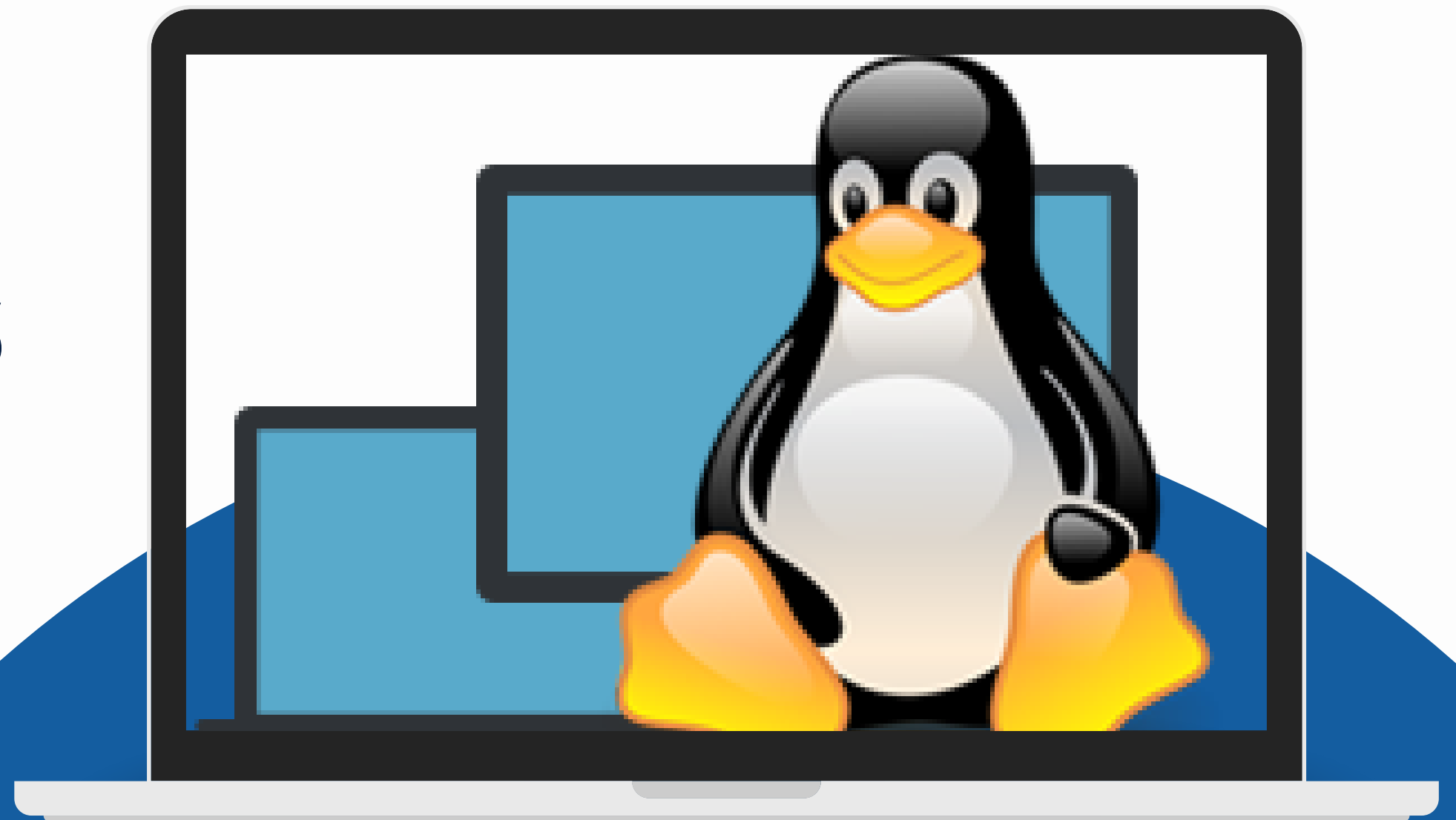




# Lab - Servicios administrativos

Presentación realizada por Brendon Buriol



# Objetivos

- Verificaremos el estado del servicio httpd para asegurarnos de que se esté ejecutando y de que pueda establecer una conexión http a la dirección IP del host local.
- También aprenderemos cómo monitorear la instancia EC2 de Amazon Linux 2.
  - Usando el comando top de Linux
  - Usando AWS CloudWatch

# Tarea 1: Verificar el estado del servicio httpd

Httpd es el servicio para el servidor http Apache que está instalado en nuestro host. Este es un servidor web liviano como los que ejecutan sus sitios web favoritos (piense en... digamos amazon.com). En este ejercicio, verificaremos el estado del servicio httpd, lo iniciaremos usando el comando systemctl y verificaremos que el servicio esté funcionando.

El comando sudo systemctl status httpd.service y el resultado dice que la sección del servicio httpd está inactiva (dead).

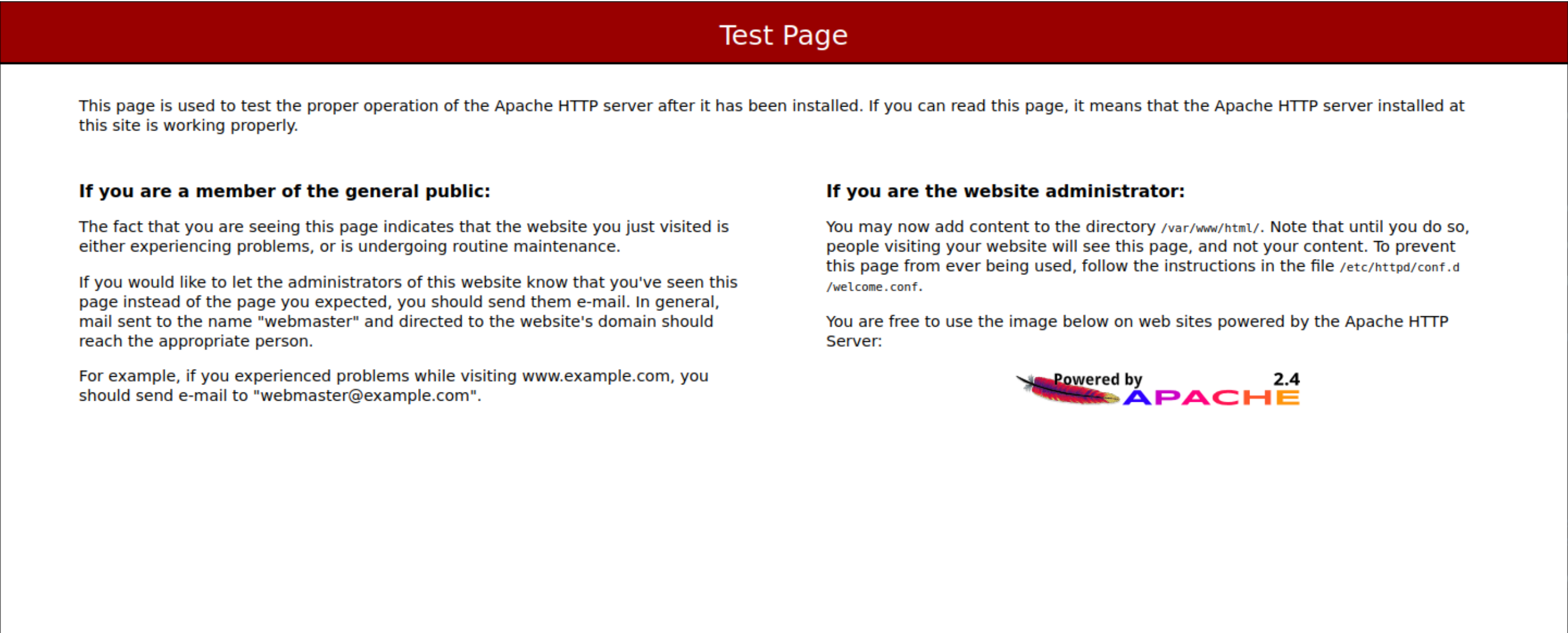
```
[ec2-user@ip-10-0-10-104 ~]$ sudo systemctl status httpd.service ←
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: inactive (dead) ←
   Docs: man:httpd.service(8)
```

Luego de haber iniciado el servicio, el comando sudo systemctl status httpd.service y el resultado dice que la sección del servicio httpd está activa (en ejecución).

```
[ec2-user@ip-10-0-10-104 ~]$ sudo systemctl start httpd.service
[ec2-user@ip-10-0-10-104 ~]$ sudo systemctl status httpd.service ←
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: di
sabled)
   Active: active (running) since Tue 2023-10-31 17:19:20 UTC; 22s ago
   Docs: man:httpd.service(8)
  Main PID: 2594 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/
sec:  0 B/sec"
   CGroup: /system.slice/httpd.service
           └─2594 /usr/sbin/httpd -DFOREGROUND
             └─2595 /usr/sbin/httpd -DFOREGROUND
               └─2596 /usr/sbin/httpd -DFOREGROUND
                 └─2602 /usr/sbin/httpd -DFOREGROUND
                   └─2604 /usr/sbin/httpd -DFOREGROUND
                     └─2609 /usr/sbin/httpd -DFOREGROUND

Oct 31 17:19:20 ip-10-0-10-104.us-west-2.compute.internal systemd[1]: Startin...
Oct 31 17:19:20 ip-10-0-10-104.us-west-2.compute.internal systemd[1]: Started...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-10-0-10-104 ~]$ sudo systemctl status httpd.service
```

Ahora que httpd se está ejecutando, verificaremos que funciona correctamente. Abrimos una nueva pestaña en el navegador e ingresamos: `http://<publicip>`. Reemplazaremos `<publicip>` con la ip pública que recuperamos al comienzo del laboratorio.



Cuando el httpd se ejecute correctamente, aparecerá una página de prueba de Apache con información general sobre el funcionamiento adecuado del servidor Apache HTTP.



# Tarea 2: Monitoreo de la instancia EC2 de Linux

En este ejercicio, utilizaremos comandos de Linux para monitorear la instancia EC2 de Amazon Linux2. También abriremos la consola de AWS e iniciaremos sesión en CloudWatch para ver cómo este servicio puede proporcionarnos datos para monitorear la instancia.

```
top - 17:25:20 up 15 min, 1 user, load average: 0,00, 0,00, 0,02
Tasks: 86 total, 1 running, 47 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0,0 us, 0,2 sy, 0,0 ni, 99,8 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
KiB Mem : 966816 total, 430228 free, 76176 used, 460412 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 748132 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	123628	5472	3852	S	0,0	0,6	0:01.35	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kthreadd
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/0:0H
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	mm_percpu_wq
7	root	20	0	0	0	0	S	0,0	0,0	0:00.02	ksoftirqd/0
8	root	20	0	0	0	0	I	0,0	0,0	0:00.07	rcu_sched
9	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_bh
10	root	rt	0	0	0	0	S	0,0	0,0	0:00.00	migration/0
11	root	rt	0	0	0	0	S	0,0	0,0	0:00.00	watchdog/0
12	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/0
13	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/1
14	root	rt	0	0	0	0	S	0,0	0,0	0:00.00	watchdog/1
15	root	rt	0	0	0	0	S	0,0	0,0	0:00.22	migration/1
16	root	20	0	0	0	0	S	0,0	0,0	0:00.02	ksoftirqd/1
18	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/1:0H
20	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kdevtmpfs
21	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	netns
22	root	20	0	0	0	0	I	0,0	0,0	0:00.07	kworker/u4:1
32	root	20	0	0	0	0	I	0,0	0,0	0:00.11	kworker/u4:3

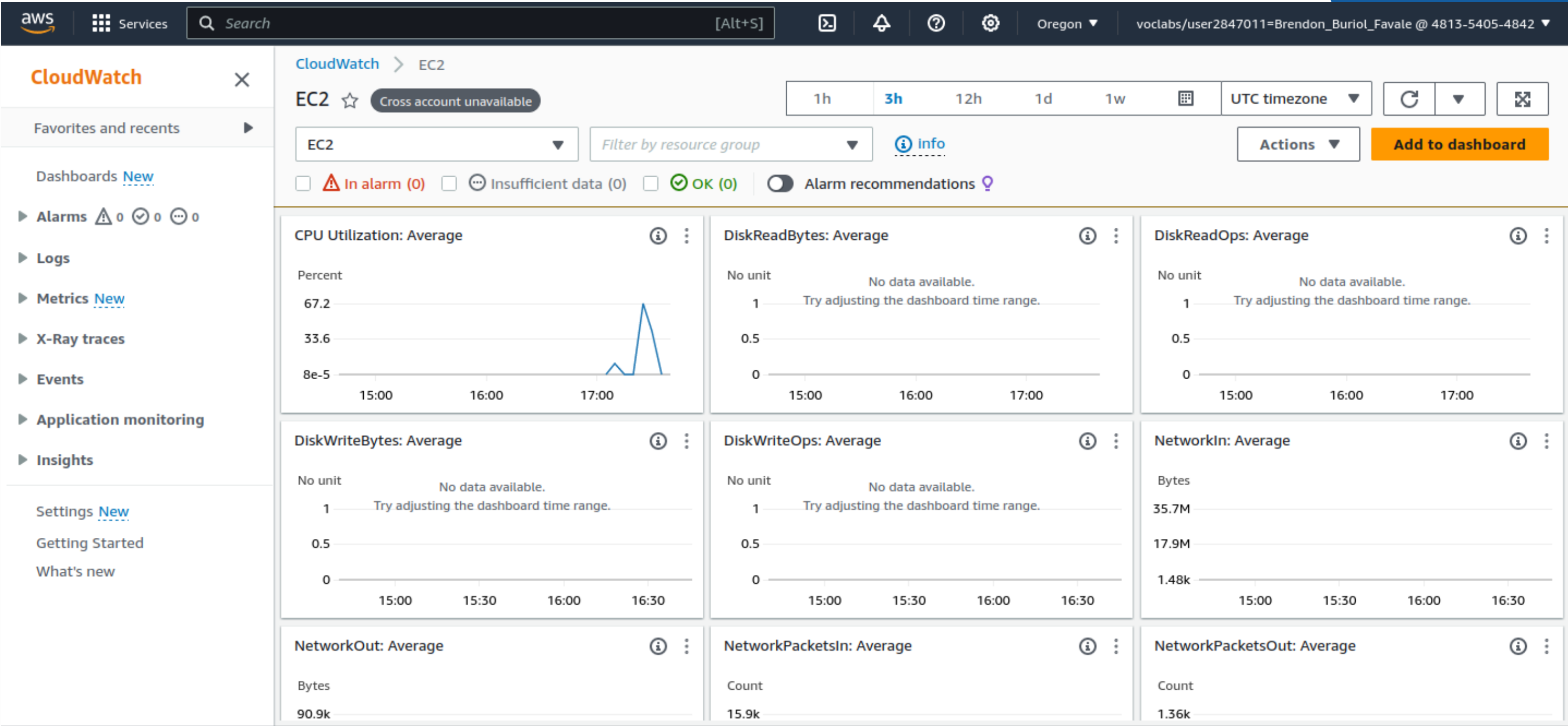
Al ejecutar el comando top, el resultado mostrará los procesos en ejecución actuales y el uso de recursos.

```
top - 17:25:58 up 16 min, 1 user, load average: 2,15, 0,46, 0,16
Tasks: 102 total, 15 running, 49 sleeping, 0 stopped, 0 zombie
%Cpu(s): 56,6 us, 35,3 sy, 0,0 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,0 si, 8,2 st
KiB Mem : 966816 total, 402756 free, 103644 used, 460416 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 720664 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2688	ec2-user	20	0	7580	92	0	R	13,6	0,0	0:01.49	stress
2685	ec2-user	20	0	7580	92	0	R	13,3	0,0	0:01.49	stress
2694	ec2-user	20	0	7580	92	0	R	13,3	0,0	0:01.50	stress
2697	ec2-user	20	0	7580	92	0	R	13,3	0,0	0:01.50	stress
2684	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.49	stress
2686	ec2-user	20	0	138656	11716	276	R	13,0	1,2	0:01.49	stress
2687	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.49	stress
2690	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.48	stress
2692	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.49	stress
2693	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.49	stress
2695	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.50	stress
2696	ec2-user	20	0	7580	92	0	R	13,0	0,0	0:01.61	stress
2689	ec2-user	20	0	138656	14356	276	R	12,6	1,5	0:01.48	stress
2691	ec2-user	20	0	7580	92	0	R	12,6	0,0	0:01.49	stress
8	root	20	0	0	0	0	I	0,3	0,0	0:00.08	rcu_sched
1	root	20	0	123628	5472	3852	S	0,0	0,6	0:01.35	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kthreadd
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/0:0H
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	mm_percpu_wq

El comando muestra un uso elevado del CPU después de ejecutar un script. Muestra al usuario como ec2-ser y el porcentaje de CPU.

Luego de ingresar a la plataforma de Amazon y luego al servicio EC2 CloudWatch, los gráficos que se muestran son la utilización de CPU, DiskReadBytes, DiskReadOps, DiskWriteBytes, DiskWriteOps y NetworkIn para las instancias EC2 de la cuenta.



Fin ;).