CDH5相关组件手动安装

目录

[一、集群设置 1](#_Toc433878539)

[二、Zookeeper安装、配置及卸载 3](#_Toc433878540)

[三、HDFS安装、配置及卸载 6](#_Toc433878541)

[四、MapReduce安装、配置及卸载 9](#_Toc433878542)

本文主要介绍如何手动安装几个Cloudera主要相关组件。

1. 集群设置

1. **启动Apache http server(非master服务器，udh-yf-dev-20)**

sudo service httpd start # 启动

sudo service httpd status # 查看状态

**2. 创建CDH5仓库 repo(非master服务器，便于共享资源进行安装)**

# get the online repo from Cloudera

cd /etc/yum.repos.d/

wget http://archive.cloudera.com/cdh5/redhat/6/x86\_64/cdh/cloudera-cdh5.repo

# 从.repo 文件中构建仓库，下载到web server根目录

cd /var/www/html/

wget -c http://archive.cloudera.com/cdh5/redhat/6/x86\_64/cdh/5/ -r

wget -c http://archive-primary.cloudera.com/cdh5/redhat/6/x86\_64/cdh/5/ -r

# 创建本地的.repo文件

cd /etc/yum.repos.d/

vi cloudera-cdh5.repo

在.repo文件中加入如下内容：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[cloudera-cdh5]

# Packages for Cloudera's Distribution for Hadoop, Version 5, on RedHat or CentOS 6 x86\_64

name=Cloudera's Distribution for Hadoop, Version 5

baseurl=http://20.12.6.22/archive.cloudera.com/cdh5/redhat/6/x86\_64/cdh/5/

enabled=1

gpgcheck=0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**3. 关闭防火墙**

service iptables status

service iptables stop #暂停 (service iptables start, 开启)

chkconfig iptables off #停止

chkconfig --list |grep iptables #查看状态

**4. 服务器连接，建立无密码登录**

四台服务器：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# dev1 - udh-yf-dev-17: 20.12.6.19 (master) #

# dev2 - udh-yf-dev-18: 20.12.6.20 (slave) #

# dev3 - udh-yf-dev-19: 20.12.6.21 (slave) #

# dev4 - udh-yf-dev-20: 20.12.6.22 (slave) #

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 可分别对应代入以下安装过程！

# dev1-4

ssh-keygen -t rsa #生成密钥对

# dev1

scp .ssh/id\_rsa.pub dev2:mypublickey #将master公钥复制到各服务器中

scp .ssh/id\_rsa.pub dev3:mypublickey

scp .ssh/id\_rsa.pub dev4:mypublickey

# dev2-4

scp .ssh/id\_rsa.pub dev1:mypublickey2 #将各公钥复制到master服务器中

scp .ssh/id\_rsa.pub dev1:mypublickey3

scp .ssh/id\_rsa.pub dev1:mypublickey4

# dev1

cat mypublickey2 >.ssh/authorized\_keys #将公钥拷贝到authorized\_keys

cat mypublickey3 >>.ssh/authorized\_keys

cat mypublickey4 >>.ssh/authorized\_keys

# dev2-4

cat mypublickey >.ssh/authorized\_keys

**5. 修改linux安全级别**

vi /etc/selinux/config #将文件中 SELINUX=XXX -->XXX 代表级别改为 SELINUX=disabled

**6. 添加用户(各服务器均建立相同用户名用户，设置相同权限)**

adduser ae\_seven

passwd ae\_seven #之后更新密码(123\*udh)

**7. 增加用户权限**

chmod u+w /etc/sudoers #添加写权限

vim /etc/sudoers

# 编辑/etc/sudoers文件，将 'root ALL=(ALL) ALL'下加入

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ae\_seven ALL=(ALL) ALL # 编辑过程中：Esc-进入命令输入模式；‘i‘-编辑模式；‘：wq’(’：x‘) -保存并退出，可继续执行命令

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

chmod u-w /etc/sudoers #撤销写权限

su - ae\_seven #切换用户

sudo ls / #可利用ae\_seven用户编辑root文件

**8. 为用户ae\_seven建立相关目录**

su - ae\_seven #切换用户(su - root)

mkdir dev

mkdir tools

**9. 安装JDK**

# 安装

sudo rpm -ivh /root/jdk/jdk-7u25-linux-x64.rpm #i代表安装,e代表卸载,v代表显示安装过程,h代表显示#号样式的进度

# 设置环境变量信息 (不需要设置)

sudo vim /etc/profile #为系统的每个用户 设置环境信息,当用户第一次登录时,该文件被执行. 并从/etc/profile.d目录的配置文件中搜集shell的设置。

export JAVA\_HOME=/usr/java/jdk1.7.0\_25

export CLASSPATH=.:$JAVA\_HOME/jre/lib/rt.jar:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar

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

sudo source /etc/profile #使得刚刚添加到环境变量生效

ll /usr/java/jdk1.7.0\_25

# 建立软链接

cd /usr/bin

ln -s -f /usr/java/jdk1.7.0\_25/jre/bin/java

ln -s -f /usr/java/jdk1.7.0\_25/bin/javac

#获取安装java版本

java -version

1. Zookeeper安装、配置及卸载

###### Zookeeper简介

Zookeeper用于协同工作作用，集群中需要奇数个节点，在集群中占一半以上，提供的功能包括：配置维护、名字服务、分布式同步、组服务等。ZooKeeper的目标就是封装好复杂易出错的关键服务，将简单易用的接口和性能高效、功能稳定的系统提供给用户。

###### Zookeeper安装

##安装节点：dev2,dev3,dev4

**1. 安装zookeeper组件**

#节点dev2 - dev4:

sudo yum install zookeeper

sudo yum install zookeeper-server

# 可以在zoo.cfg中设置dataDir 和 dataLogDir

# dataDir=xxx

# dataLogDir=xxx

#

# 默认目录(yum)：

# bin - /usr/lib/zookeeper/bin

# log - /var/log/zookeeper

# data - /var/lib/zookeeper

**2. 修改配置文件**

sudo vim /etc/zookeeper/conf/zoo.cfg #在zoo.cfg文件的末尾加入下面三行信息，同时拷贝到其他节点

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

server.1=dev2:2888:3888

server.2=dev3:2888:3888

server.3=dev4:2888:3888

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

sudo scp /etc/zookeeper/conf/zoo.cfg dev3:/etc/zookeeper/conf/

sudo scp /etc/zookeeper/conf/zoo.cfg dev4:/etc/zookeeper/conf/ #拷贝

# 为每个节点建立myid文件，其中myid各不相同

sudo service zookeeper-server init --myid=1 # 不小心设置相同了可利用 sudo service zookeeper-server init --force --myid=x进行重设

sudo service zookeeper-server init --myid=2

sudo service zookeeper-server init --myid=3

# 启动/暂停/查看状态

sudo service zookeeper-server start

sudo service zookeeper-server stop

sudo service zookeeper-server status

/usr/lib/zookeeper/bin/zkServer.sh status

/usr/lib/zookeeper/bin/zkCli.sh -server 127.0.0.1:2181

**3. 安装namenode, datanode, jobtracker, tashtracker**

#namenode,jobtracker必须安装在同一服务器(master)，集群中除了master之外的服务器安装datanode和tasktracker

# 节点dev1(master)

sudo yum install hadoop-hdfs-namenode

sudo yum install hadoop-0.20-mapreduce-jobtracker

sudo yum install hadoop-client

sudo yum install hadoop-yarn-resourcemanager

sudo yum install hadoop-mapreduce-historyserver hadoop-yarn-proxyserver

# 节点dev2-4

sudo yum install hadoop-0.20-mapreduce-tasktracker

sudo yum install hadoop-client

sudo yum install hadoop-hdfs-datanode

sudo yum install hadoop-yarn-nodemanager hadoop-mapreduce

# 建立二级namenode(非必须)

# 仅在master节点上进行！！！

sudo yum clean all; sudo yum install hadoop-hdfs-secondarynamenode

**4. 安装hadoop-lzo(可暂时不安装，与其他组件协同工作时常常出现错误)**

(1). 节点dev1-4分别建立.repo文件

cd /etc/yum.repos.d

vi /etc/yum.repos.d/cloudera-gplextras5.repo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[cloudera-gplextras5]

# Packages for Cloudera's GPLExtras, Version 5, on RedHat or CentOS 6 x86\_64

name=Cloudera's GPLExtras, Version 5

baseurl=http://xxx/archive.cloudera.com/gplextras5/redhat/6/x86\_64/gplextras/5/

enabled=1

gpgcheck=0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(2). 安装

sudo yum install hadoop-lzo

(3). 修改配置文件core-site.xml.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<property>

<name>io.compression.codecs</name>

<value>org.apache.hadoop.io.compress.DefaultCodec,org.apache.hadoop.io.compress.GzipCodec,org.apache.hadoop.io.compress.BZip2Codec,com.hadoop.compression.lzo.LzoCodec,com.hadoop.compression.lzo.LzopCodec,org.apache.hadoop.io.compress.SnappyCodec</value>

</property>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#卸载

sudo yum remove hadoop-lzo #同时删除core-site.xml中相应内容

1. HDFS安装、配置及卸载

**1. 复制默认配置文件到custom目录**

sudo cp -r /etc/hadoop/conf.dist /etc/hadoop/conf.my\_cluster

**2. 设置新文件为hadoop的默认配置文件：**

sudo alternatives --verbose --install /etc/hadoop/conf hadoop-conf /etc/hadoop/conf.my\_cluster 50

sudo alternatives --set hadoop-conf /etc/hadoop/conf.my\_cluster

**3. 修改配置文件**

**core-site.xml:**

<property>

<name>fs.defaultFS</name>

<value>hdfs://dev1.yonyou.com:9000/</value> #dev1=udh-yf-dev-17

</property>

<property>

<name>fs.trash.interval</name>

<value>1440</value>

</property>

<property>

<name>fs.trash.checkpoint.interval</name>

<value>0</value>

</property>

<property>

<name>hadoop.proxyuser.oozie.hosts</name>

<value>\*</value>

</property>

<property>

<name>hadoop.proxyuser.oozie.groups</name>

<value>\*</value>

</property>

#以下部分若安装lzo时添加到配置文件

<property>

<name>io.compression.codecs</name>

<value>org.apache.hadoop.io.compress.DefaultCodec,org.apache.hadoop.io.compress.GzipCodec,org.apache.hadoop.io.compress.BZip2Codec,com.hadoop.compression.lzo.LzoCodec,com.hadoop.compression.lzo.LzopCodec,org.apache.hadoop.io.compress.SnappyCodec</value>

</property>

**hdfs-site.xml:**

<property>

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

<value>/data/dfs/nn</value>

</property>

<property>

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

<value>/data1/hadoop/dfs/dn,/data2/hadoop/dfs/dn,/data3/hadoop/dfs/dn,/data4/hadoop/dfs/dn</value>

</property>

<property>

<name>dfs.datanode.failed.volumes.tolerated</name>

<value>3</value>

</property>

<property>

<name>dfs.permissions.superusergroup</name>

<value>hadoop</value>

</property>

<property>

<name>dfs.replication</name>

<value>3</value>

</property>

<property>

<name>dfs.datanode.max.xcievers</name>

<value>4096</value>

</property>

<property>

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

<value>20.12.6.19:50070</value>

</property>

<property>

<name>dfs.datanode.fsdataset.volume.choosing.policy</name>

<value>org.apache.hadoop.hdfs.server.datanode.fsdataset.AvailableSpaceVolumeChoosingPolicy</value>

</property>

<property>

<name>dfs.datanode.available-space-volume-choosing-policy.balanced-space-threshold</name>

<value>53687091200</value>

</property>

<property>

<name>dfs.datanode.available-space-volume-choosing-policy.balanced-space-preference-fraction</name>

<value>0.75</value>

</property>

<property>

<name>dfs.webhdfs.enabled</name>

<value>true</value>

</property>

<property>

<name>dfs.client.read.shortcircuit</name>

<value>true</value>

</property>

<property>

<name>dfs.domain.socket.path</name>

<value>/var/run/hadoop-hdfs/dn.\_PORT</value>

</property>

<property>

<name>dfs.client.file-block-storage-locations.timeout</name>

<value>10000</value>

</property>

<property>

<name>dfs.datanode.hdfs-blocks-metadata.enabled</name>

<value>true</value>

</property>

#各节点配置文件相同

**4. 设置权限**

# 节点dev1

sudo mkdir -p /data/dfs/nn

sudo chown -R hdfs:hdfs /data/dfs/nn

2014-7-31 /data/dfs/nn

# 节点dev2-dev4

sudo mkdir -p /data1/hadoop/dfs/dn /data2/hadoop/dfs/dn /data3/hadoop/dfs/dn /data4/hadoop/dfs/dn

sudo chown -R hdfs:hdfs /data1/hadoop/dfs/dn /data2/hadoop/dfs/dn /data3/hadoop/dfs/dn /data4/hadoop/dfs/dn

sudo chmod 700 /data1/hadoop/dfs/dn /data2/hadoop/dfs/dn /data3/hadoop/dfs/dn /data4/hadoop/dfs/dn

**5. format(仅namenode)**

sudo -u hdfs hadoop namenode -format

**6. 卸载**

sudo yum remove hadoop-hdfs-namenode

#删除namenode节点上相关目录

sudo rm -rf /data/dfs/nn

#删除datanode节点上相关目录

sudo rm -rf /data1/hadoop/dfs/dn /data2/hadoop/dfs/dn /data3/hadoop/dfs/dn /data4/hadoop/dfs/dn

#接着可以安装新的HDFS组件

1. MapReduce安装、配置及卸载

**1. 修改配置文件**

mapred-site.xml

<property>

<name>mapred.job.tracker</name>

<value>dev1.yonyou.com:9001</value> #dev1=udh-yf-dev-17

</property>

<property>

<name>mapred.tasktracker.map.tasks.maximum</name>

<value>12</value>

</property>

<property>

<name>mapred.tasktracker.reduce.tasks.maximum</name>

<value>12</value>

</property>

<property>

<name>mapred.local.dir</name>

<value>/data1/hadoop/mapred/local,/data2/hadoop/mapred/local,/data3/hadoop/mapred/local,/data4/hadoop/mapred/local</value>

</property>

slaves.xml

dev2.yonyou.com

dev3.yonyou.com

dev4.yonyou.com

**2. 创建mapreduce目录，并进行权限设置(针对所有节点)**

sudo mkdir -p /data1/hadoop/mapred/local /data2/hadoop/mapred/local /data3/hadoop/mapred/local /data4/hadoop/mapred/local

sudo chown -R mapred:hadoop /data1/hadoop/mapred/local /data2/hadoop/mapred/local /data3/hadoop/mapred/local /data4/hadoop/mapred/local

**3. 将配置文件复制到各个节点**

sudo scp -r /etc/hadoop/conf.my\_cluster dev2:/etc/hadoop/

sudo scp -r /etc/hadoop/conf.my\_cluster dev3:/etc/hadoop/

sudo scp -r /etc/hadoop/conf.my\_cluster dev4:/etc/hadoop/

**4. 设置新的默认配置文件在节点dev2-4**

sudo alternatives --verbose --install /etc/hadoop/conf hadoop-conf /etc/hadoop/conf.my\_cluster 50

sudo alternatives --set hadoop-conf /etc/hadoop/conf.my\_cluster

**5. 在集群中的每个节点上启动HDFS**

for x in `cd /etc/init.d ; ls hadoop-hdfs-\*` ; do sudo service $x start ; done

for x in `cd /etc/init.d ; ls hadoop-hdfs-\*` ; do sudo service $x stop ; done

for x in `cd /etc/init.d ; ls hadoop-hdfs-\*` ; do sudo service $x restart ; done

**6. 设置HDFS目录(仅master)**

sudo -u hdfs hadoop fs -mkdir /tmp

sudo -u hdfs hadoop fs -chmod -R 1777 /tmp

sudo -u hdfs hadoop fs -mkdir -p /var/lib/hadoop-hdfs/cache/mapred/mapred/staging

sudo -u hdfs hadoop fs -chmod 1777 /var/lib/hadoop-hdfs/cache/mapred/mapred/staging

sudo -u hdfs hadoop fs -chown -R mapred /var/lib/hadoop-hdfs/cache/mapred

sudo -u hdfs hadoop fs -mkdir /tmp/mapred/system

sudo -u hdfs hadoop fs -chown mapred:hadoop /tmp/mapred/system

sudo -u hdfs hadoop fs -mkdir /user/ae\_seven

sudo -u hdfs hadoop fs -chown ae\_seven /user/ae\_seven

sudo -u hdfs hadoop fs -ls -R / #查看目录详情

**7. 启动map-reduce**

# 节点dev1

sudo service hadoop-0.20-mapreduce-jobtracker start

#节点dev2-4

sudo service hadoop-0.20-mapreduce-tasktracker start

**8. 测试**

hadoop fs -put /etc/hadoop/conf.my\_cluster/hdfs-site.xml #放置文件

sudo -u ae\_seven hadoop jar /usr/lib/hadoop-0.20-mapreduce/hadoop-examples.jar wordcount /user/ae\_seven/hdfs-site.xml /user/ae\_seven/output1(未成功)

hadoop jar /usr/lib/hadoop-0.20-mapreduce/hadoop-examples.jar wordcount /user/root/hdfs-site.xml /user/root/output1

#查看输出

hadoop fs -cat /user/root/output4/\*

#查看版本

hadoop version

**9. 卸载**

sudo yum remove hadoop-0.20-mapreduce-jobtracker

#删除相关目录

sudo remove -rf /data1/hadoop/mapred/local /data2/hadoop/mapred/local /data3/hadoop/mapred/local /data4/hadoop/mapred/local