Laboratorio 24/01/2020

0. habilitar eth0 y cambiar hostname: nmtui

1. Actualizar la maquina: yum update -y

2. Instalar Java JDK: yum install java-1.8.0-openjdk -y

2.1. Verificar la version de Java: java -version

2.2. Obtener el path: update-alternatives --config java

2.3. Actualizar el Bash Profile: vim .bash\_profile

2.4. Agregar la ruta para el JAVA\_HOME: export JAVA\_HOME=<PATH> --sin el bin/java

export JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.232.b09-0.el7\_7.x86\_64/jre

2.5. Actualizar las variables de sistema: source .bash\_profile

2.6. Probar el path: echo $JAVA\_HOME

2.7. Instalar wget y git: yum install wget git -y

2.8. Acceso SSH paswordless

$ ssh localhost

$ ssh-keygen -t rsa -P '' -f ~/.ssh/id\_rsa

$ cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

$ chmod 0600 ~/.ssh/authorized\_keys

2.9. Revisar el estado del firewall

systemctl status firewalld

systemctl stop firewalld

systemctl start firewalld

Laboratorio 1: HDFS stand alone

1. Descargar la versionde HDFS desde uno de los mirrors: wget https://www-us.apache.org/dist/hadoop/common/hadoop-3.2.1/hadoop-3.2.1.tar.gz

2. Descomprimir el archivo: tar -xvf hadoop-3.2.1.tar.gz

3. Revisar la instalacion:

> cd hadoop-3.2.1

> ls -l

Ejercicio 1: conteo de palabras

$ mkdir input

$ cp etc/hadoop/\*.xml input

$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar grep input output 'dfs[a-z.]+'

$ cat output/\*

Ejercicio 2: Operacion Semi distribuida

2.1. Modificar los archivos de configuracion del Core: etc/hadoop/core-site.xml

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

</configuration>

2.2. Modificar los archivos de configuracion de HDFS: etc/hadoop/hdfs-site.xml

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

</configuration>

2.3. Modificar las variablas de entorno: etc/hadoop/hadoop-env.sh

JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.232.b09-0.el7\_7.x86\_64/jre

export HDFS\_NAMENODE\_USER=root

export HDFS\_DATANODE\_USER=root

export HDFS\_SECONDARYNAMENODE\_USER=root

export YARN\_RESOURCEMANAGER\_USER=root

export YARN\_NODEMANAGER\_USER=root

2.4. Limpiar el ambiente para dejar listo storage

$ cd tmp

$ rm -rf \*

2.5. Preparar el filesystem

$ bin/hdfs namenode -format

2.6. Iniciar los servicios de NameNode y DataNode

$ sbin/start-dfs.sh

2.7. Probar el acceso al URL del NameNode

$ curl http://localhost:9870/

2.8. Hacer port forwarding para que la VM exponga el puerto

settings --> network --> advanced --> port forwarding

2.9. crear la carpeta donde se van a dejar los archivos y copiar los archivos

$ bin/hdfs dfs -mkdir /user

$ bin/hdfs dfs -mkdir /user/root

$ bin/hdfs dfs -mkdir input

$ bin/hdfs dfs -put etc/hadoop/\*.xml input

2.10. Ejecutar el job de conteo

$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar grep input output 'dfs[a-z.]+'

2.11. Descargar el output

$ bin/hdfs dfs -get output output

$ cat output/\*

2.12. Revisar el output desde el hdfs

$ bin/hdfs dfs -cat output/\*

2.13. Detener los servicios de NameNode y DataNode

$ sbin/stop-dfs.sh

Ejercicio 3: Habilitar el YARN

3.1. Modificar la configuracion del servicio de MapReduce: etc/hadoop/mapred-site.xml

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

<property>

<name>mapreduce.application.classpath</name>

<value>$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/\*:$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/lib/\*</value>

</property>

</configuration>

3.2. Modificar la configuracion del servicio de YARN: etc/hadoop/yarn-site.xml

<configuration>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.env-whitelist</name>

<value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>

</property>

</configuration>

3.3. Limpiar el ambiente para dejar listo storage

$ cd tmp

$ rm -rf \*

3.4. Preparar el filesystem

$ bin/hdfs namenode -format

3.5. Iniciar los servicios de NameNode y DataNode

$ sbin/start-dfs.sh

3.6. Iniciar los servicios de ResourceManager y NodeManager

$ sbin/start-yarn.sh

3.7. Probar el acceso al URL del ResourceManager

$ curl http://localhost:8088/

3.8. Hacer port forwarding para que la VM exponga el puerto

settings --> network --> advanced --> port forwarding

3.9. crear la carpeta donde se van a dejar los archivos y copiar los archivos

$ bin/hdfs dfs -mkdir /user

$ bin/hdfs dfs -mkdir /user/root

$ bin/hdfs dfs -mkdir input

$ bin/hdfs dfs -put etc/hadoop/\*.xml input

3.10. Ejecutar el job de conteo

$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar grep input output 'dfs[a-z.]+'

3.11. Descargar el output

$ bin/hdfs dfs -get output output

$ cat output/\*

3.12. Revisar el output desde el hdfs

$ bin/hdfs dfs -cat output/\*

3.13. Detener todos los servicios

$ sbin/stop-dfs.sh

$ sbin/stop-yarn.sh