为虚拟机新建立一个hadoop对象: sudo useradd -m hadoop -s /bin/bash

为新用户设置登陆密码: sudo passwd hadoop

为新用户添加管理员权限: sudo adduser hadoop sudo

登录hadoop 用户后 更新系统软件：

更新软件列表：sudo apt-get update

更新软件：sudo apt-get upgrade

安装ssh-server：sudo apt-get install openssh-server

配置 ssh 无密登录：

进入ssh目录：cd ~/.ssh/

生成ssh密匙：ssh-keygen -t rsa （提示默认按回车即可）

将密匙加入授权中：cat ./id\_rsa.pub >> ./authorized\_keys

此后可以直接使用 ssh localhost 免密登录

安装 jdk8：sudo apt-get install openjdk-8-jdk

查看java是否安装成功：java -version

配置java环境：gedit ~/.bashrc

在最前面加入：export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

使环境变量生效：source ~/.bashrc

使用：java -version 查看jdk的版本 whereis java 查看jdk的路径

下来进行 Hadoop、Hbase的安装 均将安装包移动到 /usr/local/ 路径下进行解压

并改名为 hadoop、hbase

1. Hadoop

解压文件到 /usr/local/ 目录下 改名为 hadoop

然后修改文件的权限：sudo chown -R hadoop ./hadoop (hbase)

（将文件置为 hadoop 用户所有）

查看hadoop是否安装成功：

进入目录：cd hadoop

执行指令：./bin/hadoop version

进行Hadoop 伪分布式的配置

gedit ./etc/hadoop/core-site.xml

<configuration>

<property>

<name>hadoop.tmp.dir</name>

<!-- hadoop 存储路径-->

<value>file:/usr/local/hadoop/tmp</value>

<description>A base for other temporary directories.</description>

</property>

<property>

<name>fs.defaultFS</name>

<!-- 告诉 NN 在那个机器，NN 使用哪个端口号接收客户端和 DN 的RPC请求-->

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

</property>

</configuration>

gedit ./etc/hadoop/hdfs-site.xml

<configuration>

<property>

<!-- 设置默认副本数量为3-->

<name>dfs.replication</name>

<value>3</value>

</property>

<property>

<!-- 主namenode 信息存储目录-->

<name>dfs.namenode.name.dir</name>

<value>file:/usr/local/hadoop/tmp/dfs/name</value>

</property>

<property>

<name>dfs.namenode.secondary.http-address</name>

<value>slave1:50090</value>

</property>

<property>

<!--datanode 存储位置-->

<name>dfs.datanode.data.dir</name>

<value>file:/usr/local/hadoop/tmp/dfs/data</value>

</property>

</configuration>

gedit ./etc/hadoop/yarn-site.xml

<configuration>

<property>

<name>yarn.resourcemanager.hostname</name>

<value>master</value>

</property>

<property>

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

<value>mapreduce\_shuffle</value>

</property>

</configuration>

gedit ./etc/hadoop/mapred-site.xml

<configuration>

<property>

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

<value>yarn</value>

</property>

<property>

<name>mapreduce.jobhistory.address</name>

<value>master:10020</value>

</property>

<property>

<name>mapreduce.jobhistory.webapp.address</name>

<value>master:19888</value>

</property>

<property>

<name>yarn.app.mapreduce.am.env</name>

<value>HADOOP\_MAPRED\_HOME=file:/usr/local/hadoop</value>

</property>

<property>

<name>mapreduce.map.env</name>

<value>HADOOP\_MAPRED\_HOME=file:/usr/local/hadoop</value>

</property>

<property>

<name>mapreduce.reduce.env</name>

<value>HADOOP\_MAPRED\_HOME=file:/usr/local/hadoop</value>

</property>

</configuration>

执行namenode格式化：./bin/hdfs namenode -format (当提示输入Y/N时，输入Y)

开启namenode、datanode：./sbin/start-dfs.sh

查看是否开启成功：jps（4个信息JPS,SecondaryNameNode,NameNode,DataNode）

开启yarn：./sbin/start-yarn.sh

查看是否开启成功：jps (新加ResourceManager与NodeManager)

查看网页端的信息：http://localhost:9870

伪分布读取的是HDFS上的数据使用时需要上传数据到HDFS上

在HDFS上创建目录：./bin/hdfs dfs -mkdir -p /data/input （创建了data/input路径的文件夹）

上传文件到该目录下：./bin/hdfs dfs -put text/\* /data/input

查看是否上传成功：./bin/hdfs dfs -ls /data/input

HDFS 退出安全模式：./bin/hdfs dfsadmin -safemode leave

---------------------------------------------------------------------------------------------------------------------------------

③Hbase

解压改名后授权：sudo chown -R hadoop ./hbase

查看是否解压成功：./bin/hbase version

配置Hbase:

# 配置JDK安装路径

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

# 设置HBase的日志目录

export HBASE\_LOG\_DIR=/usr/local/hbase/logs

export HBASE\_CLASSPATH=/usr/local/hadoop/etc/hadoop

export HBASE\_PID\_DIR=/usr/local/hbase/pids

# 使用独立的ZooKeeper集群

export HBASE\_MANAGES\_ZK=false

gedit ./conf/hbase-site.xml

<configuration>

<property>

<!-- 同hadoop配置文件core-site.xml中fs.defaultFS的值 -->

<name>hbase.root.dir</name>

<value>hdfs://master:9000/hbase</value>

</property>

<property>

<!-- false是单机模式，true是分布式模式 -->

<!-- 当使用外置zookeeper时必须配置为true，仅在使用hbase自带zookeeper时配置为false -->

<name>hbase.cluster.distributed</name>

<value>true</value>

</property>

<property>

<!-- zookeeper的数据目录，启动hbase的用户需对此目录有读写权限 -->

<name>hbase.zookeeper.property.dataDir</name>

<value>/usr/local/zookeeper/tmp</value>

</property>

<property>

<name>hbase.zookeeper.quorum</name>

<value>master,slave1,slave2</value>

</property>

<property>

<!-- 指定HBase Master web页面访问端口-->

<name>hbase.master.info.port</name>

<value>60010</value>

</property>

<property>

<!-- 指定HBase RegionServer web页面访问端口 -->

<name>hbase.regionserver.info.port</name>

<value>60030</value>

</property>

<property>

<name>hbase.unsafe.stream.capability.enforce</name>

<value>false</value>

</property>

</configuration>

启动hbase：./bin/start-hbase.sh

查看是否启动成功：jps （多三个结点HRegionServer、HQuorumPeer、HMaster）

关闭hbase：./bin/stop-hbase.sh

打开shell：./bin/hbase shell