**IES Ataúlfo Argenta**

Big Data Aplicado 2025

**Monitoreo de un**

**Clúster AWS EMR con**

**Prometheus y Grafana**

**Priscila Sabino**

****

Índice

[Parte 1: Configuración del clúster AWS EMR 2](#_Toc193893567)

[1.1 Creación del clúster EMR 2](#_Toc193893568)

[1.2 Conectar al nodo maestro 3](#_Toc193893569)

[Parte 2: Configuración de JMX Exporter 3](#_Toc193893570)

[2.1 Instalar JMX Exporter 3](#_Toc193893571)

[2.2 Crear archivo de configuración 4](#_Toc193893572)

[2.3 Configurar el NameNode para usar JMX Exporter 4](#_Toc193893573)

[2.4 Reiniciar el NameNode 4](#_Toc193893574)

[Parte 3: Despliegue de Prometheus y Grafana 5](#_Toc193893575)

[3.1 Crear una instancia EC2 para Prometheus y Grafana 5](#_Toc193893576)

[3.2 Instalar Prometheus: 5](#_Toc193893577)

[3.3 Configurar Prometheus: 6](#_Toc193893578)

[3.4 Instalar Grafana 7](#_Toc193893579)

[3.5 Configurar Grafana 7](#_Toc193893580)

[Parte 4: Visualización de métricas en Grafana 8](#_Toc193893581)

[4.1 Crear un dashboard en Grafana 8](#_Toc193893582)

[4.2 Explorar métricas: 9](#_Toc193893583)

[Conclusión 9](#_Toc193893584)

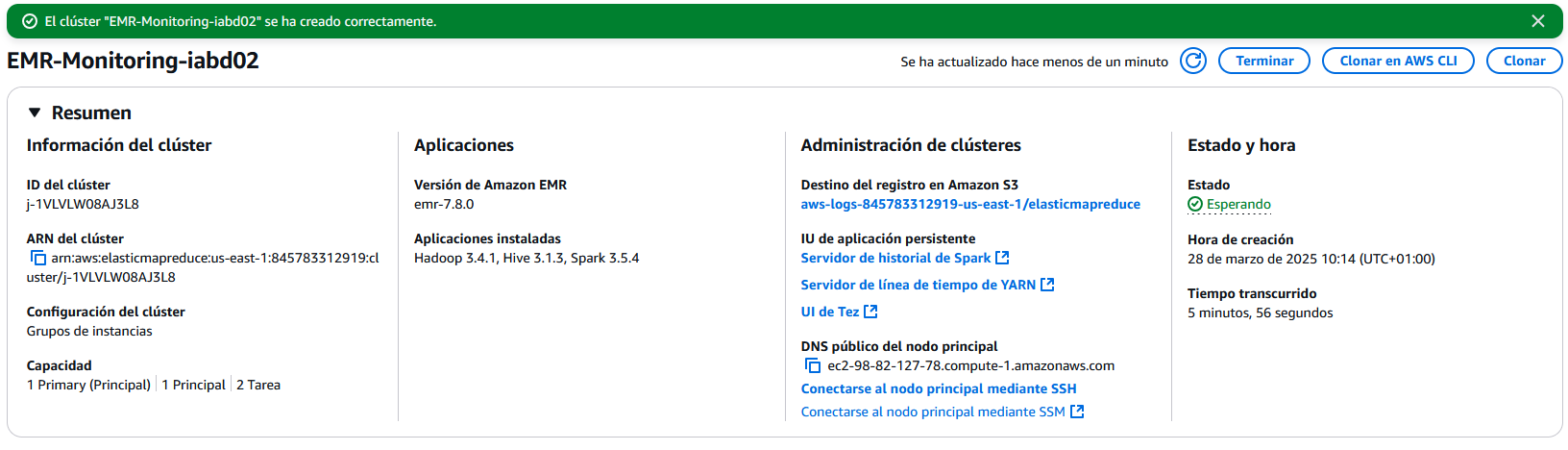
# Parte 1: Configuración del clúster AWS EMR

## Creación del clúster EMR

En la página <https://awsacademy.instructure.com/login/canvas>, iniciamos el laboratorio para montar nuestro clúster.

Accedemos a la consola de AWS y creamos el clúster EMR-Monitoring-iabd02.

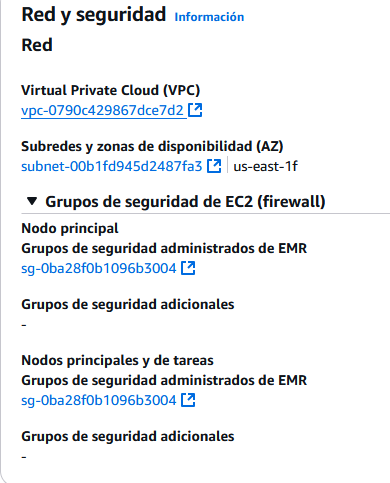
* Aplicaciones: Haddop 3.4.1, Spark 3.5.4 y Hive 3.1.3
* Permisos: Vokey y por defectos en los permisos IAM
* Instancias: 1 principal y 2 Tarea
* Estado del Clúster: **Esperando**



* Revisamos las reglas de entrada para acceder a SSH.



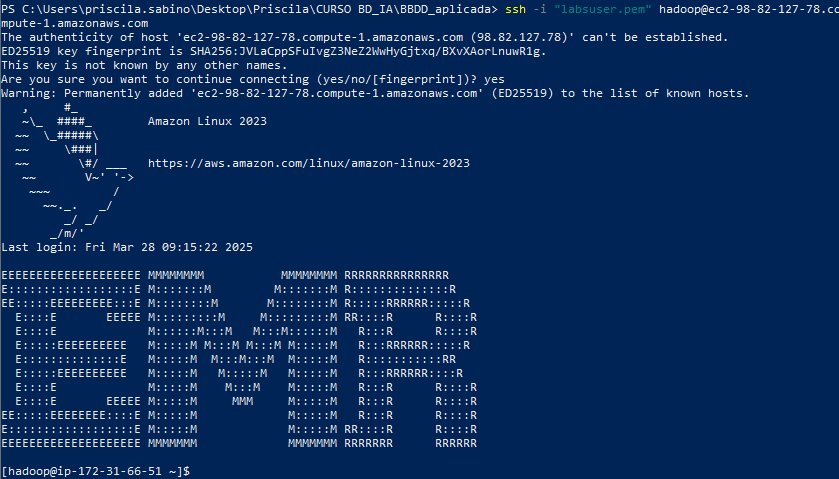
* VPC–0790c429867dce7d2



## Conectar al nodo maestro

Para establecer una conexión SSH con el nodo principal, utilizamos el siguiente comando:

ssh -i "labsuser.pem" [hadoop@ec2-98-82-127-78.compute-1.amazonaws.com](mailto:hadoop@ec2-3-239-56-32.compute-1.amazonaws.com)



# Parte 2: Configuración de JMX Exporter

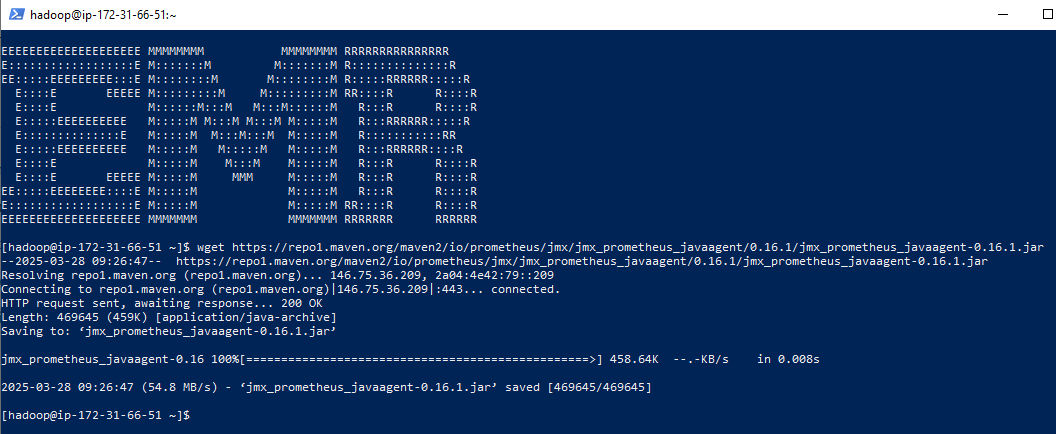
## Instalar JMX Exporter

**JMX (Java Management Extensions)** es una tecnología de monitoreo y gestión que forma parte de la plataforma Java. JMX se utiliza para proporcionar una forma estándar de exponer métricas y gestionar aplicaciones Java de manera remota. A través de JMX, se pueden obtener datos de rendimiento, estado y otros parámetros relevantes de una aplicación o sistema en tiempo real. JMX permite a los administradores y desarrolladores interactuar con las aplicaciones Java, acceder a sus métricas internas, configuraciones y controlarlas de manera remota.

El monitoreo continuo con **JMX** también contribuye a mejorar la **resiliencia del sistema**. Si se detecta una métrica anómala, como un uso de memoria excesivo, el sistema puede ser ajustado antes de que se convierta en un fallo o caída del servicio. JMX permite realizar estas modificaciones y ajustes en tiempo real, ayudando a evitar interrupciones en los trabajos de procesamiento.

Vemos la versión de java que tenemos en el nodo maestro e instalamos el **JMX con el comando:**

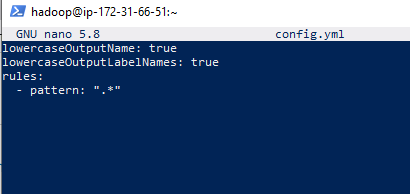
wget <https://repo1.maven.org/maven2/io/prometheus/jmx/jmx_prometheus_javaagent/0.16.1/jmx_prometheus_javaagent-0.16.1.jar>



## Crear archivo de configuración

Creamos un archivo config.yml para el JMV Exporter con el comando:

nano config.yml



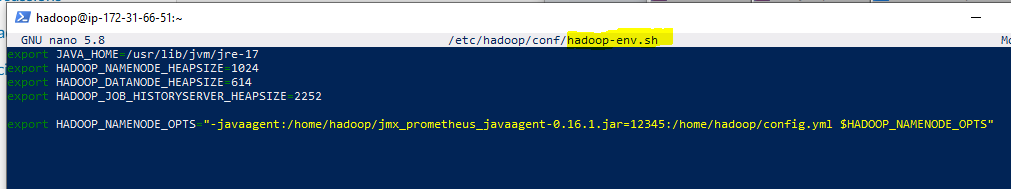
## Configurar el NameNode para usar JMX Exporter

Añadimos la línea en el fichero hadoop-env.sh.

Con los comandos:

sudo nano /etc/hadoop/conf/hadoop-env.sh

export HADOOP\_NAMENODE\_OPTS="-javaagent:/home/hadoop/jmx\_prometheus\_javaagent-0.16.1.jar=1234:/home/hadoop/config.yml $HADOOP\_NAMENODE\_OPTS"



## Reiniciar el NameNode

Reinicia el servicio del NameNode para aplicar los cambios:

sudo systemctl restart hadoop-hdfs-namenod

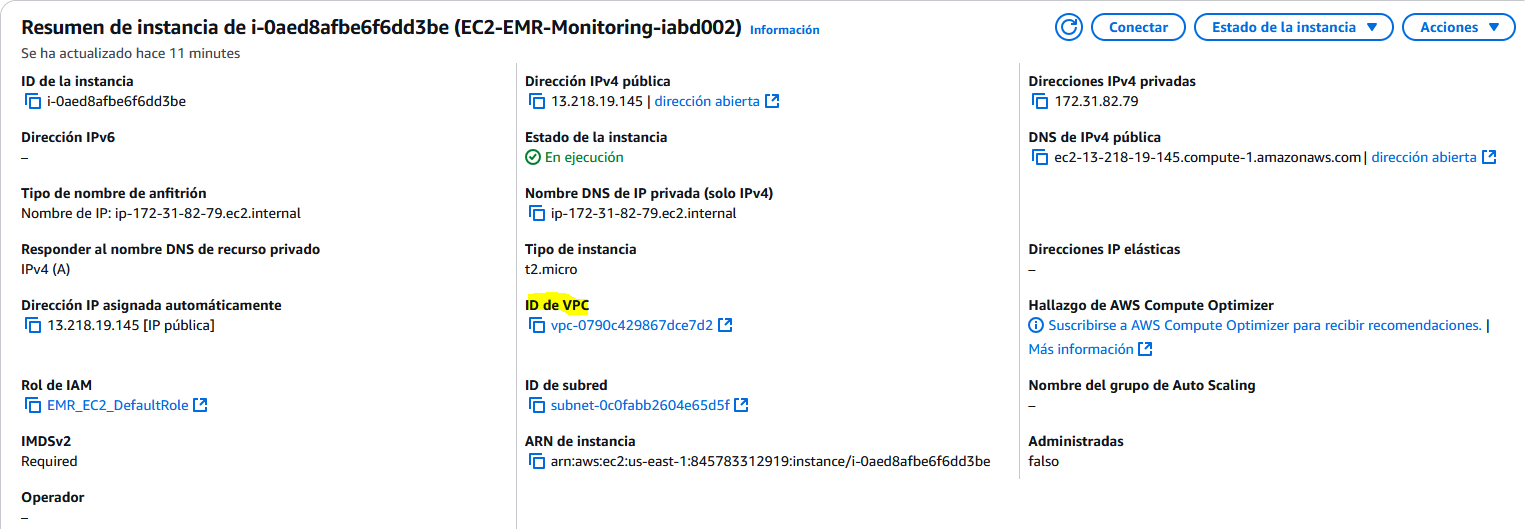


# Parte 3: Despliegue de Prometheus y Grafana

## Crear una instancia EC2 para Prometheus y Grafana

Crea una instancia EC2 con Ubuntu en la misma VPC que el clúster EMR.

Nombre de la instancia: **EC2-EMR-Monitoring-iabd002**

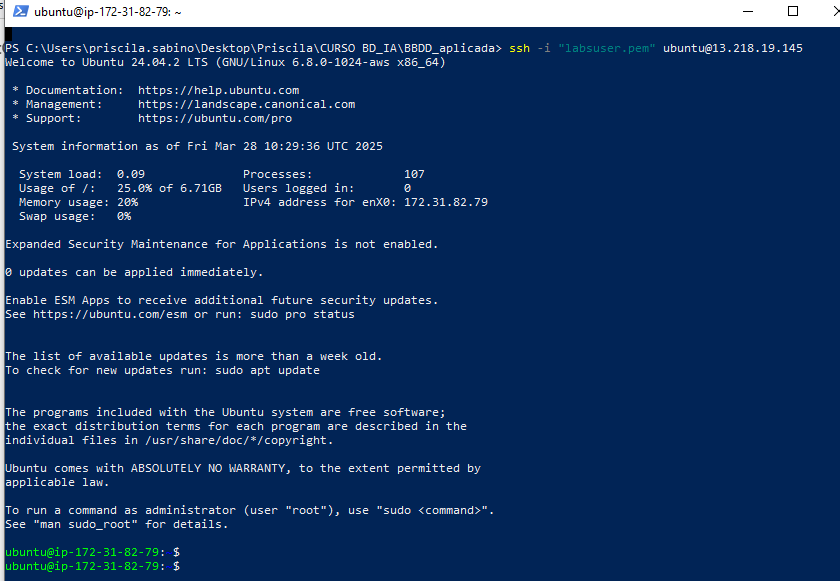


## Instalar Prometheus:

Conéctate a la instancia EC2 y sigue los pasos para instalar Prometheus:

* Conectamos con el comando:

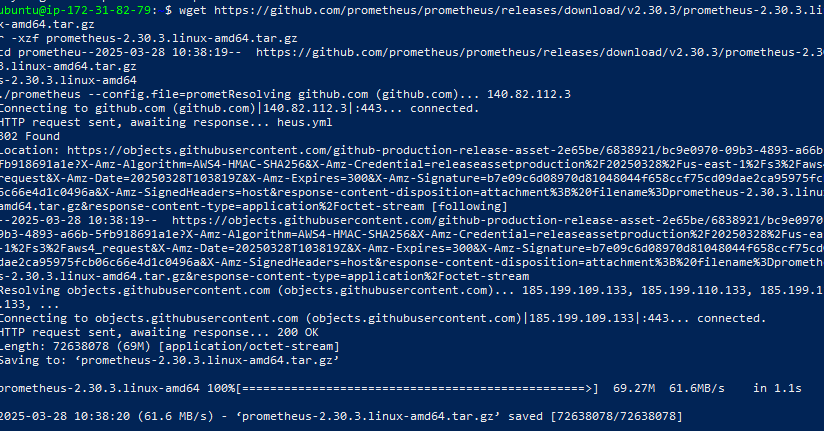
ssh -i "labsuser.pem" ubuntu@13.218.19.145



* Instalamos Promehetus con estos comandos:

wget https://github.com/prometheus/prometheus/releases/download/v2.30.3/prometheus-2.30.3.linux-amd64.tar.gz

tar -xzf prometheus-2.30.3.linux-amd64.tar.gz



cd prometheus-2.30.3.linux-amd64

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

## Configurar Prometheus:

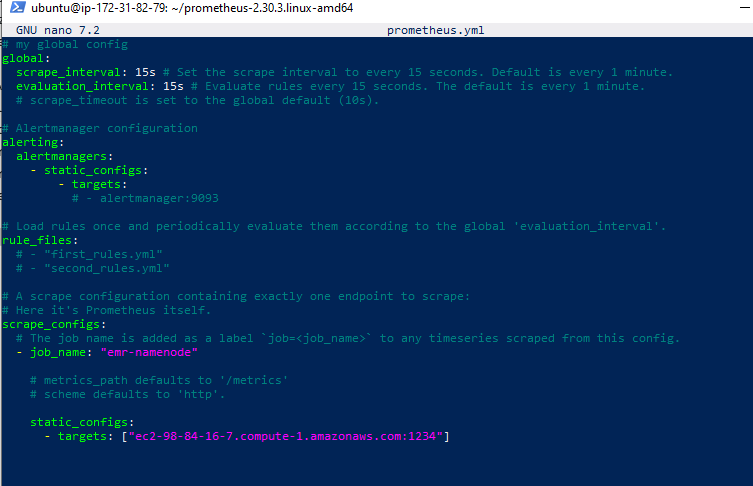
Edita el archivo prometheus.yml para agregar el clúster EMR como objetivo:

scrape\_configs:

- job\_name: 'emr-namenode'

static\_configs:

- targets: [“ec2-98-84-16-7.compute-1.amazonaws.com:1234”]



## Instalar Grafana

Instala Grafana en la misma instancia EC2. Lanzamos los comandos para la instalación de Grafana en la instancia:

sudo apt-get install -y apt-transport-https

sudo apt-get install -y software-properties-common wget

wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add –

echo "deb https://packages.grafana.com/oss/deb stable main" | sudo

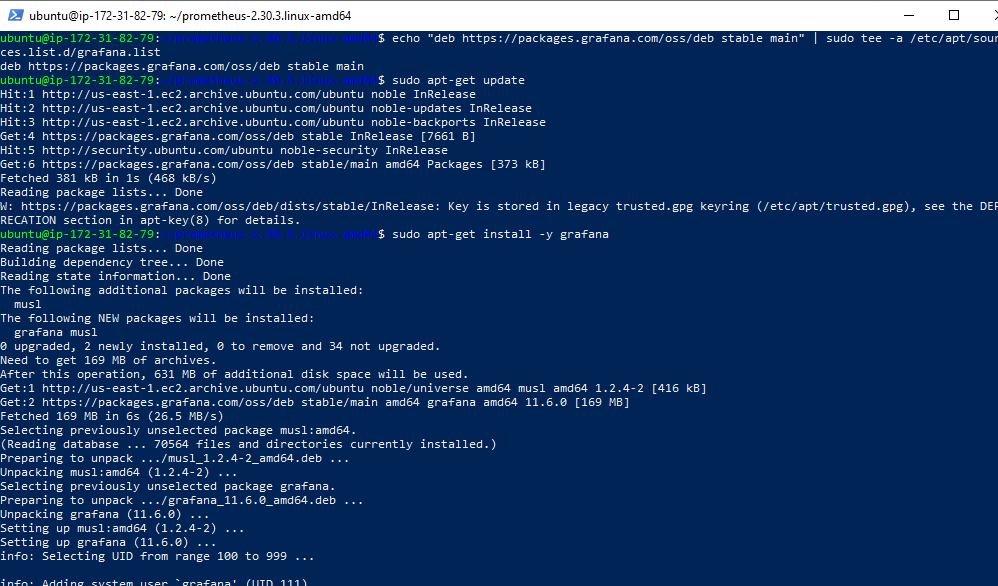
tee -a /etc/apt/sources.list.d/grafana.list

sudo apt-get update

sudo apt-get install grafana

sudo systemctl start grafana-server

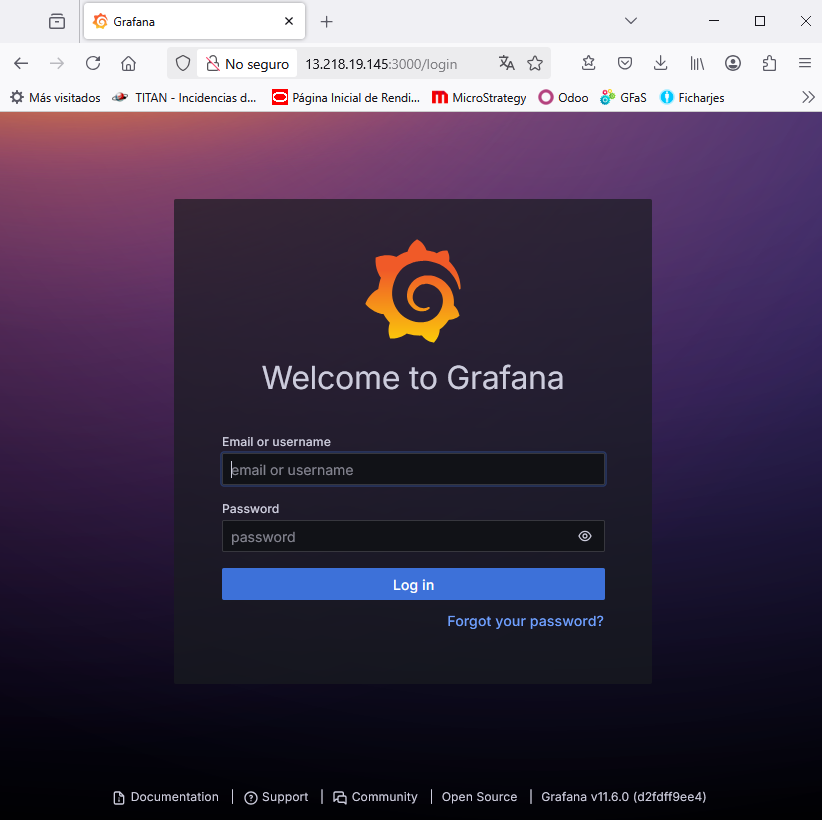
sudo systemctl enable grafana-server



## Configurar Grafana

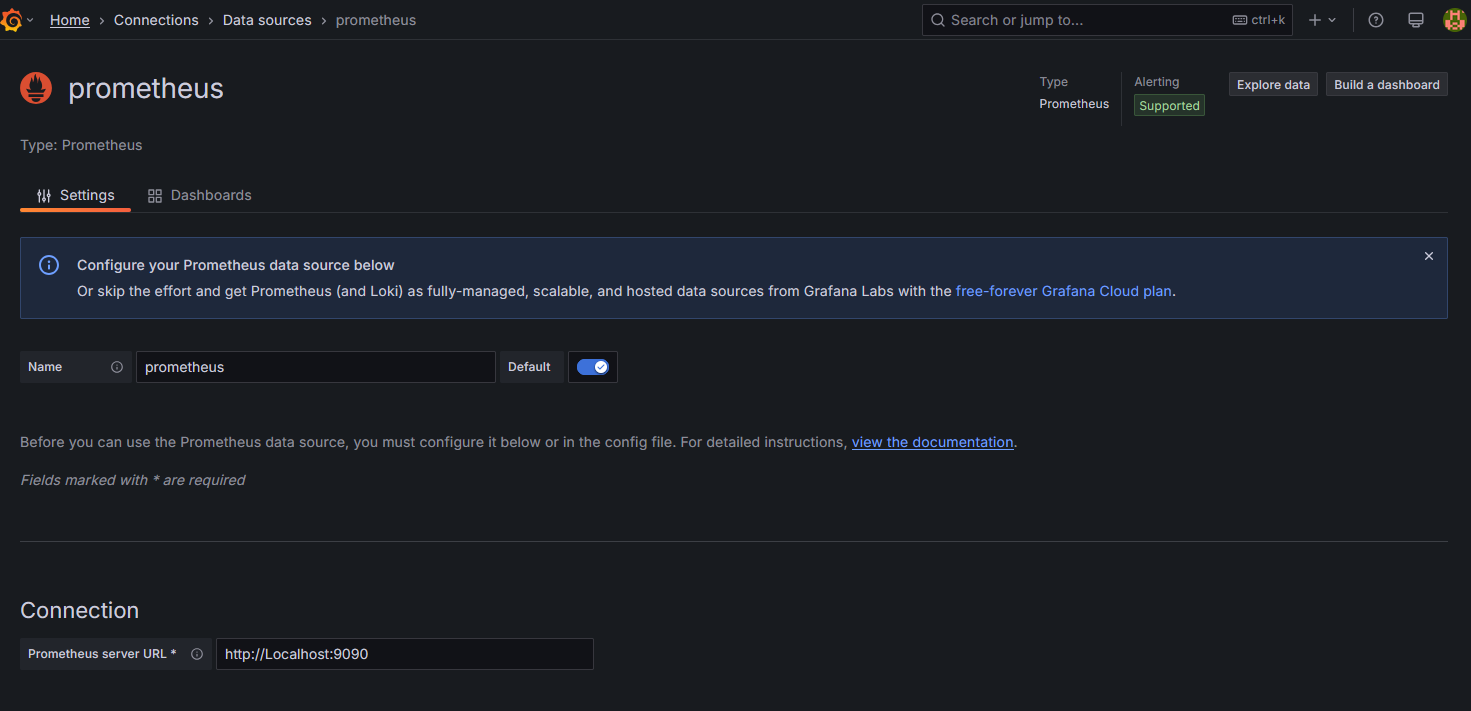
Accede a Grafana en http://34.203.204.237:3000.

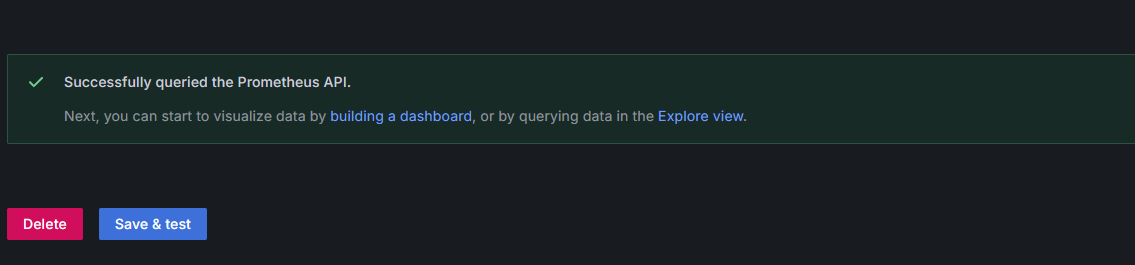
Usuario: admin 🡪 Contraseña: Castro-1



* Agrega Prometheus como fuente de datos:

▪ URL: <http://localhost:9090>





# Parte 4: Visualización de métricas en Grafana

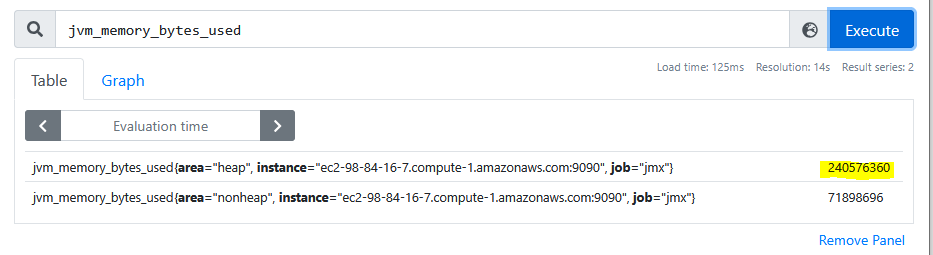
## 4.1 Crear un dashboard en Grafana

Crea un nuevo dashboard y agrega paneles para monitorear métricas como:

* Uso de CPU y RAM.

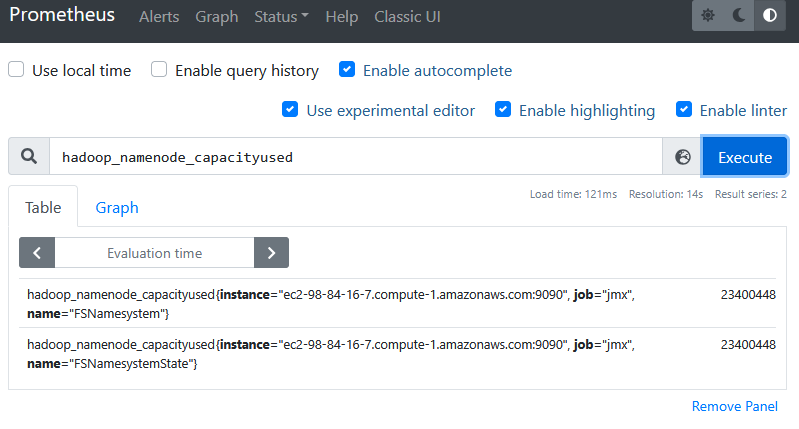
process\_cpu\_seconds\_total



jvm\_memory\_bytes\_used{area=”heap”}

* Espacio utilizado en HDFS.

hadoop\_namenode\_capacityused



* Estado del NameNode.

hadoop\_namenode\_health\_status

## 4.2 Explorar métricas:

Usa las métricas expuestas por JMX Exporter para crear gráficos en Grafana.



# Conclusión

En un entorno de clústeres como **Amazon EMR**, el monitoreo adecuado es esencial para asegurar la eficiencia y estabilidad del sistema. Existen varias métricas clave que debemos monitorear para garantizar que el clúster funcione correctamente. **Métricas de uso de recursos**, como el **uso de CPU**, **memoria**, **disco** y **red**, son fundamentales para identificar posibles cuellos de botella. **Métricas de Hadoop**, como el estado de los **Namenodes** y **Datanodes**, junto con la **latencia de las operaciones de HDFS**, son cruciales para detectar posibles fallas en el sistema de archivos distribuido. Además, las **métricas de YARN**, como el uso de recursos y las colas de trabajos, ayudan a monitorear el rendimiento de los trabajos en el clúster. Estas métricas permiten predecir y prevenir posibles problemas de rendimiento, lo que a su vez reduce tiempos de inactividad y asegura un uso eficiente de los recursos.

En cuanto a **JMX Exporter**, se puede mejorar su configuración ajustando las métricas que se recopilan, asegurándose de que se incluyan métricas más específicas y relevantes para el sistema. Por ejemplo, se podrían personalizar los yml para que recopilen métricas más detalladas sobre la **carga de trabajo** o el **rendimiento de los jobs en YARN**, lo que ayuda a proporcionar información más precisa sobre el comportamiento y las áreas de mejora dentro del clúster. También, incluir métricas específicas de la configuración de red, memoria o bloqueos de disco en HDFS podría ayudar a identificar problemas de manera más granular y permitir una gestión más eficiente de los recursos.

Utilizar herramientas como **Prometheus** y **Grafana** para monitoreo tiene varias ventajas frente a otras soluciones. **Prometheus** es ideal para la recolección y almacenamiento de métricas a gran escala debido a su capacidad de manejar métricas en tiempo real y su flexibilidad al integrarse con diferentes sistemas. Además, su capacidad para realizar consultas avanzadas mediante **PromQL** lo convierte en una opción poderosa para la supervisión detallada de los clústeres. **Grafana**, por su parte, es una de las mejores herramientas de visualización que permite crear paneles de control intuitivos y personalizables, lo cual facilita la interpretación de las métricas recolectadas. Juntas, **Prometheus** y **Grafana** ofrecen una solución robusta y fácil de usar para monitorear clústeres como EMR, proporcionando visibilidad completa sobre el estado del sistema, lo que es esencial para mantener un rendimiento óptimo y detectar problemas de forma temprana.