

## Servicios de gestión – Monitoreo

### Objetivos

En este laboratorio usted:

- Verifique el estado del servicio httpd para asegurarse de que se esté ejecutando y de que pueda establecer una conexión http a la dirección IP del host local.
- También aprenderá cómo monitorear su instancia EC2 de Amazon Linux 2.
- Usando el comando superior de Linux
- Uso de AWS CloudWatch

Los siguientes componentes se crean para usted como parte del entorno de laboratorio :

Amazon EC2: host de comandos (en la subred pública): inicie sesión en esta instancia para utilizar los comandos enumerados en esta práctica de laboratorio.

### Tarea 1: utilizar SSH para conectarse a una instancia EC2 de Amazon Linux

En esta tarea, se conectará a una instancia EC2 de Amazon Linux. Utilizará una utilidad SSH para realizar todas estas operaciones. Las siguientes instrucciones varían ligeramente dependiendo de si está utilizando Windows o Mac/Linux.

#### USUARIOS DE WINDOWS: USO DE SSH PARA CONECTARSE

Estas instrucciones son específicamente para usuarios de Windows. Si está utilizando macOS o Linux, pase a la siguiente sección.

3. En el panel **Información del laboratorio** , seleccione el enlace **PPK** y guarde el archivo. El nombre del archivo será similar a *Ec2KeyPair-PPK.ppk* . Normalmente su navegador lo guardará en el directorio de Descargas.
4. Tome nota de la dirección **PublicIP** .
5. Descargue **PuTTY** a SSH en la instancia de Amazon EC2.
6. Abrir **PuTTY .exe**

7. Configure el tiempo de espera de PuTTY para mantener abierta la sesión de PuTTY durante un período de tiempo más largo:

- Seleccionar **conexión**

The screenshot shows the AWS training and certification interface. On the left, there's a sidebar with navigation links like 'AWS service restrictions', 'Scenario', 'Start lab', 'Task 1: Use SSH to connect to an Amazon Linux EC2 Instance', 'Task 2: Exercise - Explore the Linux man pages', 'End lab', and 'Additional Resources'. The main content area displays a 'Machine Image (AMI) (EN)' section with instructions for connecting to an Amazon EC2 instance using PuTTY. A 'PUTTY Configuration' dialog box is open, showing the 'Connection' tab. The 'Options controlling the connection' section has 'Seconds between keepalives (0 to turn off)' set to 30. The 'Low-level TCP connection options' section has 'Disable Nagle's algorithm (TCP\_NODELAY option)' checked. The 'Internet protocol version' is set to 'Auto'. The 'Logical name of remote host' is empty, and the 'Logical name of remote host (e.g. for SSH key lookup)' is also empty. A 'Copied' tooltip is visible over the '30' value in the 'Seconds between keepalives' field.

The screenshot shows the AWS Management Console interface. On the left, there's a sidebar with navigation links like 'Panel de EC2', 'Vista global de EC2', 'Eventos', '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', 'Volumenes', 'Instancias', 'Administrador del ciclo de vida', 'Red y seguridad', 'Security Groups', 'Direcciones IP elásticas', and 'Grupos de ubicación'. The main content area displays the 'Detalles de red' section for an EC2 instance. The 'Dirección IPv4 pública' is '35.89.75.219 [dirección abierta]'. The 'Dirección IP asignada automáticamente' is '35.89.75.219 [IP pública]'. A 'PuTTY Configuration' dialog box is overlaid on the console, showing the 'Basic options for your PuTTY session' section. The 'Host (Name or IP address)' is '35.89.75.219', the 'Port' is '22', and the 'Connection type' is 'SSH'. The 'Default Settings' section is empty. A 'Copied' tooltip is visible over the '35.89.75.219' value in the 'Host (Name or IP address)' field.

10. Cuando se le solicite **iniciar sesión como** , ingrese:

**ec2-user**

Esto lo conectará a la instancia EC2.

The screenshot shows the AWS training and certification interface. On the left, there is a sidebar with the following sections:

- Introduction to a**
- Información del laboratorio**
  - 1 hora
  - Idiomas disponibles
  - Valoración
- Recursos**
  - PEM de par de claves de EC2
    - Descargar PEM
  - PPK de par de claves de EC2
    - Descargar PPK
  - LabRegion

In the center, a PuTTY terminal window is open, showing the following text:

```
35.89.75.219 - PuTTY
Unable to use certificate file "C:\Users\miguel iligaray\Downloads\Ec2KeyPair
-PPK.ppk" (PuTTY SSH-2 private key)
login as: ec2-user
```

On the right, a green box contains the text: "a estas instrucciones."

Below the terminal window, there are instructions for step 10:

- Select **Open** again.
- 9. Select **Yes**, to trust and con
- 10. When prompted **login as**, enter: **ec2-user** This will connect you to the EC2 instance.
- 11. Windows Users, skip ahead to the next task.

## Tarea 2: verificar el estado del servicio httpd

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

**Consejo útil** Es posible que tengas que usar **sudo** para completar este ejercicio si no eres root.

18. Verifique el estado del servicio httpd usando los comandos **systemctl** como se muestra a continuación y presionando ENTER

```
sudo systemctl status httpd.service
```

Esto indica que el servicio **httpd está cargado**, lo que significa que está instalado y listo para funcionar pero está **inactivo**. Entonces el siguiente paso es iniciarlo.

19. Verifique el estado del servicio httpd usando los comandos **systemctl** como se muestra a continuación y presionando ENTER

```
sudo systemctl start httpd.service
```

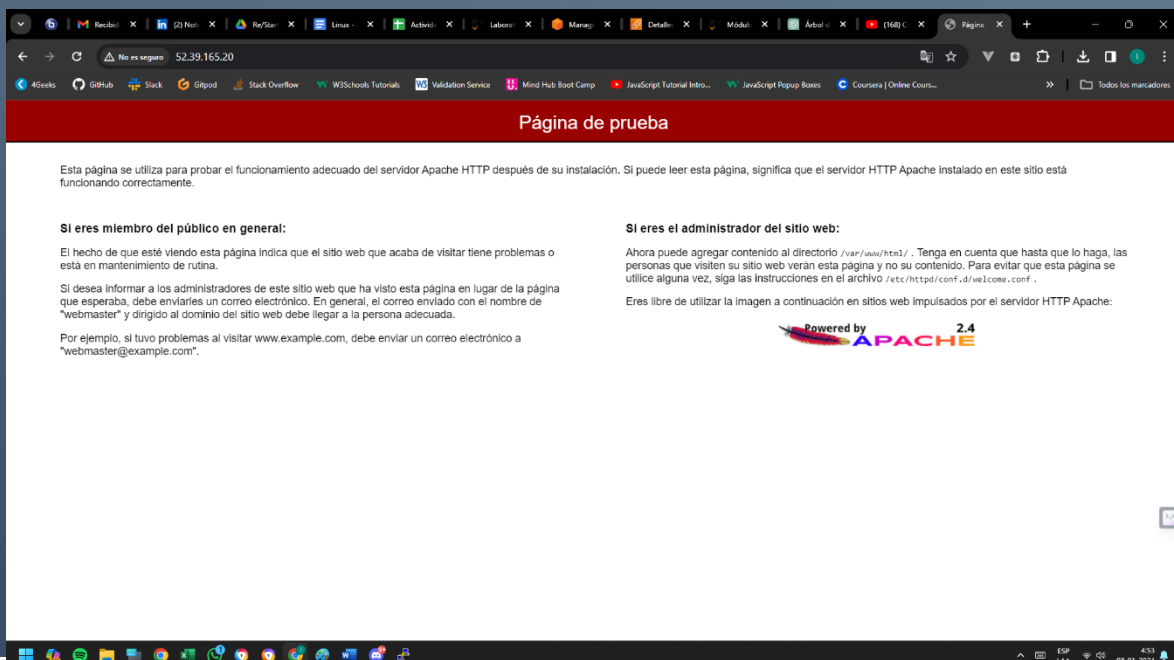
20. Verifique nuevamente el estado del servicio httpd usando los comandos **systemctl** como se muestra a continuación y presionando ENTER

```
sudo systemctl status httpd.service
```

21. Ahora que **httpd** se está ejecutando, verifiquemos que funciona correctamente. Abra una nueva pestaña en su navegador e ingrese: **http://<publicip>**. Reemplace <publicip> con la ip pública que recuperó al comienzo del curso.

22. Ahora puede detener el servicio ingresando el siguiente comando y presionando ENTER

```
sudo systemctl stop httpd.service
```



```

ec2-user@ip-10-0-10-164:~
~
~ \#/
~ V~' '->
~
~ / A newer version of Amazon Linux is available!
~ . /
~ / Amazon Linux 2023, GA and supported until 2028-03-15.
~ /m' / https://aws.amazon.com/linux/amazon-linux-2023/

No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-10-164 ~]$ 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)
[ec2-user@ip-10-0-10-164 ~]$ sudo systemctl start httpd.service
[ec2-user@ip-10-0-10-164 ~]$ 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: active (running) since Wed 2024-01-03 07:51:26 UTC; 12s ago
     Docs: man:httpd.service(8)
   Main PID: 2601 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes se
rved/sec: 0 B/sec"
   CGroup: /system.slice/httpd.service
           └─2601 /usr/sbin/httpd -DFOREGROUND
             └─2602 /usr/sbin/httpd -DFOREGROUND
               └─2604 /usr/sbin/httpd -DFOREGROUND
                 └─2609 /usr/sbin/httpd -DFOREGROUND
                   └─2611 /usr/sbin/httpd -DFOREGROUND
                     └─2616 /usr/sbin/httpd -DFOREGROUND

Jan 03 07:51:26 ip-10-0-10-164.us-west-2.compute.internal systemd[1]: Startin...
Jan 03 07:51:26 ip-10-0-10-164.us-west-2.compute.internal systemd[1]: Started...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-10-0-10-164 ~]$

```

## Tarea 3: Monitorear una instancia EC2 de Linux

En este ejercicio, utilizará comandos de Linux para monitorear la instancia EC2 de Amazon Linux2. También abrirá la consola de AWS e iniciará sesión en CloudWatch para ver cómo este servicio puede proporcionarle datos para monitorear su instancia.

**Consejo útil** Es posible que tengas que usar **sudo** para completar este ejercicio si no eres root.

23. Muestre la lista de procesos en ejecución ingresando el siguiente comando y presionando ENTER

Top

24. Ejecute el script **stress.sh** que simula una gran carga de trabajo en la instancia EC2 ingresando el siguiente comando y presionando ENTER

`./stress.sh & top`

25. Como en el paso 1, muestre la lista de procesos en ejecución ingresando el comando **top** y presionando ENTER

Puede ver que el proceso que acaba de ejecutar tiene un uso elevado de CPU. El script está diseñado para ejecutarse durante 6 minutos antes de detenerse.

En los siguientes pasos, abra la **Consola de administración de AWS** e inicie la aplicación **AWS CloudWatch** que le brindará una mejor visión de su instancia EC2.

26. En la parte superior derecha de su pantalla, seleccione el **AWS** botón. Esto muestra la **Consola de administración de AWS** en una nueva pestaña.
27. En la barra de búsqueda en la parte superior de la pantalla, ingrese **CloudWatch** y presione ENTER.
28. En la sección izquierda del panel de navegación, seleccione **Panel** y luego seleccione **Paneles automáticos**. En la lista de **paneles automáticos**, seleccione **EC2**

Esto abre el **panel de EC2** creado para usted por **AWS**.

Puede ver que, de forma predeterminada, el **panel de EC2 CloudWatch** muestra varias métricas, como la utilización de la CPU, lecturas y escrituras de disco...

Puede ver un aumento en la utilización de la CPU que coincide con el momento en que inició el script de estrés anteriormente.

### Nota

Los paneles son personalizables para que puedas agregar o eliminar widgets, reorganizarlos, personalizar colores... **AWS CloudWatch** ofrece muchas más funciones como alarmas o activadores de eventos que descubrirás más adelante y que lo convierten en un **servicio de AWS** clave para monitorear tus aplicaciones en tiempo real. Actualice el promedio de 5 minutos a 1 segundo para revisar las actualizaciones más rápidamente.

29. Espere 5 minutos y regrese al panel de AWS CloudWatch. Ves que la utilización de la CPU disminuyó.

### Nota

De forma predeterminada, AWS CloudWatch agrega datos durante 5 minutos antes de procesarlos. Esta es una configuración que se puede cambiar.

```
ec2-user@ip-10-0-10-164:~  
top - 07:56:29 up 10 min, 1 user, load average: 0.00, 0.04, 0.05  
Tasks: 87 total, 1 running, 47 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
KiB Mem : 966816 total, 431316 free, 76396 used, 459104 buff/cache  
KiB Swap: 0 total, 0 free, 0 used. 748072 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1076	root	20	0	0	0	0	S	0.3	0.0	0:00.17	xfsaild/nv+
1	root	20	0	123624	5532	3924	S	0.0	0.6	0:01.34	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	I	0.0	0.0	0:00.03	kworker/0:0
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:+
5	root	20	0	0	0	0	I	0.0	0.0	0:00.09	kworker/u4+
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_+
7	root	20	0	0	0	0	S	0.0	0.0	0:00.01	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	0:00.20	migration/1
16	root	20	0	0	0	0	S	0.0	0.0	0:00.03	ksoftirqd/1
18	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:+
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	0:00.09	kworker/u4+
119	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
181	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom_reaper
203	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	writeback
205	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kcompactd0
206	root	25	5	0	0	0	S	0.0	0.0	0:00.00	ksmd
207	root	39	19	0	0	0	S	0.0	0.0	0:00.00	khugepaged
208	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	crypto
209	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kintegrityd
210	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kblockd



```
ec2-user@ip-10-0-10-164:~  
top - 07:57:34 up 11 min, 1 user, load average: 7.40, 1.99, 0.71  
Tasks: 103 total, 15 running, 49 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 61.6 us, 38.4 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
KiB Mem : 966816 total, 340028 free, 167684 used, 459104 buff/cache  
KiB Swap: 0 total, 0 free, 0 used. 656784 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2676	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.53	stress
2677	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.52	stress
2678	ec2-user	20	0	138656	76600	212	R	14.3	7.9	0:05.53	stress
2679	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.53	stress
2680	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.53	stress
2682	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.54	stress
2684	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.54	stress
2685	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.52	stress
2686	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.54	stress
2687	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.54	stress
2689	ec2-user	20	0	7580	96	0	R	14.3	0.0	0:05.54	stress
2681	ec2-user	20	0	138656	12976	212	R	14.0	1.3	0:05.52	stress
2683	ec2-user	20	0	7580	96	0	R	14.0	0.0	0:05.53	stress
2688	ec2-user	20	0	7580	96	0	R	14.0	0.0	0:05.63	stress
1	root	20	0	123624	5532	3924	S	0.0	0.6	0:01.34	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	I	0.0	0.0	0:00.03	kworker/0:0
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:+
5	root	20	0	0	0	0	I	0.0	0.0	0:00.09	kworker/u4+
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_+
7	root	20	0	0	0	0	S	0.0	0.0	0:00.01	ksoftirqd/0
8	root	20	0	0	0	0	I	0.0	0.0	0:00.08	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	0:00.20	migration/1
16	root	20	0	0	0	0	S	0.0	0.0	0:00.03	ksoftirqd/1

