BigData

LABORATORIO BIDDATA

CONFIGURACIÓN HADOOP YARN

ALCANCE

- Comprensión de los conceptos básicos de configuración de YARN y HDFS.
- Configurar los recursos del cluster, como la cantidad de memoria y CPU disponibles para cada nodo
- Subir aplicación en el cluster y ejecutar trabajos de MapReduce, monitorear el proceso.

INTRODUCCIÓN:

En este laboratorio, se pone en práctica los conceptos teóricos de YARN (Yet Another Resource Negotiator), el sistema de gestión de recursos de Hadoop que permite la ejecución de aplicaciones distribuidas en un clúster de Hadoop de manera **eficiente** y **escalable**. YARN permite la ejecución simultánea de múltiples aplicaciones, lo que mejora significativamente el rendimiento y la eficiencia del procesamiento de grandes conjuntos de datos distribuidos.

El archivo yarn-site.xml es un archivo de configuración utilizado por el sistema de gestión de recursos de Hadoop, YARN (Yet Another Resource Negotiator), para definir varias propiedades relacionadas con el funcionamiento de YARN en un clúster de Hadoop. Algunas de las propiedades que se pueden definir en el archivo yarn-site.xml incluyen:

- Cantidad de memoria que se debe asignar a cada contenedor de aplicación que se ejecuta en el clúster.
  Cantidad máxima de memoria que se puede utilizar por nodo en el clúster.
- Cantidad de núcleos de CPU que se pueden asignar a cada contenedor de aplicación.
- Cantidad máxima de núcleos de CPU que se pueden utilizar por nodo en el clúster.
- La dirección de red del ResourceManager y los nodos nodemanager en el clúster.
- La ubicación del archivo de registro de YARN.

El archivo yarn-site.xml se encuentra en el directorio de configuración de Hadoop en cada nodo del clúster y como en este laboratorio se usa los contenedores, los archivos de confracción se enchutar en el directorio **config**. Para conocer la configuración básica conocer los parámetros puede consultar:

https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-common/yarn-default.xml .

RECURSO A UTILIZAR.

1. **Clúster de haddop en docker:** Docker es una plataforma de contenedores que permite empaquetar una aplicación y sus dependencias en un contenedor aislado, que se puede ejecutar en cualquier entorno sin cambios. Para este laboratorio, Docker será utilizado para simplificar el proceso de configuración y despliegue del clúster de Hadoop. En el git del curso se cuenta con la imagen Docker que contiene todas las dependencias necesarias para ejecutar Hadoop y de esta manera lanzar múltiples contenedores (nodos) y crear el clúster

de Hadoop. Lo importante es que este proceso permite la creación y la gestión de un clúster de Hadoop, y la misma configuración del clúster puede ser utilizada en diferentes entornos, sin necesidad de instalar y configurar todas las dependencias de Hadoop manualmente en cada equipo.

2. **La Interfaz web de YARN**: YARN proporciona una interfaz web que permite monitorear el estado de los recursos del clúster, las aplicaciones que se están ejecutando y las colas de aplicaciones. La interfaz web de YARN también proporciona gráficos y estadísticas sobre el uso de recursos del clúster. Una vez, tenga el cluster configurado puede ingresar a la interfaz web a localhost:8088
3. **La interfaz web de HDFS:** HDFS proporciona una interfaz gráfica para acceder y administrar el sistema de archivos distribuido de Hadoop. Esta interfaz se ejecuta en un servidor web que se ejecuta en el nodo NameNode del clúster de Hadoop, para el laboratorio se puede acceder en el enlace localhost:9870. A través de la interfaz web, se pueden realizar diversas tareas, como: ver estado del clúster de Hadoop, el número de bloques de datos y otros detalles relevantes del sistema, ver los detalles de los archivos y directorios almacenados en el clúster, cargar o descargar archivos en el sistema de archivos distribuido de Hadoop.
4. **Los comandos de YARN por CLI:** al igual que HDFS, YARN proporciona una serie de comandos en CLI que permiten monitorear el estado de las aplicaciones, los recursos del clúster y las colas de aplicaciones. Estos comandos incluyen yarn application, yarn node, yarn queue, yarn rmadmin, yarn logs, entre otros. Ver más información en la web de hadoop: https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YarnCommands.html
5. **Archivos de configuración XML:** los archivos XML en HDFS y YARN se utilizan para configurar los diferentes componentes de los sistemas de clúster en Hadoop y definir cómo deben operar. En HDFS, el archivo XML principal es el archivo de configuración **hdfs-site.xml**, que define cómo se deben realizar operaciones como el almacenamiento y la recuperación de datos en el clúster Hadoop. Este archivo se utiliza para configurar parámetros como la ubicación de los bloques de datos, el tamaño de los bloques, el número de réplicas de los bloques, etc. https://hadoop.apache.org/docs/r2.4.1/hadoop-project-dist/hadoop-hdfs/hdfs-default.xml

   En YARN, el archivo XML principal es el archivo de configuración **yarn-site.xml**, que se utiliza para configurar parámetros relacionados con la gestión de recursos del clúster, como la cantidad de memoria y CPU que se asignan a las aplicaciones que se ejecutan en el clúster. Este archivo también se utiliza para definir las políticas de cola y el número de nodos que se utilizan para ejecutar las aplicaciones. https://hadoop.apache.org/docs/r2.7.3/hadoop-yarn/hadoop-yarn-common/yarn-default.xml

CONFIGURACIÓN INICIAL:

1. Obtener la imagen de docker, debe ingresar al Git del curso y descarga el docker_hadoop_UR: https://github.com/labDatascience/docker_hadoop_UR, recuerde analizar el analizar el fichero Dockerfile y el fichero docker-compose.yml
2. Antes de iniciar el cluster, revise la carpeta **config** que se encuentra al interior de haddop_UR y comprenda cada parametrización de los archivos XML

a. En **hdfs-site.xml**, configuración de tamaño de bloque, con la propiedad **dfs.blocksize,** a 16M, configure el número de réplicas a 3 con el parámetro **dfs.replication**
b. En **yarn-site.xml**, habilitar la auto detección de memoria y CPU del nodo, configurar a 3 el número máximo de contenedores que se pueden ejecutar en un nodo.Como referencia puede usar las siguientes propiedades
yarn.nodemanager.resource.memory-mb: Cantidad total de memoria que se puede asignar a los contenedores de YARN en un nodo.
yarn.nodemanager.resource.cpu-vcores: Número de núcleos de CPU que se pueden asignar a los contenedores de YARN en un nodo.
yarn.nodemanager.resource.detect-hardware-capabilities: Habilitar la detección automática de memoria y CPU en un nodo.
c. Especificar la política de planificación que utilizara el clúster en modo "CapacityScheduler". Revise la propiedad:
yarn.resourcemanager.scheduler.class: especifica la clase a utilizar como planificador de recursos (resource scheduler) en el ResourceManager de Hadoop YARN
d. indague como se puede configurar la planificación por capacidad con dos pilas, que consuman cada una el 50% de recursos. Revise el archivo **capacity-scheduler.xml,** que lo encuentra en los nodos Hadoop

3. Ejecute el cluster de hadoop en Docker , recuerde que puede usar el comando:

docker compose up –d

4. En la interfaz web de HDFS localhost:9870 y de YARN localhost:8088 comprobar la configuración de los puntos 1 y 2.

DESARROLLO:

Test de la configuración:

1. Con el DataSet construido la evaluación use el script y los libros dados en la carpeta resultados, cargue los libros a HDFS, e indique cuantos bloques se realizaron por libro, y porque esa cantidad ¿Por qué esa cantidad, es lo que se esperaba? ¿Genero algún error, como se soluciona?
    a. Si lo hace por CLI, recuerde activar el nodo hadoop, poner la información en la carpeta data. Luego desde el nodo hadoop recuerde que puede usar:

hadoop fs –copyToLocal <hdfsfile> <localdestination>

    b. Si lo hace via grafica usar la web de HDFS

2. Cargue el trabajo de contar palabras de mapreduce. ¿El resultado es el esperado?. Revise la carpeta de salida, cuantos archivos part-, ¿encuentra?, si hay más de un archivo part-, ¿alguna palabra duplicada entre estos archivos 'parciales'?
3. ¿Cuántos Reduce han intervenido en el proceso?, puedes explicar el resultado

BigData

Escalado del cluster

1. Aumente el número de nodos worker a 6 para ello en el bash ejecutar:

   docker compose up --scale datanode=6 -d datanode

2. Revise en el cliente web, que la configuración de los nodos está acorde a la configuración anterior.
3. Ejecute de nuevo el **análisis de palabras** del parcial pero con todos los libros del curso "Libros.txt", y los 6 nodos del cluster. Repita los puntos 2 y 3 del ítem anterior

Construcción de DataSet propio

1. Construya usted mismo un problema de Bigdata, con un dataSet. Para ello siéntase libre de buscar o construir un DataSet público, y de generar la pregunta a resolver. A partir de ello, configure hadoop según sus consideraciones para garantizar el rendimiento e intente resolver la problemática planteada con Mapreduce