# INSTALACIÓN KAFKA

En Kafka hay tres tipos de clústeres:

* **Un nodo con un broker:**
  + Este es el tipo más sencillo de clúster Kafka y consiste en una única instancia de Kafka corriendo en un servidor, a la cual se le llama broker.
  + Es útil para el desarrollo, pruebas o entornos de baja escala donde la alta disponibilidad y la tolerancia a fallos no son una preocupación.
  + Aquí, todas las particiones de todos los temas se almacenan en este único broker.
* **Un nodo con varios brokers:**
  + Aunque físicamente es un solo servidor o nodo, se configura para ejecutar múltiples instancias de Kafka como brokers separados.
  + Esto puede utilizarse para pruebas más avanzadas, como la partición de temas y el balanceo de carga entre brokers dentro del mismo nodo.
  + Sin embargo, al estar todos los brokers en el mismo nodo físico, no se obtiene la verdadera tolerancia a fallos o alta disponibilidad, ya que la falla del nodo afectará a todos los brokers.
* **Varios nodos con múltiples brokers:**
  + Este es un clúster Kafka distribuido y es el escenario de producción estándar para Kafka.
  + Consiste en múltiples nodos (servidores físicos o instancias de máquina virtual), cada uno ejecutando una instancia de un broker.
  + Los temas se dividen en particiones y estas particiones se distribuyen y replican a través de los diferentes brokers en diferentes nodos.
  + Esto proporciona alta disponibilidad y tolerancia a fallos, ya que si un broker o nodo falla, las particiones replicadas en otros brokers pueden continuar sirviendo los datos.
  + También permite la escalabilidad horizontal, agregando más nodos al clúster para aumentar la capacidad y el rendimiento.

## CLÚSTER KAFKA EN UBUNTU

Vamos a crear un clúster Kafka con 3 nodos y 3 brokers.

Crear un clúster de Kafka con 3 nodos y 3 brokers involucra varios pasos, incluyendo la configuración de Zookeeper (que Kafka utiliza para el manejo de clústeres y la coordinación) y la configuración de los brokers de Kafka.

Vamos a partir de tres máquinas con Ubuntu instalado y con la red correctamente configurada (tienen salida a Internet y se ven entre ellas).

* nodo1 - 192.0.2.5
* nodo2 - 192.0.2.10
* nodo3 - 192.0.2.20

### INSTALACIÓN Y CONFIGURACIÓN

Configurar **etc/hostname** **/** con el nombre del equipo en todos los equipos:

* nodo1
* nodo2
* nodo3

Configurar **/etc/hosts** con los nombres de todos los equipos del clúster en todos los equipos:

***sudo nano /etc/hosts***

***/etc/hosts***

192.0.2.5 nodo1

192.0.2.10 nodo2

192.0.2.20 nodo3

Instalar **ssh** y **java** en todos los equipos:

***sudo apt update***

***sudo apt install openssh-server***

***sudo apt install -y openjdk-21-jdk***

**Descargar e instalar Kafka** en todos los equipos: puedes descargar la última versión de Kafka desde el sitio web oficial de Apache Kafka (<https://kafka.apache.org/downloads>). Selecciona el archivo binario que corresponda a la última versión estable, puedes usar wget para descargarlo.

***wget https://dlcdn.apache.org/kafka/3.9.0/kafka\_2.13-3.9.0.tgz***

Descomprimir Kafka en una ubicación adecuada en todos los equipos:

***tar -xzf kafka\_2.13-3.9.0.tgz***

***sudo mv kafka\_2.13-3.9.0 /opt/kafka***

***sudo chown -R iabd:iabd /opt/kafka***

Creamos una carpeta para los datos en todos los equipos.

***sudo mkdir /datos***

***sudo chown -R iabd:iabd /datos***

**Configurar Zookeeper** en todos los equipos: Kafka usa Zookeeper para gestionar el clúster, por lo que necesitamos configurar Zookeeper en los nodos, para lo que hay que añadir o modificar las siguientes propiedades.

***nano /opt/kafka/config/zookeeper.properties***

***zookeeper.properties***

dataDir=/datos/zookeeper

server.101=192.0.2.5:2888:3888

server.102=192.0.2.10:2888:3888

server.103=192.0.2.20:2888:3888

# the number of ticks that the initial synchronization phase can take

initLimit=5

# the number of ticks that can pass between sending a request and getting an acknowledgement

syncLimit=2

dataDir: indica donde Zookeeper almacenará sus archivos

**Inicializar el ID de Zookeeper** en todos los equipos: en cada nodo, crear un archivo llamado myid en el directorio dataDir de Zookeeper que contenga un único dígito, que será el ID de ese nodo (1, 2 o 3):

nodo1:

***mkdir /datos/zookeeper***

***echo "101" > /datos/zookeeper/myid***

nodo2:

***mkdir /datos/zookeeper***

***echo "102" > /datos/zookeeper/myid***

nodo3

***mkdir /datos/zookeeper***

***echo "103" > /datos/zookeeper/myid***

**Iniciar Zookeeper** en todos los equipos:

***/opt/kafka/bin/zookeeper-server-start.sh /opt/kafka/config/zookeeper.properties***

Si queremos comprobar que zookeeper está funcionando abrir otra terminal cuando lo hayamos arrancado en todos los equipos y ejecutar el siguiente comando:

*lsof -i :2181*

Abrir otra terminal para seguir trabajando (si no la hemos abierto ya para la comprobación) y **configurar los brokers** de Kafka en todos los equipos: hay que añadir o modificar las siguientes propiedades.

***nano /opt/kafka/config/server.properties***

nodo1:

***server.properties***

broker.id=101

zookeeper.connect=192.0.2.5:2181,192.0.2.10:2181,192.0.2.20:2181

listeners=PLAINTEXT://:9092

nodo2:

***server.properties***

broker.id=102

zookeeper.connect=192.0.2.5:2181,192.0.2.10:2181,192.0.2.20:2181

listeners=PLAINTEXT://:9092

nodo3:

***server.properties***

broker.id=103

zookeeper.connect=192.0.2.5:2181,192.0.2.10:2181,192.0.2.20:2181

listeners=PLAINTEXT://:9092

broker.id: asignar a cada broker un identificador único.

zookeeper.connect: indica a Kafka cómo conectarse a Zookeeper.

listeners: permite que los brokers se comuniquen entre sí y con los productores/consumidores:

**Iniciar el broker de Kafka** en todos los equipos:

***/opt/kafka/bin/kafka-server-start.sh /opt/kafka/config/server.properties***

Para seguir trabajando en los equipos tendremos que abrir otra terminal.

### VERIFICAR CLÚSTER

Para verificar que tu clúster está funcionando correctamente, podemos crear un tema de prueba y listar todos los temas para asegurarnos de que se replica en todos los nodos.

Crear un tema en uno de los brokers:

***/opt/kafka/bin/kafka-topics.sh --create --topic test-topic --bootstrap-server localhost:9092 --replication-factor 3 --partitions 1***

Listar los temas en todos los brokers:

***/opt/kafka/bin/kafka-topics.sh --list --bootstrap-server localhost:9092***

En vez de localhost puedes poner cualquiera de los nodos (nodo1, nodo2 o nodo3), una vez conectados a uno nos conectamos a todo el clúster.