Hbase安装

作者 TT

前言：

在hadoop生态圈中，非关系型数据库Hbase占有重要一席之地。这里介绍一下Hbase安装过程，首先需要明白的是，hbase的安装条件：

1. **JDK1.7+以上**
2. **Hadoop2.5+以上**
3. **Zookeeper3.4.x以上**

那么详细安装步骤如下：

1、首先在cdh中http://archive.cloudera.com/cdh5/cdh/5/下载Hadoop版本，这里我选择hadoop-2.5.0-cdh5.3.10.tar.gz，因为Hbase安装是基于Hadoop与hdfs的，另外还需要安装zookeeper，下面我们一个个来

为什么选用cdh版本呢？是为了解决Hadoop生态圈工具兼容的问题，等于说是配套设施。

2、同样habse我也下载hbase-0.98.6-cdh5.3.10.tar.gz，保证版本号一致，免得在lniux下报一些不寻常的错误。

hadoop安装：

3、下载完成之后，rz上传文件到指定目录，我这里上传到

```
?[root@zt hadoop]# pwd
/usr/local/hadoop
[root@zt hadoop]#
```

```
-rw-r--r--.  1 root root 314672011 Aug 14 09:39 hadoop-2.5.0-cdh5.3.10.tar.gz
```

tar -zvxf hadoop-2.5.0-cdh5.3.10

4、进到 解压后的hadoop中的etc目录

```
[root@zt hadoop]# pwd
/usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/etc/hadoop
```

查看该目录下所有文件

```
[root@zt hadoop]# ls
capacity-scheduler.xml  hadoop-env.sh               hdfs-site.xml           httpfs-site.xml       kms-site.xml            mapred-site.xml.template  yarn-env.sh
configuration.xsl       hadoop-metrics2.properties  httpfs-env.sh           kms-acls.xml          log4j.properties        slaves                    yarn-site.xml
container-executor.cfg  hadoop-metrics.properties   httpfs-log4j.properties kms-env.sh            mapred-env.sh           ssl-client.xml.example
core-site.xml           hadoop-policy.xml           httpfs-signature.secret kms-log4j.properties  mapred-queues.xml.template ssl-server.xml.example
```

5、修改hadoop中的hadoop-env.sh文件

vi hadoop-env.sh

在其中修改JAVA_HOME的默认地址，不知道可以使用echo $JAVA_HOME查看你的java安装目录

```
# remote nodes.

# The java implementation to use.
export JAVA_HOME=/usr/java/jdk1.8.0_111
```

同时可以修改默认日志输出地址

```
# Where log files are stored.  $HADOOP_HOME/logs by default.
export HADOOP_LOG_DIR=/usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/logs
```

6、创建数据存放目录

```
[root@zt hadoop-2.5.0-cdh5.3.10]# mkdir -p data/tmp
[root@zt hadoop-2.5.0-cdh5.3.10]# cd data/tmp
[root@zt tmp]# pwd
/usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/data/tmp
[root@zt tmp]#
```

7、配置core-site.xml

查看官方文档的单节点配置：http://hadoop.apache.org/docs/r2.5.2/hadoop-project-dist/hadoop-common/SingleCluster.html

```
<configuration>
      <property>
        <name>fs.defaultFS</name>
        <value>hdfs://one.zhangtao.com:9000</value>
      </property>
      <property>
        <name>hadoop.tmp.dir</name>
        <value>/usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/data/tmp</value>
      </property>
</configuration>
```

8、配置hdfs-site.xml

```
[root@zt hadoop]# cat hdfs-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
    <property>
        <name>dfs.permissions.enabled</name>
        <value>false</value>
    </property>
</configuration>
[root@zt hadoop]#
```

9、配置slaves

```
[root@zt hadoop]# cat slaves
#localhost
192.168.10.128
[root@zt hadoop]#
```

10、格式化

```
[root@zt hadoop]# cd ../../
[root@zt hadoop-2.5.0-cdh5.3.10]# bin/hdfs namenode -format
```

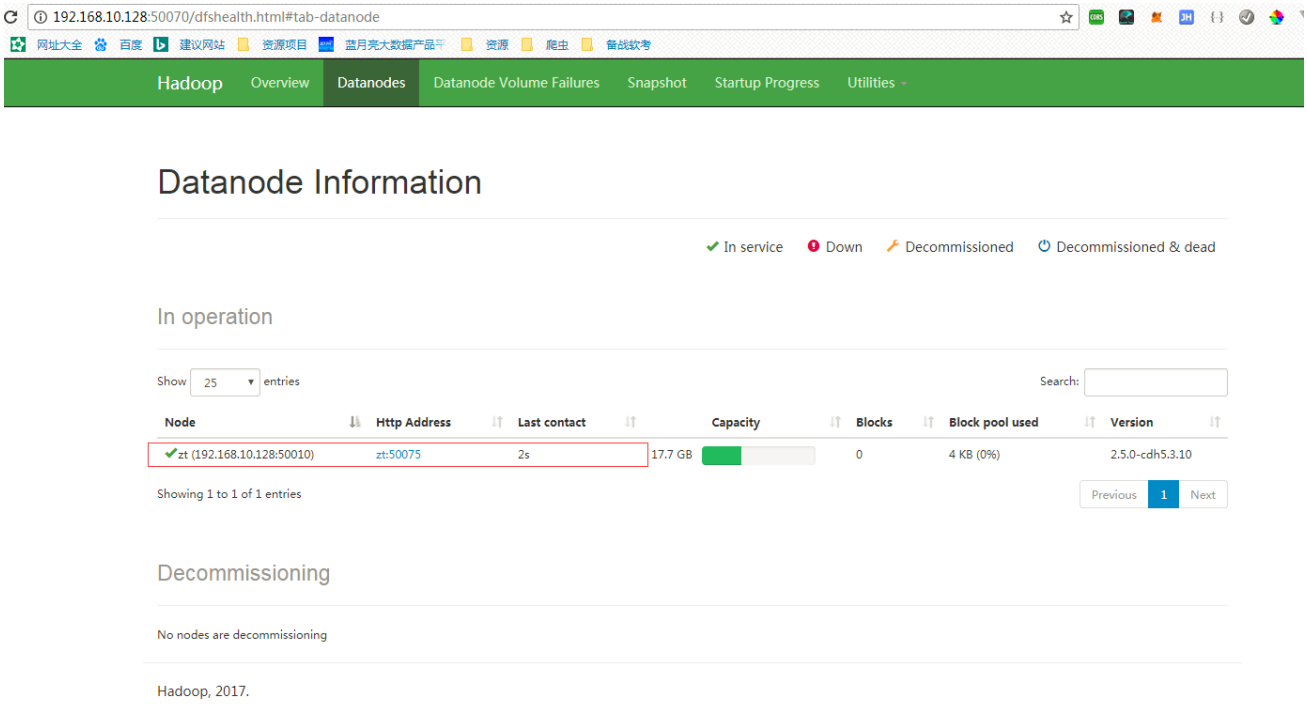
格式化后的效果

```
02 INFO namenode.NNConf: XAttrs enabled? true
02 INFO namenode.NNConf: Maximum size of an xattr: 16384
02 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1640666074-192.168.10.128-1534212902115
02 INFO common.Storage: Storage directory /usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/data/tmp/dfs/name has been successfully formatted.
03 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
03 INFO util.ExitUtil: Exiting with status 0
03 INFO namenode.NameNode: SHUTDOWN_MSG:
************************************************
utting down NameNode at zt/192.168.10.128
************************************************/
```

11、启动hadoop

```
[root@zt hadoop-2.5.0-cdh5.3.10]# sbin/hadoop-daemon.sh start namenode
starting namenode, logging to /usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/logs/hadoop-root-namenode-zt.out
[root@zt hadoop-2.5.0-cdh5.3.10]# sbin/hadoop-daemon.sh start datanode
starting datanode, logging to /usr/local/hadoop/hadoop-2.5.0-cdh5.3.10/logs/hadoop-root-datanode-zt.out
[root@zt hadoop-2.5.0-cdh5.3.10]#
```

12、效果



在浏览器中输入当前机器IP加上50070端口号；从上面可以看到hadoop启动成功！！！

zookeeper安装部署集群，这里我部署三台机器组成集群。

1、安装zookeeper

同样把下载好的zookeeper上传到linux下

```
[root@two zookeeper]# ll
total 34272
drwxr-xr-x. 10 1001 1001     4096 Apr 27 15:17 zookeeper-3.4.10
-rw-r--r--.  1 root root 35042811 Nov 16  2017 zookeeper-3.4.10.tar.gz
-rw-r--r--.  1 root root    42838 Apr 27 15:17 zookeeper.out
[root@two zookeeper]#
```

2、解压zookeeper

tar -zvxf zookeeper-3.4.10.tar.gz

3、配置 zoo.cfg

拷贝一份zoo.cfg出来

cp ./zoo_sample.cfg ./zoo.cfg

vi zoo.cfg

server.A=B：C：D：其中 A 是一个数字，表示这个是第几号服务器；**B 是这个服务器的 ip 地址；C 表示的是这个服务器与集群中的 Leader 服务器交换信息的端口；D 表示的是万一集群中的 Leader 服务器挂了，需要一个端口来重新进行选举，选出一个新的 Leader，而这个端口就是用来执行选举时服务器相互通信的端口。**

```
# Purge task interval in hours
# Set to "0" to disable auto purge feature
#autopurge.purgeInterval=1
server.1=one.zhangtao.com:2881:3881
server.2=two.zhangtao.com:2881:3881
server.3=three.zhangtao.com:2881:3881
~
~
```

3、配置dataDir：即是存放内存数据库快照的位置，dataLogDir是事务日志目录

```
# the directory where the snapshot is stored.
# do not use /tmp for storage, /tmp here is just
dataDir=/usr/local/zookeeperData
dataLogDir=/usr/local/zookeeperData/logs
# the port at which the clients will connect
clientPort=2181
```

4、配置myid

```
[root@two zookeeperData]# ll
total 8
drwxr-xr-x. 3 root root 22 Dec  1  2017 logs
-rw-r--r--. 1 root root  2 Dec  1  2017 myid
drwxr-xr-x. 2 root root 94 Aug 14 11:41 version-2
-rw-r--r--. 1 root root  4 Aug 14 11:41 zookeeper_server.pid
[root@two zookeeperData]#
```

根据zoo.cfg中的集群设置的编号来写入：**vi myid**

```
[root@two zookeeperData]# cat myid
2
```

5、配置映射

vi /etc/hosts

```
127.0.0.1   localhost localhost.localdomain lo
::1         localhost localhost.localdomain lo
#zookeeper clusterservers
192.168.10.128  one.zhangtao.com
192.168.10.129  two.zhangtao.com
192.168.10.130  three.zhangtao.com
~
~
```

6、分发另外两台机器即可

scp -r ./zookeeper-3.4.10 192.168.10.128:/usr/local/zookeeper/ 192.168.10.130:/usr/local/zookeeper/

同样修改一下服务器映射信息 vi /etc/hosts，实质上分发一下也是很好的，因为都一样的嘛

7、启动zookeeper

进到

```
[root@two bin]# pwd
/usr/local/zookeeper/zookeeper-3.4.10/bin
```

./zkServer.sh start 启动 注意同时启动三台，然后**./zkServer.sh status** 查看zookeeper当前状态

我这里集群成功：

```
6939 Jps
[root@two bin]# ./zkServer.sh  status
ZooKeeper JMX enabled by default
Using config: /usr/local/zookeeper/zookeeper-3.4.10/bin/../conf/zoo.cfg
Mode: leader
```

```
[root@three bin]# ./zkServer.sh  status
ZooKeeper JMX enabled by default
Using config: /usr/local/zookeeper/zookeeper-3.4.10/bin/../conf/zoo.cfg
Mode: follower
```

```
[root@one bin]# ./zkServer.sh  status
ZooKeeper JMX enabled by default
Using config: /usr/local/zookeeper/zookeeper-3.4.10/bin/../conf/zoo.cfg
Mode: follower
```

一台master 两台follower,至此集群成功部署。

安装hbase：

1、rz上传hbase到指定目录

tar -zxf hbase-0.98.6-cdh5.3.10.tar.gz

```
[root@zt hbase]# ls
hbase-0.98.6-cdh5.3.10
[root@zt hbase]#
```

2、Hbase分布式配置

修改hbase-env.sh，配置JAVA_HOME

```
# The java implementation to use.  Java 1.6 required.
export JAVA_HOME=/usr/java/jdk1.8.0_111
```

修改默认使用自带的zookeeper配置，这里采用自己上面配置的zookeeper集群

```
# Tell HBase whether it should manage it's own instance of Zookeeper or not.
export HBASE_MANAGES_ZK=false
```

3、修改hbase-site.xml

创建数据目录

```
[root@zt conf]# cd ../
[root@zt hbase-0.98.6-cdh5.3.10]# mkdir -p data/tmp
[root@zt hbase-0.98.6-cdh5.3.10]# cd /data/tmp
```

```
<configuration>
  <property>
    <name>hbase.tmp.dir</name>
    <value>/usr/local/hbase/hbase-0.98.6-cdh5.3.10/data/tmp</value>
  </property>
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://one.zhangtao.com:9000/hbase</value>
  </property>
  <property>
    <name>hbase.cluster.distributed</name>
    <value>true</value>
  </property>
  <property>
    <name>hbase.zookeeper.quorum</name>
    <value>one.zhangtao.com,two.zhangtao.com,three.zhangtao.com</value>
  </property>
</configuration>
```

4、修改regionservers文件

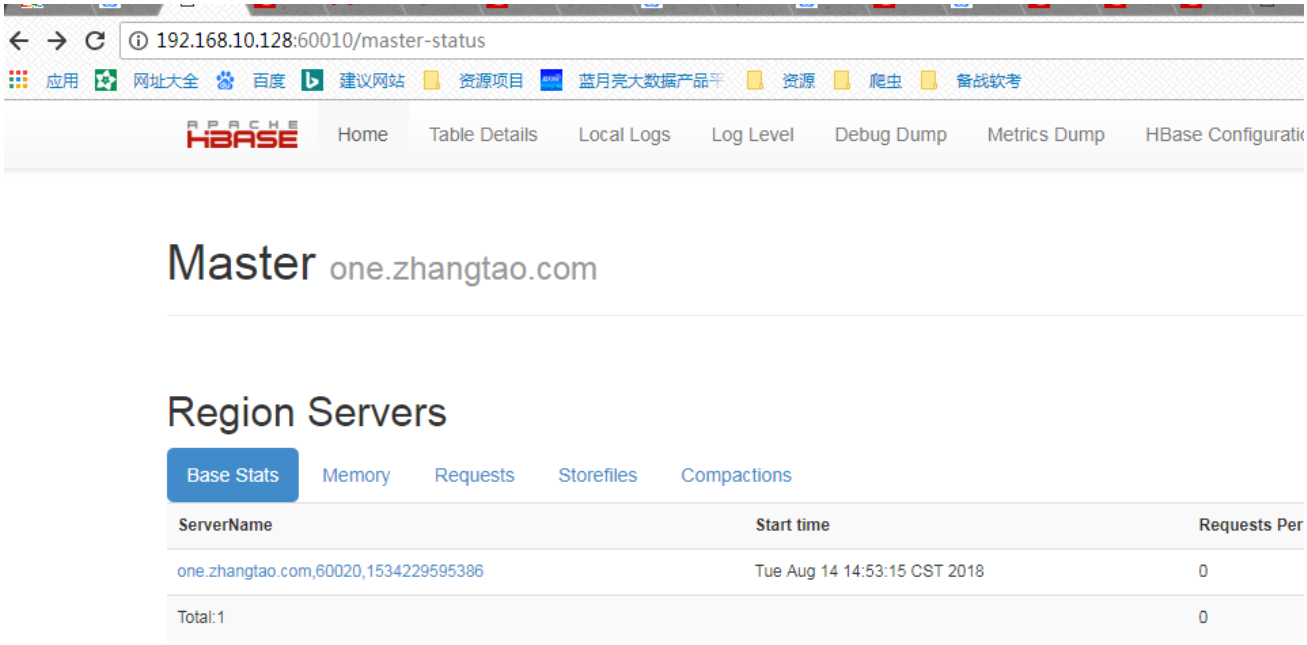
```
[root@one conf]# more regionservers
#localhost
one.zhangtao.com
```

5、启动hbase

./start-hbase.sh start

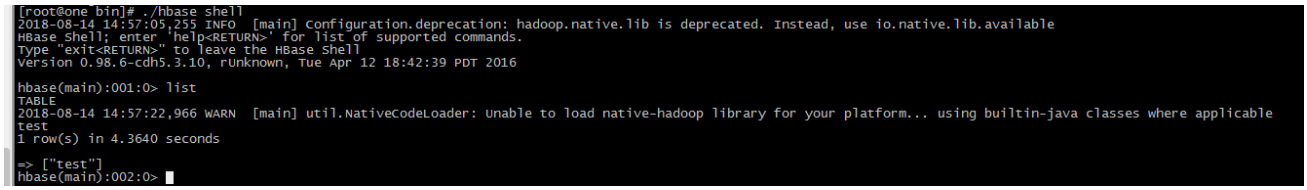
```
[root@one bin]# ./start-hbase.sh start
starting master, logging to /usr/local/hbase/hbase-0.98.6-cdh5.3.10/logs//hbase-root-master-one.zhangtao.com.out
root@one.zhangtao.com's password:
one.zhangtao.com: starting regionserver, logging to /usr/local/hbase/hbase-0.98.6-cdh5.3.10/logs//hbase-root-regionserver-one.zhangtao.com.out
#localhost: ssh: Could not resolve hostname #localhost: Name or service not known
[root@one bin]# jps
10212 HMaster
3877 NameNode
3927 DataNode
3737 QuorumPeerMain
10651 Jps
10381 HRegionServer
[root@one bin]#
```

查看一下监控界面：在浏览器中输入192.168.10.128:60010



可以看到Hbase安装成功！！！

最后让本少来操作一下hbase数据库，看看是不是真的有货~~~~~~~



果然如此，有东东