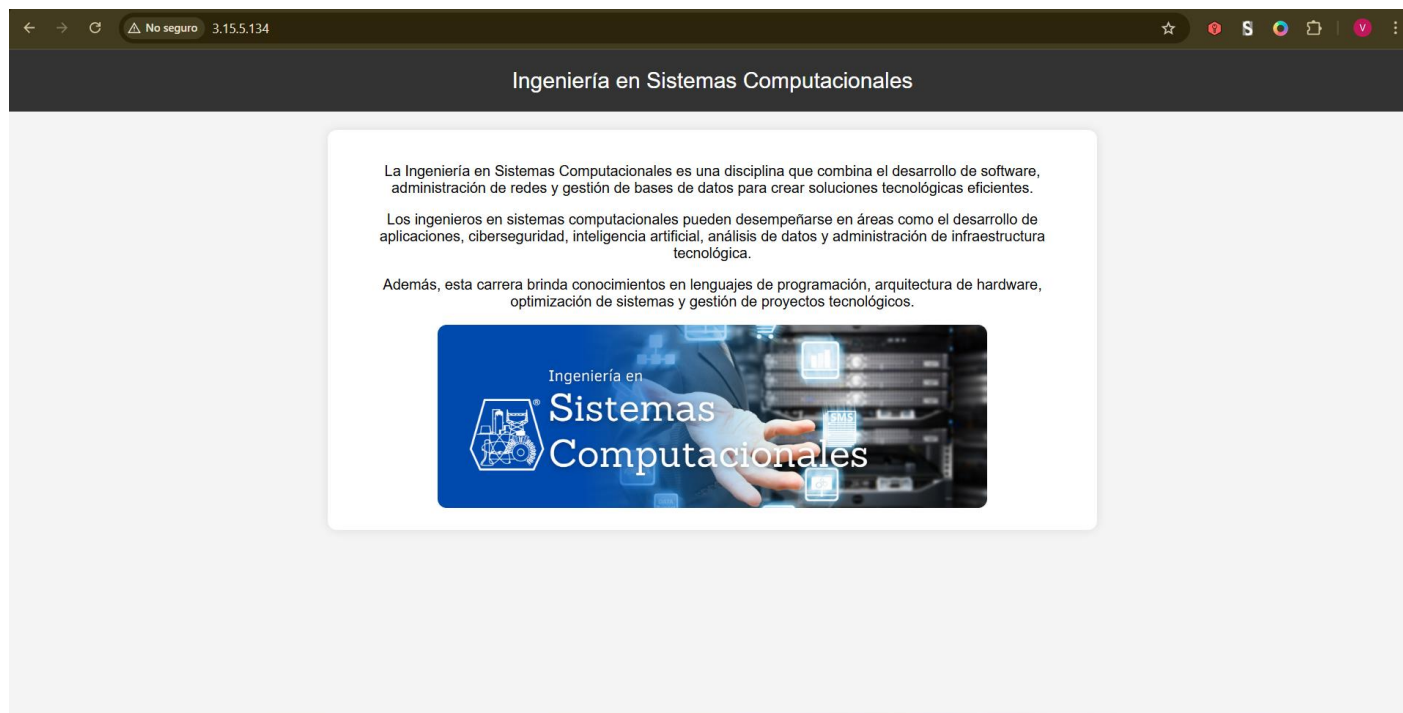


22. Reseteamos el servicio, después lo volvemos a iniciar y verificamos su status.

```
[ec2-user@ip-172-31-12-60 html]$ sudo systemctl restart httpd
[ec2-user@ip-172-31-12-60 html]$ sudo systemctl start httpd
[ec2-user@ip-172-31-12-60 html]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Tue 2025-03-25 20:46:04 UTC; 16s ago
     Docs: man:httpd.service(8)
  Main PID: 30292 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1111)
   Memory: 12.9M
      CPU: 62ms
   CGroup: /system.slice/httpd.service
           └─30292 /usr/sbin/httpd -DFOREGROUND
             └─30294 /usr/sbin/httpd -DFOREGROUND
               └─30295 /usr/sbin/httpd -DFOREGROUND
                 └─30296 /usr/sbin/httpd -DFOREGROUND
                   └─30297 /usr/sbin/httpd -DFOREGROUND

Mar 25 20:46:04 ip-172-31-12-60.us-east-2.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Mar 25 20:46:04 ip-172-31-12-60.us-east-2.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Mar 25 20:46:04 ip-172-31-12-60.us-east-2.compute.internal httpd[30292]: Server configured, listening on: port 80
[ec2-user@ip-172-31-12-60 html]$
```

23. En el navegador se vería así:



## Conclusión

La implementación de Apache (httpd) en una instancia de AWS nos permitió comprender el proceso de configuración y despliegue de un servidor web en la nube. A lo largo de la práctica, verificamos la instalación y ejecución del servicio, configuramos los permisos adecuados en el **Security Group** para permitir el acceso HTTP y accedimos al sitio web mediante la dirección IP pública de la instancia.

Este ejercicio resalta la importancia de la administración de servidores en entornos cloud, permitiéndonos explorar aspectos clave como la seguridad de acceso, la configuración de servicios en Linux y la gestión de instancias en AWS. La práctica también nos muestra cómo AWS facilita la escalabilidad y disponibilidad de aplicaciones web mediante el uso de instancias EC2, lo que es fundamental en la infraestructura moderna