

Memoria Prácticas ISE. Monitorización de un Servidor Linux

Arturo Olivares Martos

30 de mayo de 2025

Resumen

En esta pequeña memoria, se incluirán las capturas más relevantes que pongan de manifiesto la monitorización de un servidor Linux, empleando para ello el software Node Exporter de Prometheus.

Índice

1. Node Exporter	2
2. Panel de Servicios	5
3. Uso de la CPU	5
3.1. Lanzamiento Carga CPU	6
3.2. Alerta de Uso de la CPU	6

1. Node Exporter

En la presente sección, describiremos cómo hemos instalado y configurado el software Node Exporter de Prometheus en la Máquina Virtual Rocky Linux.

1. En primer lugar, hemos de descargar y descomprimir el `node_exporter`, y guardarlo en `/usr/local/bin/`.

```
$ cd /tmp
$ curl -LO https://github.com/prometheus/node_exporter/releases/download/v0.18.1/node_exporter-0.18.1.linux-amd64.tar.gz
$ tar -xvf node_exporter-0.18.1.linux-amd64.tar.gz
$ sudo mv node_exporter-0.18.1.linux-amd64/node_exporter /usr/local/bin/
```

Este proceso se muestra en la Figura 1.

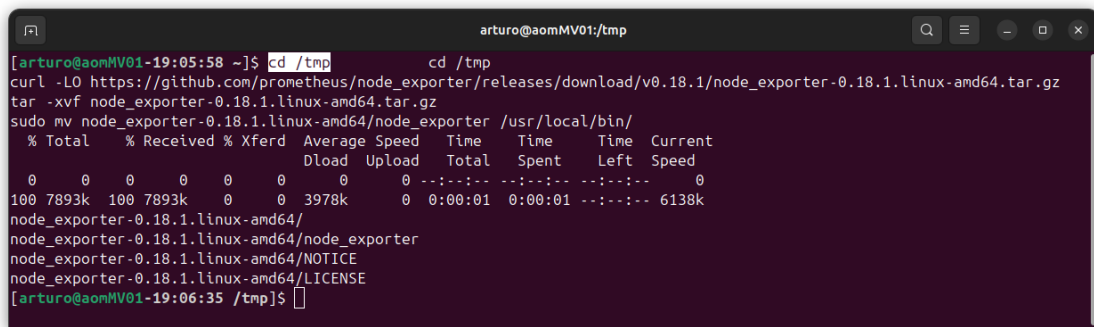


Figura 1: Descarga y descompresión de Node Exporter

2. A continuación, hemos de crear un usuario específico para ejecutar el `node_exporter`, de forma que no se ejecute con permisos de `text`. Esto es una buena práctica de seguridad, ya que limita los permisos del servicio. También cambiamos los permisos del binario para que el usuario creado sea el propietario.

```
$ sudo useradd -rs /bin/false node_exporter
$ sudo chown node_exporter:node_exporter /usr/local/bin/node_exporter
```

También le indicamos a SELinux que este fichero es un ejecutable:

```
$ sudo chcon -t bin_t /usr/local/bin/node_exporter
```

3. Ahora, hemos de crear un archivo de configuración para el servicio `node_exporter`. Este archivo se encuentra en `/etc/systemd/system/node_exporter.service` y contiene la configuración del servicio. El contenido del archivo es el siguiente:

```
[Unit]
Description=Node Exporter
After=network.target

[Service]
User=node_exporter
Group=node_exporter
Type=simple
ExecStart=/usr/local/bin/node_exporter --collector.systemd
--collector.processes

[Install]
WantedBy=multi-user.target
```

4. A continuación, hemos de abrir el puerto 9100 en el firewall para permitir el acceso al servicio `node_exporter` (puesto que, por defecto, este servicio escucha en el puerto 9100). Recordamos que también debemos hacer uso de **SELinux** para evitar problemas de permisos. Para ello, ejecutamos los siguientes comandos:

```
$ sudo firewall-cmd --add-port=9100/tcp --permanent
$ sudo firewall-cmd --reload
$ sudo semanage port -a -t http_port_t -p tcp 9100
```

La Figura 2 muestra el proceso de apertura del puerto 9100 en el firewall.



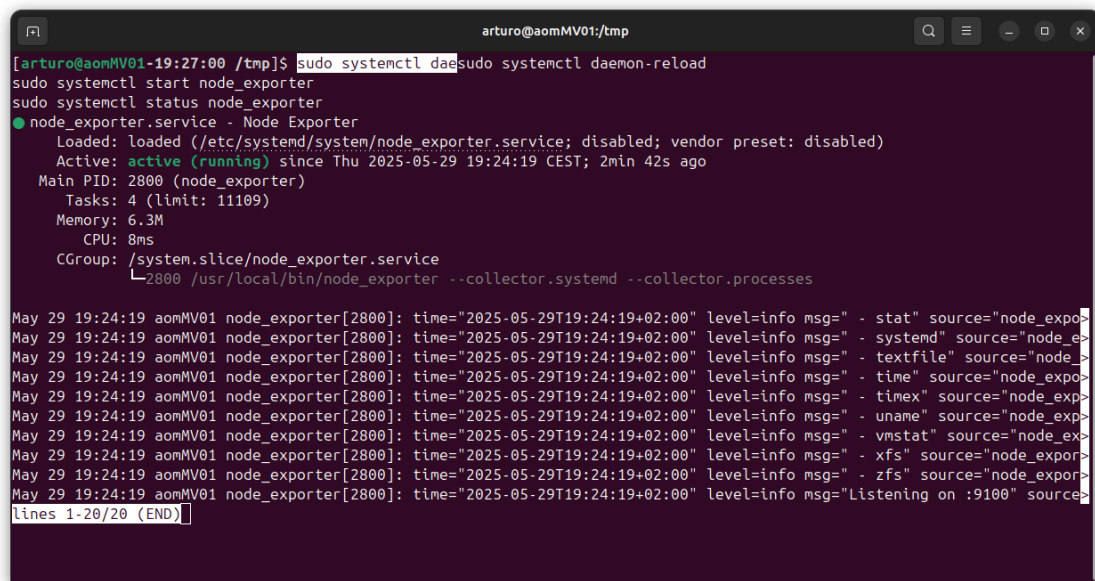
```
arturo@aomMV01/tmp
[arturo@aomMV01-19:18:51 /tmp]$ sudo firewall-cmd --add-port=9100/tcp --permanent
sudo firewall-cmd --reload
sudo semanage port -a -t http_port_t -p tcp 9100
Warning: ALREADY_ENABLED: 9100:tcp
success
success
Port tcp/9100 already defined, modifying instead
[arturo@aomMV01-19:19:09 /tmp]$
```

Figura 2: Apertura del puerto 9100 en el firewall

5. Ahora, hemos de recargar el demonio `systemd` para que reconozca el nuevo servicio y luego iniciar el servicio `node_exporter`. Comprobamos que el servicio se ha iniciado correctamente y que está escuchando en el puerto 9100:

```
$ sudo systemctl daemon-reload
$ sudo systemctl start node_exporter
$ sudo systemctl status node_exporter
```

La Figura 3 muestra el estado del servicio `node_exporter`. Habilitamos el servicio para que se inicie automáticamente al arrancar la máquina:



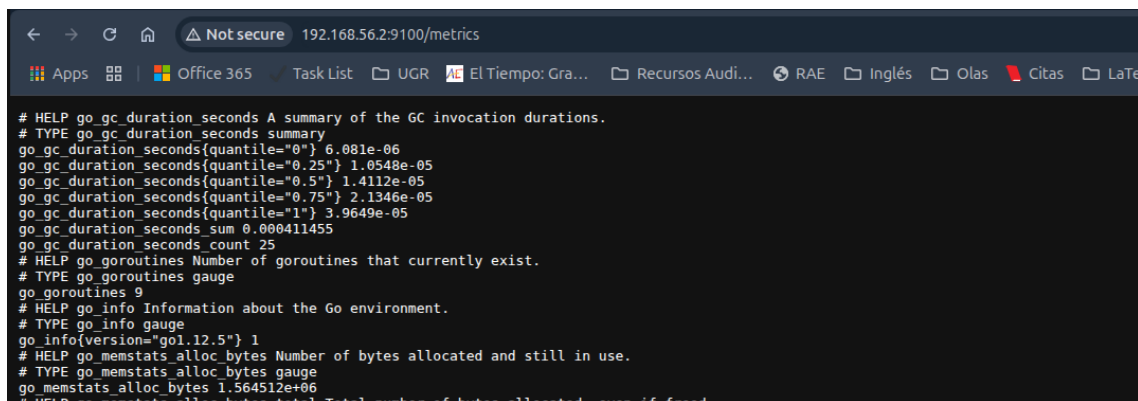
```
arturo@aomMV01:/tmp
[arturo@aomMV01-19:27:00 /tmp]$ sudo systemctl daemon-reload
sudo systemctl start node_exporter
sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; disabled; vendor preset: disabled)
   Active: active (running) since Thu 2025-05-29 19:24:19 CEST; 2min 42s ago
     Main PID: 2800 (node_exporter)
        Tasks: 4 (limit: 11109)
       Memory: 6.3M
          CPU: 8ms
    CGroup: /system.slice/node_exporter.service
            └─2800 /usr/local/bin/node_exporter --collector.systemd --collector.processes

May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - stat" source="node_exp
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - systemd" source="node_e
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - textfile" source="node_e
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - time" source="node_exp
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - timex" source="node_exp
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - uname" source="node_exp
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - vmstat" source="node_ex
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - xfs" source="node_exp
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg=" - zfs" source="node_exp
May 29 19:24:19 aomMV01 node_exporter[2800]: time="2025-05-29T19:24:19+02:00" level=info msg="Listening on :9100" source=
lines 1-20/20 (END)
```

Figura 3: Estado del servicio Node Exporter

```
$ sudo systemctl enable node_exporter
```

Una vez realizado todo, podemos comprobar que el servicio `node_exporter` está funcionando correctamente accediendo desde un navegador a la dirección dada por `http://<IP Maquina Virtual>:9100/metrics`. Aquí vemos una lista de métricas en formato texto, que son las que `node_exporter` está exponiendo. La Figura 4 muestra un ejemplo de estas métricas.



```
Notsecure 192.168.56.2:9100/metrics

# HELP go_gc_duration_seconds A summary of the GC invocation durations.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 6.081e-06
go_gc_duration_seconds{quantile="0.25"} 1.0548e-05
go_gc_duration_seconds{quantile="0.5"} 1.4112e-05
go_gc_duration_seconds{quantile="0.75"} 2.1346e-05
go_gc_duration_seconds{quantile="1"} 3.9649e-05
go_gc_duration_seconds_sum 0.000411455
go_gc_duration_seconds_count 25
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 9
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.12.5"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 1.564512e+06
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed
```

Figura 4: Métricas expuestas por Node Exporter

También podemos comprobar que el servicio funciona correctamente desde Prometheus, comprobando que funciona correctamente. La Figura 5 muestra cómo se ve el servicio en la interfaz de Prometheus.

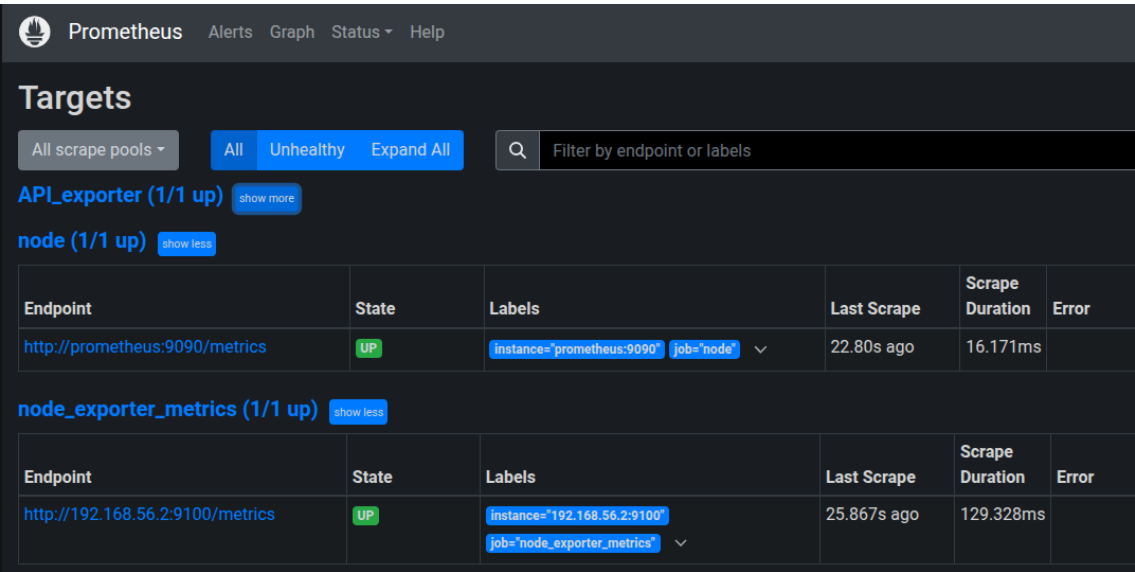


Figura 5: Node Exporter en Prometheus

2. Panel de Servicios

En el siguiente panel, mostramos si dos servicios (en este caso SSHD y HTTPD) están activos o no. Varias pruebas se ven en las Figuras 6 y 7.

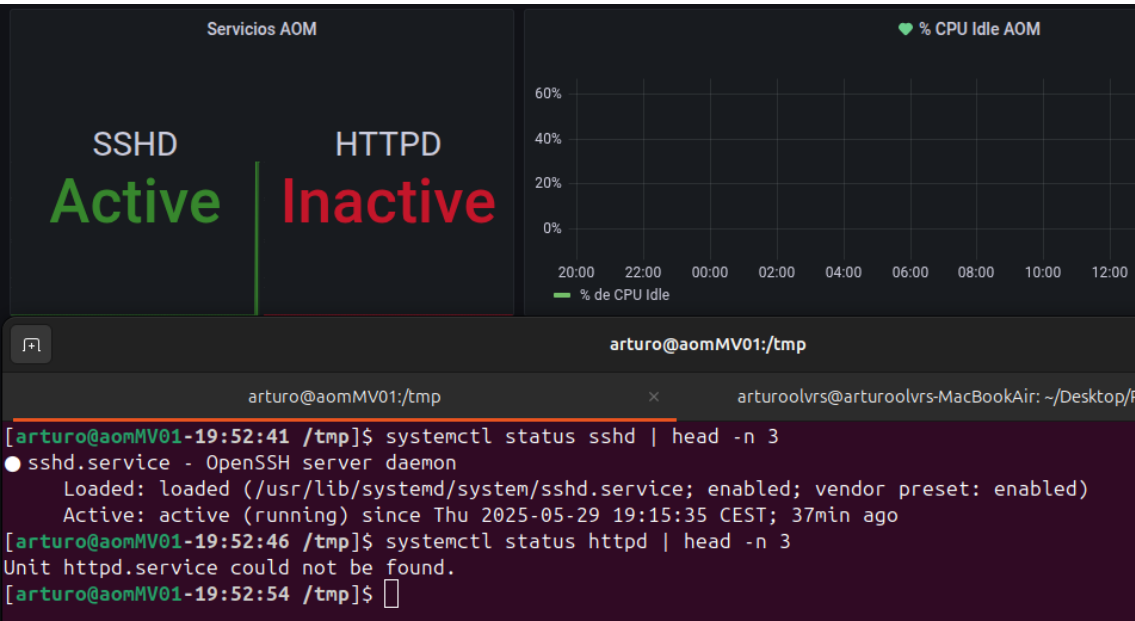


Figura 6: Panel de Servicios. SSHD activo y HTTPD desinstalado.

3. Uso de la CPU

En el siguiente panel, mostramos el uso de la CPU del servidor. Añadimos también el hecho de poner una alerta si el uso de la CPU supera el 75 % durante más de 5 minutos.

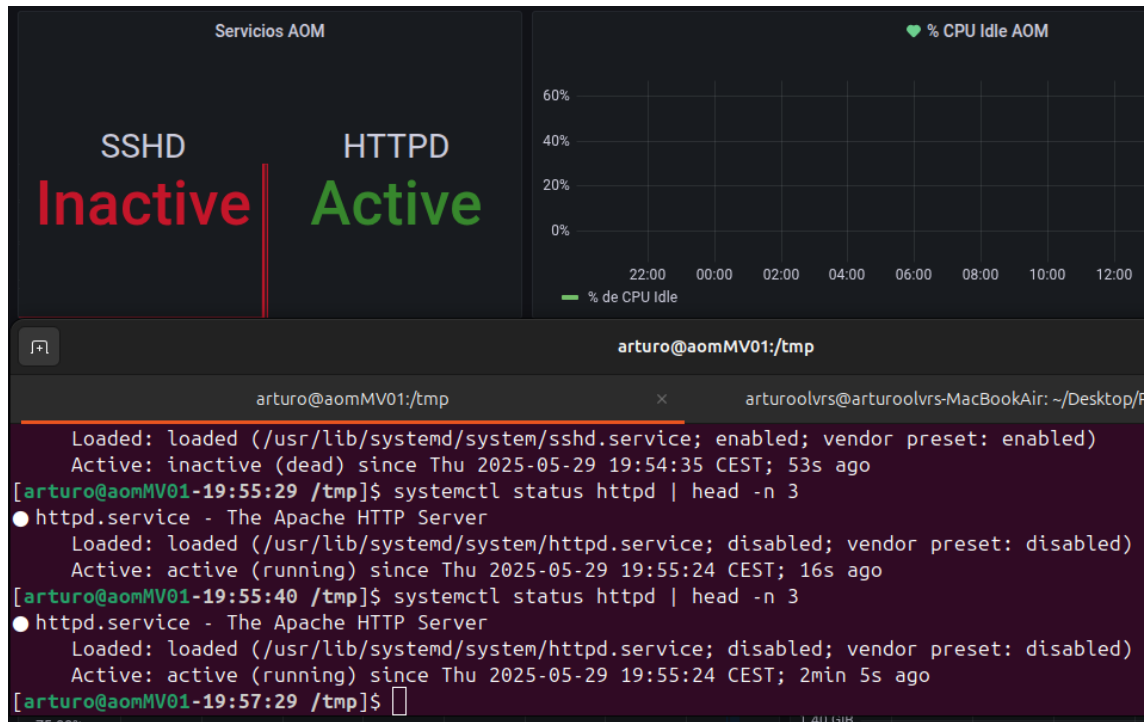


Figura 7: Panel de Servicios. SSHD inactivo y HTTPD activo.

3.1. Lanzamiento Carga CPU

El lanzamiento de carga en la CPU se ha realizado con el siguiente comando:

```
$ stress-ng --cpu 4 --timeout 1000s
```

La Figura 8 muestra el uso de la CPU antes y después de lanzar la carga. Además, en la parte inferior del panel, se muestra el comando que se ha ejecutado para lanzar la carga en la CPU.

3.2. Alerta de Uso de la CPU

La Figura 9 muestra el disparo de alerta que se ha generado al superar el 75 % de uso de la CPU durante más de 5 minutos. Como vemos, se detectó rápidamente el uso elevado de la CPU y se generó una alerta pendiente. Cuando está por encima del 75 % durante más de 5 minutos, se genera una alerta pendiente.

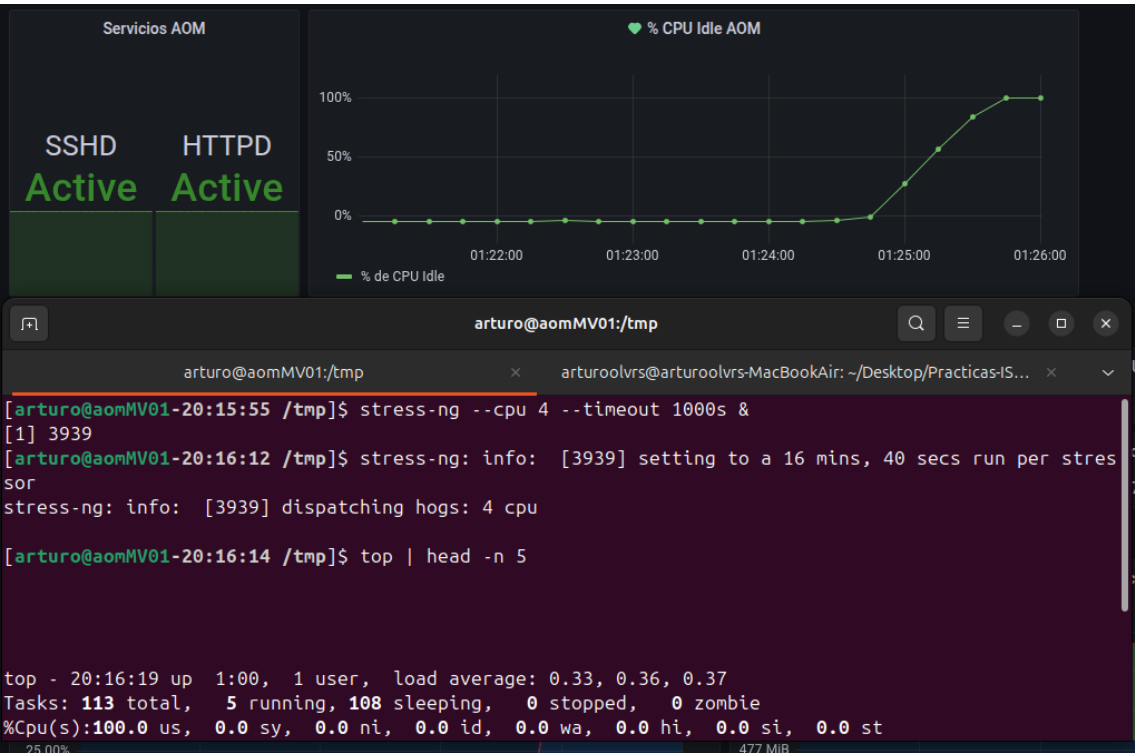


Figura 8: Uso de la CPU antes y después de lanzar la carga.



Figura 9: Alerta de uso elevado de la CPU.