title: Ubuntu20.04安装Hadoop和Hive(PD虚拟机 MAC M1)

author: dinggc

tags:

- hadoop

categories:

- 大数据

date: 2021-12-01 15:09:30

Ubuntu20.04安装Hadoop和Hive(PD虚拟机 MAC M1)

Ubuntu20.04安装mysql

安装mysql

```
#更新源
sudo apt-get update
#安装mysql服务
sudo apt-get install mysql-server
```

配置mysql

```
sudo mysql_secure_installation
```

```
VALIDATE PASSWORD PLUGIN can be used to test passwords...
Press y|Y for Yes, any other key for No: N (选择N ,不会进行密码的强校验)

Please set the password for root here...
New password: (输入密码)
Re-enter new password: (重复输入)

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them...
Remove anonymous users? (Press y|Y for Yes, any other key for No) : N
(选择N，不删除匿名用户)

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network...
Disallow root login remotely? (Press y|Y for Yes, any other key for No)
: N (选择N，允许root远程连接)

By default, MySQL comes with a database named 'test' that
anyone can access...
Remove test database and access to it? (Press y|Y for Yes, any other key
for No) : N (选择N，不删除test数据库)

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.
Reload privilege tables now? (Press y|Y for Yes, any other key for No) :
Y (选择Y，修改权限立即生效)
```

配置远程访问

```
sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf #找到bind-address 修改值为0.0.0.0
sudo /etc/init.d/mysql restart #重启mysql
```

刷新权限

远程连接(navicat)

ip地址为10.211.55.4

ubuntu20.04安装Hadoop

安装jdk1.8

官方链接

```
mkdir dingdm                                    #创建dingdm文件夹
```

上传jdk文件到pd虚拟机

```
sudo tar zxvf jdk-8u311-linux-aarch64.tar.gz  -C /dingdm  #/解压到/dingdm目录下
mv  jdk1.8.0_311 java                                    #重命名为java
vi ~/.bashrc                                                    #给JDK配置环境变量
```

在 ~/.bashrc 文件中添加如下代码

```
export JAVA_HOME=/dingdm/java
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
```

```
source ~/.bashrc                                #使新配置的环境变量生效
```

安装hadoop

官方链接

上传hadoop文件到pd虚拟机

配置环境变量

```
sudo tar -zxvf  hadoop-3.2.0.tar.gz                    #解压
sudo mv  hadoop-3.2.0    hadoop                       #重命名为hadoop，可改可不改，如果修改下边的名字也要对应
```

```
export HADOOP_HOME=/dingdm/hadoop
export CLASSPATH=$($HADOOP_HOME/bin/hadoop classpath):$CLASSPATH
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
```

```
source ~/.bashrc
```

伪分布式配置

```
cd /dingdm/hadoop/etc/hadoop/
```

hadoop-env.sh中配置java环境

```
export JAVA_HOME=/dingdm/java
```

修改core-site.xml与hdfs-site.xml文件，在configuration下添加property标签配置

```
\*<property>
    <name>hadoop.tmp.dir</name>
    <value>file:/dingdm/hadoop/tmp</value>
    <description>Abase for other temporary directories.</description>
</property>
<property>
    <name>fs.defaultFS</name>
    <value>hdfs://localhost:9000</value>
</property>
```

```
 <property>
     <name>dfs.replication</name>
     <value>1</value>
</property>
<property>
     <name>dfs.namenode.name.dir</name>
     <value>file:/dingdm/hadoop/tmp/dfs/name</value>
</property>
<property>
     <name>dfs.datanode.data.dir</name>
     <value>file:/dingdm/hadoop/tmp/dfs/data</value>
</property>
<property>
     <name>dfs.http.address</name>
     <value>0.0.0.0:50070</value>
</property>
```

执行 NameNode 的格式化

```
cd /dingdm/hadoop/bin
hdfs namenode -format
```

启动hadoop

```
cd /dingdm/hadoop/sbin
start-dfs.sh
jps
```

ubuntu20.04安装Hive

文件上传

配置环境变量

```
export HIVE_HOME=/dingdm/hive
export PATH=$PATH:$HIVE_HOME/bin
```

```
source ~/.bashrc
```

Hive配置文件

指定 Hive 数据仓库的数据存储在 HDFS 上的目录

```
sudo mkdir -p /user/hive/warehouse
sudo mkdir -p /tmp/hive
sudo chmod -R 777 /user/hive/warehouse
sudo chmod -R 777 /tmp/hive
```

在/dingdm/hive/conf目录下，创建hive-site.xml，并添加以下代码

```
<configuration>
<property>
    <name>hive.metastore.schema.verification</name>
    <value>false</value>
     <description>
```

```
          Enforce metastore schema version consistency.
          True: Verify that version information stored in metastore
matches with one from Hive jars.  Also disable automatic
          schema migration attempt. Users are required to manully
migrate schema after Hive upgrade which ensures
          proper metastore schema migration. (Default)
          False: Warn if the version information stored in metastore
doesn't match with one from in Hive jars.
      </description>
    </property>
    <!-- 存储在hdfs上的数据路径 -->
    <property>
    <name>hive.metastore.warehouse.dir</name>
    <value>/user/hive/warehouse</value>
    <description>location of default database for the
warehouse</description>
  </property>
  <property>
    <name>hive.exec.scratchdir</name>
    <value>/tmp/hive</value>
    <description>HDFS root scratch dir for Hive jobs which gets created
with write all (733) permission. For each connecting user, an HDFS
scratch dir: ${hive.exec.scratchdir}/&lt;username&gt; is created, with
${hive.scratch.dir.permission}.</description>
  </property>
    <!-- 本地mysql -->
  <property>
                <name>javax.jdo.option.ConnectionURL</name>
                <value>jdbc:mysql://localhost:3306/hive?
createDatabaseIfNotExist=true</value>
                <description>JDBC connect string for a JDBC
metastore</description>
                <!-- 如果 mysql 和 hive 在同一个服务器节点，那么请更改
hadoop02 为 localhost -->
        </property>
        <property>
```

```
                <name>javax.jdo.option.ConnectionDriverName</name>
                <value>com.mysql.cj.jdbc.Driver</value>
                <description>Driver class name for a JDBC
    metastore</description>
        </property>
        <property>
                <name>javax.jdo.option.ConnectionUserName</name>
                <value>root</value>
                <description>username to use against metastore
    database</description>
        </property>
        <property>
                <name>javax.jdo.option.ConnectionPassword</name>
                <value>你的数据库密码</value>
        <description>password to use against metastore
    database</description>
        </property>
    </configuration>
```

在/dingdm/hive/conf下执行

```
    cp hive-env.sh.template hive-env.sh
```

添加以下内容

```
    export HADOOP_HOME=/dingdm/hadoop
    export HIVE_CONF_DIR=/dingdm/hive/conf
```

拷贝jdbc的包到/dingdm/hive/lib目录下

在dingdm/hive/bin目录下启动hive

```
schematool -dbType mysql -initSchema
schematool -dbType mysql -info
hive
```