## 1 修改主机名

1 编辑文件：vim /etc/hostname

添加主机名：node01

3 重启机器

reboot

2 添加IP与主机的映射关系(可选)

编辑文件sudo vim /etc/hosts

192.168.37.131 node01

192.168.37.132 node02

192.168.37.133 node03

4 设置静态IP

编辑文件：sudo vim /etc/network/interfaces

auto ens33

iface ens33 inet static

address 192.168.37.131

netmask 255.255.255.0

gateway 192.168.37.2

dns-nameserver 8.8.8.8

重启网络：/etc/init.d/networking restart

## 2 关闭防火墙

关闭防火墙：sudo ufw disable

开启防火墙：sudo ufw enable

查看防火墙状态：sudo ufw status

## 3 免密码登录

执行命令：ssh-keygen -t rsa

将公钥追加到文件authorized\_keys中；

## 4 JDK安装

环境变量生效

Source /etc/profile

配置JAVA\_HOME,CLASSPATH的环境变量

需要重新登录才能生效；

## 5 Hadoop安装

5.1 配置环境变量HADOOP\_HOME

5.2 编辑core-site.xml

<property>

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

<value>file:/home/hduser/cluster/data/tmp</value>

<description>Abase for other temporary directories.</description>

</property>

<property>

<name>fs.defaultFS</name>

<value>hdfs://node01:9000</value>

</property>

5.3 编辑hdfs-site.xml

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

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

<value>file:/home/hduser/cluster/data/dfs/name</value>

</property>

<property>

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

<value>file:/home/hduser/cluster/data/dfs/data</value>

</property>

5.4编辑mapred-site.xml

<property>

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

<value>yarn</value>

</property>

5.5 编辑slaves文件

5.6 编辑hadoop-env.sh

export JAVA\_HOME=/home/hduser/cluster/jdk1.8.0\_91

5.7 编辑yarn-site.xml

<property>

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

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.aux-services.mapreduce\_shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

<property>

<name>yarn.resourcemanager.address</name>

<value>node01:8032</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>node01:8030</value>

</property>

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>node01:8031</value>

</property>

5.8 格式化namenode

hadoop namenode –format

5.9 启动集群 start-all.sh,并查看状态！

查看namenode状态:http://node01:50070

查看resourcemanager状态：<http://node01:8088>

## 6安装sqoop

mysql驱动包copy到$SQOOP\_HOME/lib下：mysql-connector-java-5.1.32-bin.jar

配置环境变量：sqoop-env.sh文件

## 7 Zookeeper安装

解压zookeeper文件，编辑zoo.cfg文件

在目录/home/hduser/cluster/zoodata/创建文件myid

启动zookeeper: ./zkServer.sh start

查看zookeeper状态：./zkserver.sh status

## 8安装Hive

8.1安装mysql,(安装在node01)

sudo apt-get install [mysql](https://www.2cto.com/database/MySQL/)-server

设置root用户密码为12345678

查看mysql状态：service mysql status

启动mysql：service mysql start

关闭mysql: service mysql stop

mysql -u root –p

mysql>CREATE USER 'hive'@'%' IDENTIFIED BY '12345678';

mysql>create database hive;

mysql>GRANT ALL PRIVILEGES ON \*.\* TO 'hive'@'%' WITH GRANT OPTION;

mysql>flush privileges;

8.2 安装hive;(安装在node03)

配置hive环境变量：

将mysql驱动上传到$HIVE\_HOME/lib下面

编辑hive-env.sh

HADOOP\_HOME=/home/hduser/cluster/hadoop-2.6.5

直接解压即可，配置环境环境

<configuration>

<property>

<name>javax.jdo.option.ConnectionURL</name>

<value>jdbc:mysql://node01:3306/hive?createDatabaseIfNotExist=true</value>

</property>

<property>

<name>javax.jdo.option.ConnectionDriverName</name>

<value>com.mysql.jdbc.Driver</value>

</property>

<property>

<name>javax.jdo.option.ConnectionUserName</name>

<value>hive</value>

</property>

<property>

<name>javax.jdo.option.ConnectionPassword</name>

<value>12345678</value>

</property>

</configuration>

将mysql驱动包拷贝到$HIVE\_HOME/lib/下：mysql-connector-java-5.1.32-bin.jar

将$HIVE\_HOME/lib/jline-2.12.jar拷贝到$HADOOP\_HOME/share/hadoop/yarn/lib下面，原有的删除！

## 9 安装HBASE

9.1编辑hbase-env.sh

export JAVA\_HOME=/home/hduser/cluster/jdk1.8.0\_91

export HBASE\_MANAGES\_ZK=false

9.2编辑hbase-site.xml

<property>

<name>hbase.rootdir</name>

<value>hdfs://node01:9000/hbase</value>

</property>

<property>

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

<value>node01,node02,node03</value>

</property>

<property>

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

<value>true</value>

</property>

9.3编辑regionservers

node01

node02

node03

测试：

hbase(main):001:0> create 't\_user','f1'

hbase(main):002:0> put 't\_user','001','f1:name','张三'

hbase(main):003:0> put 't\_user','001','f1:age','13'

hbase(main):004:0> desc 't\_user'

hbase(main):005:0> scan 't\_user'

## 10 mysql安装

解压到指定目录：D:\mysql\mysql-5.6.40-winx64

10.1: 配置文件：my.ini

[mysql]

# 设置mysql客户端默认字符集

default-character-set=utf8

[mysqld]

#设置3306端口

port = 3306

# 设置mysql的安装目录

basedir=D:\mysql\mysql-5.6.40-winx64

# 设置mysql数据库的数据的存放目录

datadir=D:\mysql\mysql-5.6.40-winx64\data

# 允许最大连接数

max\_connections=200

# 服务端使用的字符集默认为8比特编码的latin1字符集

character-set-server=utf8

# 创建新表时将使用的默认存储引擎

default-storage-engine=INNODB

10.2配置环境变量

以管理员身份启动cmd;

输入：mysqld install

启动MySQL服务：net start mysql

登录：mysql -uroot –p

创建数据库:create database sqoop;

创建用户并授权：grant all privileges on \*.\* to ‘sqoop’@’%’ identified by’12345678’;

grant all privileges on \*.\* to ‘sqoop’@’localhost’ identified by ‘12345678’;

alter database sqoop character set latin1;

flush privileges

mysql –usqoop –p12345678

## 11 ubuntu下安装mysql

/etc/apt/sources.list;镜像文件

安装Mysql;

sudo apt-get install mysql-server

sudo apt install mysql-client

sudo apt install libmysqlclient-dev

安装成功后可以通过下面的命令测试是否安装成功。

sudo netstat -tap | grep mysql

可以通过如下命令进入Mysql服务：

>mysql –uroot –p你的密码（12345678）

现在设置mysql允许远程访问，编辑文件：

sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf

注释掉：bind-address = 127.0.0.1

启动MYSQL：service mysql start

登录：mysql -uroot –p12345678

创建数据库:create database hive;

创建用户并授权：grant all privileges on \*.\* to ‘hive’@’%’ identified by’12345678’;

grant all privileges on \*.\* to ‘hive’@’localhost’ identified by ‘12345678’;

alter database hive character set latin1;

flush privileges

mysql –uhive –p12345678

## 12 spark安装与配置

Standalone模式部署：

1. 编辑spark-env.sh文件：

export SPARK\_MASTER\_IP=node01

export SPARK\_MASTER\_PORT=7077

export SPARK\_EXECUTOR\_INSTANCES=2

export SPARK\_WORKER\_INSTANCES=1

export SPARK\_WORKER\_CORES=1

export SPARK\_WORKER\_MEMORY=256M

export SPARK\_MASTER\_WEBUI\_PORT=8080

export SPARK\_CONF\_DIR=/home/hduser/cluster/spark-2.3.3-bin-hadoop2.6/conf

export JAVA\_HOME=/home/hduser/cluster/jdk1.8.0\_91

export JRE\_HOME=${JAVA\_HOME}/jre

1. 编辑slaves文件，将node01,node02,node03填入到该文件中。
2. Start-all.sh启动集群。
3. 编辑：vim /etc/profile

#spark

export SPARK\_HOME=/home/hduser/cluster/spark-2.3.3-bin-hadoop2.6

export PATH=$PATH:$SPARK\_HOME/bin

export PYTHONPATH=$SPARK\_HOME/python:$SPARK\_HOME/python/lib/py4j-0.10.7-src.zip:$PYTHONPATH

export PATH=$SPARK\_HOME/python:$PATH

保存退出：使生效：source /etc/profile

1. conda deactivate 或者配置在vim ~/.bashrc文件中。
2. Anaconda3-2019.03-Linux-x86\_64.sh 执行该脚本，安装anaconda；

配置远程登录服务器上的jupyter notebook

6-1>安装jupyter notebook

安装jupyter notebook,安装Anaconda,就已经自动安装了jupyter notebook

6-2> 生成密码

进入shell:安装ipython

$ipython

from Ipython.lib import passwd

passed()

Enter password:#输入远程登录时的密码

Verify passwod:#确认密码

Out：’sha1:.......’

6-3>生成mycert.pem

$ openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyout mycert.pem -out mycert.pem

填写相关信息，生成mycert.pem文件

6-4.配置jupyter notebook

生成配置文件：$ jupyter-notebook --generate-config

修改~/.jupyter/jupyter\_notebook\_config.py文件：

c.NotebookApp.password = u'sha1:' #刚刚生成的SHA密钥

c.NotebookApp.certfile = u'/home/hduser/.jupyter/mycert.pem' #mycert.pem文件位置

c.NotebookApp.ip = 'node01' #本机ip

c.NotebookApp.port = 9999 #端口

c.NotebookApp.allow\_remote\_access=True

6-5>在服务器输入

$ jupyter notebook --ip=Master --no-browser --allow-root

之后会打印访问链接，输入到本地浏览器打开即可

6-6> Jupyter连接pyspark

在服务器端，vim /etc/profile文件中添加变量：

#spark

export SPARK\_HOME=/home/hduser/cluster/spark-2.3.3-bin-hadoop2.6

export PATH=$PATH:$SPARK\_HOME/bin

export PYTHONPATH=$SPARK\_HOME/python:$SPARK\_HOME/python/lib/py4j-0.10.7-src.zip:

$PYTHONPATH

export PATH=$SPARK\_HOME/python:$PATH

#py-spark

export PYSPARK\_PYTHON=python3 #使用python3

export PYSPARK\_DRIVER\_PYTHON=jupyter

export PYSPARK\_DRIVER\_PYTHON\_OPTS="notebook --ip=node01 --no-browser --allow-root"

export PATH=$PATH:/home/hduser/anaconda3/bin

6-7>配置/etc/hosts文件：

192.168.80.131 node01

192.168.80.132 node02

192.168.80.133 node03

127.0.0.1 localhost

6-8>执行命令：启动pyspark