实验环境

centos6.10

jdk1.8

hadoop-2.7.2

apache-hive-1.2.1-bin

mysql5.7

sqoop-1.4.6.bin\_\_hadoop-2.0.4-alpha

安装包下载地址：

链接: https://pan.baidu.com/s/1QzZCnigOkFCqD1XlegrEoA 提取码: 8fcf

centos6.10：

https://mirrors.163.com/centos/6.10/isos/x86\_64/CentOS-6.10-x86\_64-bin-DVD1.iso

以下环境都是在centos6.10系统上进行配置，所有软件都是由用户epidemic安装在/data/soft目录下，

准备工作（root用户登录系统）

1. 创建用户epidemic，并赋予该用户超级管理员权限

useradd epidemic

为epidemic用户添加密码：

passwd epidemic

修改/etc/sudoers文件，添加如下内容：



2. 修改主机名：hadoop100，

修改/etc/sysconfig/network：



3. 修改/ect/host文件（最好配置静态ip），添加一行内容



这里的ip可以使用ifconfig命令查出！

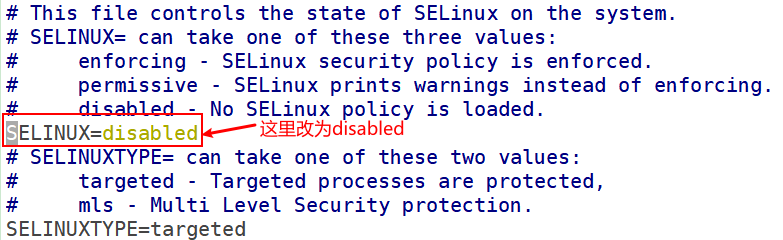
3. 关闭防火墙：

service iptables stop （临时关闭，重启就会恢复原状）

chkconfig iptables off （永久关闭）

4. 关闭 selinux，修改如下内容

vim /etc/selinux/config



5. 为epidemic用户创建文件夹/data/soft

mkdir -p /data/soft

chown -R epidemic:epidemic /data

以上操作完后最好重启电脑reboot

配置Hadoop2.7.2环境

以下采用伪分布式部署hadoop环境

使用epidemic用户登录系统：

1. 首先必须保证当前主机能免密登录ssh

验证：ssh hadoop100，若不要输入密码就能登入就不要进行如下操作

ssh-keygen -r rsa （按4次enter就行）

将生成的公钥复制~/.ssh/ authorized\_keys（应该是不存在的文件）

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

修改authorized\_keys的权限

chmod 0600 ~/.ssh/ authorized\_keys

2. 安装jdk1.8，配置环境变量

在/etc/profile文件添加：

sudo vim /etc/profile

export JAVA\_HOME=/data/soft/jdk1.8.0\_151

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

使修改生效：

source /etc/profile

3. 下载hadoop2.7.2并将其解压到/data/soft目录下，进入hadoop主目录

cd /data/soft/ hadoop-2.7.2

3.1 配置etc/hadoop/hadoop-env.sh，添加jdk的环境：

vim etc/hadoop/hadoop-env.sh



3.2 配置etc/hadoop/core-site.xml，添加以下内容：

vim etc/hadoop/core-site.xml

|  |
| --- |
| <configuration>  <!-- 指定HDFS中NameNode的地址 -->  <property>  <name>fs.defaultFS</name>  <value>hdfs://hadoop100:9000</value>  </property>  <!-- 指定Hadoop运行时产生文件的存储目录 -->  <property>  <name>hadoop.tmp.dir</name>  <value>/data/soft/hadoop-2.7.2/data/tmp</value>  </property>  </configuration> |

3.3 配置etc/hadoop/hdfs-site.xml，添加以下内容：

vim etc/hadoop/hdfs-site.xml

|  |
| --- |
| <configuration>  <!-- 指定HDFS副本的数量 -->  <property>  <name>dfs.replication</name>  <value>1</value>  </property>  </configuration> |

3.4 格式化namenode（只能格式化一次，以后启动都不需要格式化）

bin/hdfs namenode -format

3.5 启动集群

sbin/start-dfs.sh （停止dfs：sbin/stop-dfs.sh）

使用jps查看，出现NameNode，DataNode，SecondaryNameNode三个进程就说明集群启动成功，也可以通过web访问集群资源：

http://192.168.20.100:50070

3.6 配置Yarn集群资源管理

3.6.1 配置etc/hadoop/yarn-env.sh，添加jdk环境：

vim etc/hadoop/yarn-env.sh



3.6.2 配置etc/hadoop/yarn-site.xml，添加如下内容：

vim etc/hadoop/yarn-site.xml

|  |
| --- |
| <configuration>  <!-- Reducer获取数据的方式 -->  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <!-- 指定YARN的ResourceManager的地址 -->  <property>  <name>yarn.resourcemanager.hostname</name>  <value>hadoop100</value>  </property>  </configuration> |

3.6.3 配置etc/hadoop/mapred-env.sh，添加jdk环境

vim etc/hadoop/mapred-env.sh



3.6.4 配置mapred-site.xml：

cp etc/hadoop/mapred-site.xml.template etc/hadoop/mapred-site.xml

vim etc/hadoop/mapred-site.xml

|  |
| --- |
| <configuration>  <!-- 指定MR运行在YARN上 -->  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  </configuration> |

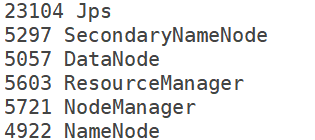
3.6.5 启动yarn

启动前必须保证dfs已经启动

sbin/start-yarn.sh （停止yarn：sbin/stop-yarn.sh）

如果启动成功应该出现如下内容：

jps



YARN的浏览器页面查看

http://hadoop101:8088

最好将Hadoop加入系统环境变量，即在/etc/profile文件中添加如下内容：

export HADOOP\_HOME=/data/soft/hadoop-2.7.2

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

配置hive

官方文档：

https://cwiki.apache.org/confluence/display/Hive/GettingStarted

使用epidemic用户登录系统，解压hive安装包到/data/soft目录下，进入/data/soft/apache-hive-1.2.1-bin

cd /data/soft/apache-hive-1.2.1-bin

1. 配置hive-env.sh文件

cp conf/hive-env.sh.template conf/hive-env.sh

添加如下内容：

vim conf/hive-env.sh

|  |
| --- |
| export HADOOP\_HOME=/data/soft/hadoop-2.7.2  export HIVE\_CONF\_DIR=/data/soft/apache-hive-1.2.1-bin/conf |

2. 启动hive

启动前必须保证hdfs和yarn都已经启动，即

启动顺序：dfs--yarn--hive

关闭顺序：hive--yarn--dfs

在hive主目录下输入bin/hive就能进入hive操作界面，退出输入exit;即可

同样，将hive添加到系统环境变量：

export HIVE\_HOME=/data/soft/apache-hive-1.2.1-bin

export PATH=$PATH:$HIVE/bin

3. 使用mysql储存hive的元数据

使用root用户登入系统

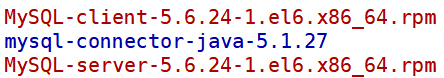
3.1 安装mysql

检查是否安装了mysql

rpm -qa | grep mysql

若安装了可直接从配置允许root用户远程连接mysql开始，以下演示没有安装mysql。解压mysql-libs.zip安装包如下：

unzip mysql-libs.zip



安装MySQL服务：rpm -ivh MySQL-server-5.6.24-1.el6.x86\_64.rpm

安装mysql客户端：rpm -ivh MySQL-client-5.6.24-1.el6.x86\_64.rpm

启动：service mysql start / /etc/init.d/mysql start

修改root用户的密码：

输入mysql登入

mysql> use mysql;

mysql> update user set password=password("你的密码") where user="root" and host="localhost";

mysql> flush privileges;

允许root用户（或者可以创建一个新用户）远程登录：

mysql> update user set host="%" where user="root";

mysql> flush privileges;

启动mysql失败 解决：

https://www.cnblogs.com/weibanggang/p/11230528.html

以上只是可能会遇到的问题，关键看错误信息，一般都能根据错误信息，找到启动日志解决！！！

3.2 在hive中配置mysql

3.2.1 解压mysql-connector-java-5.1.27.tar.gz驱动包，将其目录下的mysql-connector-java-5.1.27-bin.jar到/data/soft/ apache-hive-1.2.1-bin/lib下

3.2.2 在/data/soft/ apache-hive-1.2.1-bin/conf目录下新建hive-site.xml文件，添加如下内容：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" standalone="no"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>javax.jdo.option.ConnectionURL</name>  <value>jdbc:mysql://hadoop100:3306/hive?createDatabaseIfNotExist=true</value>  <description>JDBC connect string for a JDBC metastore</description>  </property>  <property>  <name>javax.jdo.option.ConnectionDriverName</name>  <value>com.mysql.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>root</value>  <description>password to use against metastore database</description>  </property>  <property>  <name>hive.cli.print.header</name>  <value>true</value>  </property>  <property>  <name>hive.cli.print.current.db</name>  <value>true</value>  </property>  </configuration> |

官方参考文档：

https://cwiki.apache.org/confluence/display/Hive/AdminManual+MetastoreAdmin

配置sqoop

1. 解压sqoop安装包到/data/soft目录下，进入sqoop主目录：

cd /data/soft/sqoop-1.4.6.bin\_\_hadoop-2.0.4-alpha

2. 修改配置文件：

Sqoop的配置文件与大多数大数据框架类似，在sqoop根目录下的conf目录中。

cd conf

mv sqoop-env-template.sh sqoop-env.sh

在sqoop-env.sh文件中添加如下内容：

export HADOOP\_COMMON\_HOME=/data/soft/hadoop-2.7.2

export HADOOP\_MAPRED\_HOME=/data/soft/hadoop-2.7.2

export HIVE\_HOME=/data/soft/apache-hive-1.2.1-bin

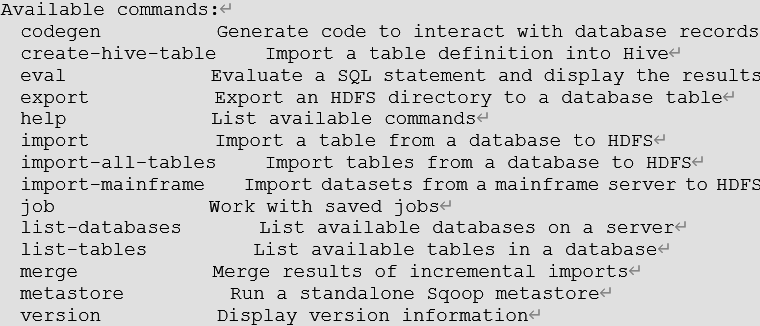
3. 拷贝jdbc驱动到sqoop的lib目录下

cp mysql-connector-java-5.1.27-bin.jar /data/soft/sqoop-1.4.6.bin\_\_hadoop-2.0.4-alpha/lib/

4. 验证Sqoop是否可用

bin/sqoop help

可能会出现一些Warning警告但是可以忽略，如果有如下信息输出表示sqoop可用：



5. 测试Sqoop是否能够成功连接数据库

bin/sqoop list-databases --connect jdbc:mysql://hadoop100:3306/ --username root --password “root的密码”

出现mysql中的数据库就说明能够正常操作mysql