Paso 1:

Vagrantfile

Vagrant.configure("2") do |config|

config.vm.define :servidorMonitor do |servidorMonitor|

servidorMonitor.vm.box = "bento/ubuntu-22.04"

servidorMonitor.vm.network :private\_network, ip: "192.168.60.3"

servidorMonitor.vm.provider "virtualbox" do |vb|

vb.memory = "1024"

vb.cpus = 2

end

servidorMonitor.vm.provision "shell", inline: <<-SHELL

# Actualizar repositorios

sudo apt-get update -y

# Instalar wget y curl si no están disponibles

sudo apt-get install -y wget curl

SHELL

end

end

Paso 2:

vagrant up

Paso 3:  
vagrant ssh servidorMonitor

Paso 4:  
cd /tmp

wget https://github.com/prometheus/prometheus/releases/download/v2.50.0/prometheus-2.50.0.linux-amd64.tar.gz

tar xvf prometheus-2.50.0.linux-amd64.tar.gz

sudo mv prometheus-2.50.0.linux-amd64 /opt/prometheus

sudo ln -s /opt/prometheus/prometheus /usr/local/bin/prometheus

sudo ln -s /opt/prometheus/promtool /usr/local/bin/promtool

Paso 5:  
sudo vi /etc/systemd/system/prometheus.service

[Unit]

Description=Prometheus

Wants=network-online.target

After=network-online.target

[Service]

User=prometheus

Group=prometheus

ExecStart=/usr/local/bin/prometheus \

--config.file=/opt/prometheus/prometheus.yml \

--storage.tsdb.path=/opt/prometheus/data

[Install]

WantedBy=default.target

Paso 6:  
sudo vi /opt/prometheus/prometheus.yml

global:

scrape\_interval: 15s

scrape\_configs:

- job\_name: "prometheus"

static\_configs:

- targets: ["localhost:9090"]

Paso 7:  
sudo useradd --no-create-home --shell /usr/sbin/nologin prometheus

sudo mkdir -p /opt/prometheus/data

sudo chown -R prometheus:prometheus /opt/Prometheus

Paso 8:  
sudo chown -R prometheus:prometheus /opt/prometheus/data

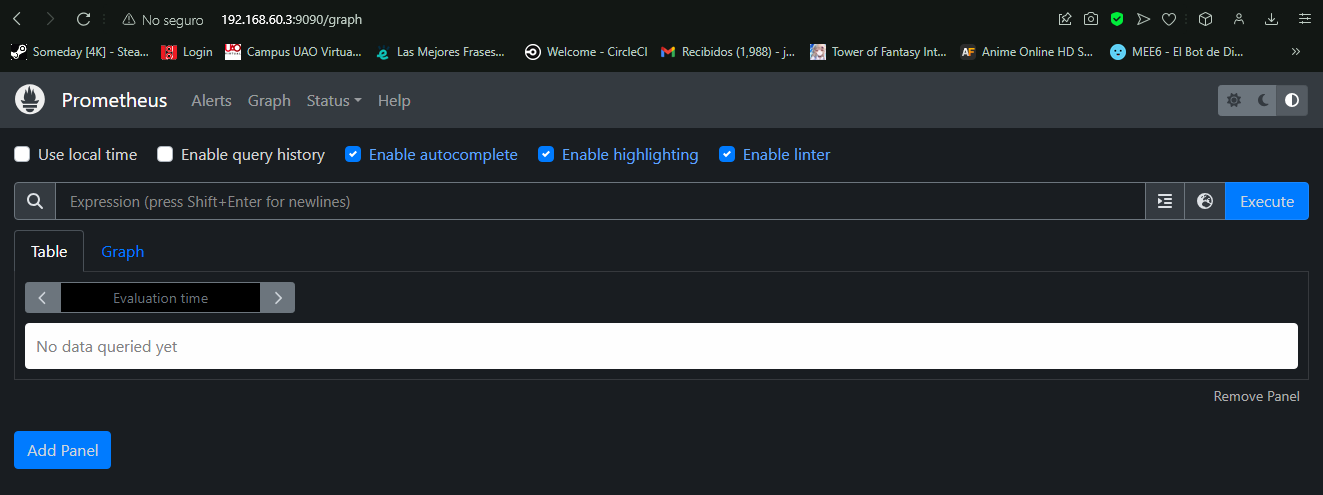
sudo chmod -R 775 /opt/prometheus/data

Paso 9:

sudo systemctl daemon-reload

sudo systemctl restart prometheus

sudo systemctl status Prometheus



Node Exporter:  
wget https://github.com/prometheus/node\_exporter/releases/download/v1.6.0/node\_exporter-1.6.0.linux-amd64.tar.gz

tar xvfz node\_exporter-1.6.0.linux-amd64.tar.gz

sudo mv node\_exporter-1.6.0.linux-amd64/node\_exporter /usr/local/bin/

sudo useradd -rs /bin/false node\_exporter

Paso 10:

sudo nano /etc/systemd/system/node\_exporter.service

[Unit]

Description=Node Exporter

Wants=network-online.target

After=network-online.target

[Service]

User=node\_exporter

ExecStart=/usr/local/bin/node\_exporter

[Install]

WantedBy=default.target

Paso 11:  
sudo systemctl daemon-reload

sudo systemctl start node\_exporter

sudo systemctl enable node\_exporter

Paso 12 añadir al Prometheus yml lo siguiente:  
scrape\_configs:

- job\_name: "node\_exporter"

static\_configs:

- targets: ["localhost:9100"]

sudo systemctl restart prometheus

**Verificaciones:**

Aplicación

Descripción generada automáticamente con confianza baja

1. node\_cpu\_seconds\_total

Esta métrica mide el tiempo total que la CPU ha pasado en diferentes estados (user, system, idle, iowait, etc.), desglosado por cada núcleo. Se expresa en segundos acumulados desde el inicio del sistema.

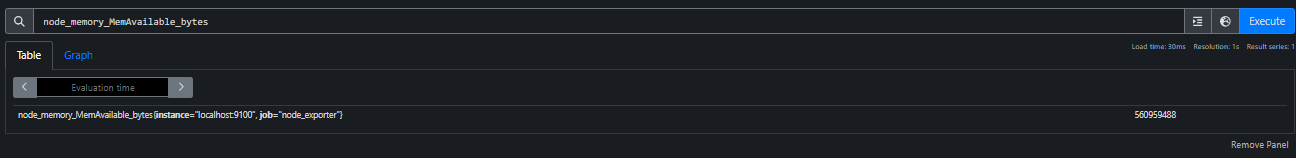
Uso en el monitoreo: Ayuda a identificar cuánta carga está enfrentando el procesador. Por ejemplo:

Si hay mucho tiempo en idle, el sistema está poco utilizado.

Si hay alto uso en iowait, puede haber problemas con el acceso a disco.

Alto uso en user o system indica que el CPU está ocupado con procesos del usuario o del sistema.

Beneficio: Detectar cuellos de botella en el procesamiento y optimizar el rendimiento al ajustar la carga o agregar más recursos.



Una pantalla verde

Descripción generada automáticamente con confianza media

2. node\_memory\_MemAvailable\_bytes

Representa la cantidad de memoria RAM disponible para nuevos procesos sin que sea necesario recurrir a la memoria de intercambio (swap). Considera memoria libre y memoria en caché reutilizable.

Uso en el monitoreo: Permite verificar si el sistema tiene suficiente memoria para operar eficientemente. Bajos niveles de memoria disponible pueden causar ralentizaciones, ya que el sistema recurre al swap o incluso provoca fallos en aplicaciones.

Beneficio: Ayuda a evitar problemas relacionados con la falta de memoria, como rendimiento degradado o cierres inesperados de aplicaciones críticas.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

3. \*\*node\_filesystem\_avail\_bytes (con filtro mountpoint="/")

Indica el espacio disponible en bytes en el sistema de archivos montado en el directorio raíz /.

Uso en el monitoreo: Garantiza que el almacenamiento tenga suficiente espacio libre para operaciones normales. Si el espacio se agota, pueden fallar procesos que dependen de la escritura, como logs, bases de datos, o aplicaciones.

Beneficio: Evitar problemas como interrupciones en servicios o pérdida de datos al monitorear y actuar antes de que se agote el espacio.

Parte Opcional

sudo apt update

sudo apt install -y software-properties-common

sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"

sudo apt install -y gnupg ca-certificates

sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://packages.grafana.com/gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/grafana-archive-keyring.gpg

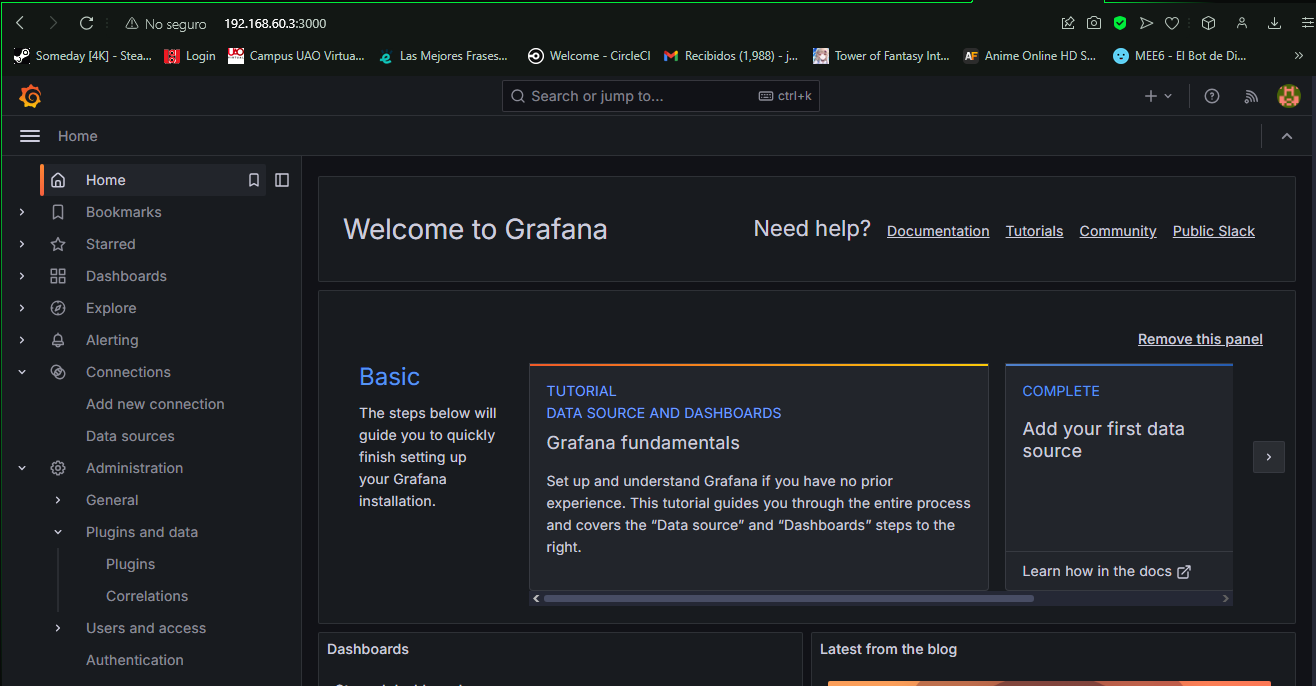
echo "deb [signed-by=/etc/apt/keyrings/grafana-archive-keyring.gpg] https://packages.grafana.com/oss/deb stable main" | sudo tee /etc/apt/sources.list.d/grafana.list

sudo apt update

sudo apt install -y Grafana

sudo systemctl start grafana-server

sudo systemctl enable grafana-server



Interfaz de usuario gráfica

Descripción generada automáticamente

Pantalla de un video juego

Descripción generada automáticamente con confianza media