Nutch Hadoop Lucene Tutorial - Setting up the master node

How to setup Nutch 0.9.0 and Hadoop 0.12.2 with Lucene 2.1.0 on Debian

This tutorial is intended for:

- Nutch running on multiple machines with mapreduce and Hadoop
- Hadoop dfs on multiple machines
- Lucene search interface on multiple machines with local search indices

Prerequisites

```
# Login as root on the first machine which is going to be the master for the code distribution and the Hadoop
cluster.
su

# Enable the contrib and non-free package sources in /etc/apt/sources.list
vi /etc/apt/sources.list

# Install java5 apache2 and tomcat5
apt-get update
apt-get install sun-java5-jdk
apt-get install apache2
apt-get install tomcat5

# Configure tomcat
echo "JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun/" >> /etc/default/tomcat5
```

Download and build

```
# Download nutch-0.9
# Download ant from apache, there seems to be something missing in the ant that comes with Debian
wget ftp://apache.essentkabel.com/apache/lucene/nutch/nutch-0.9.tar.gz
wget http://archive.apache.org/dist/ant/binaries/apache-ant-1.6.5-bin.tar.gz
tar -xzvf nutch-0.9.tar.gz
tar -xzvf apache-ant-1.6.5-bin.tar.gz

# Build nutch with apache ant
cd nutch-0.9
/root/apache-ant-1.6.5/bin/ant package
```

Install and configure

```
# Create directories for nutch
mkdir /nutch-0.9
mkdir /nutch-0.9/build
mkdir /nutch-0.9/crawler
mkdir /nutch-0.9/dist
mkdir /nutch-0.9/filesystem
mkdir /nutch-0.9/home
mkdir /nutch-0.9/scripts
mkdir /nutch-0.9/source
mkdir /nutch-0.9/tars

# Create the nutch user and group
groupadd nutch
useradd -d /nutch-0.9/home -g nutch nutch
passwd nutch

# Copy the nutch build dir for the crawler
cp -Rv /root/nutch-0.9/build/nutch-0.9/* /nutch-0.9/crawler/

# Configure the crawler
echo "export HADOOP_HOME=/nutch-0.9/crawler" >> /nutch-0.9/crawler/conf/hadoop-env.sh
echo "export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun" >> /nutch-0.9/crawler/conf/hadoop-env.sh
echo "export HADOOP_LOG_DIR=/nutch-0.9/crawler/logs" >> /nutch-0.9/crawler/conf/hadoop-env.sh
echo "export HADOOP_SLAVES=/nutch-0.9/crawler/conf/slaves" >> /nutch-0.9/crawler/conf/hadoop-env.sh
```

Now the configuration files for the Nutch crawler in /nutch-0.9/crawler/conf/ have to be edited or created, these are:

- mapred-default.xml
- hadoop-site.xml
- nutch-site.xml
- url-crawlfilter.txt

Edit mapred-default.xml configuration file.

If it's missing, create it, with the following content:

```
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<property>
  <name>mapred.map.tasks</name>
  <value>2</value>
  <description>
    This should be a prime number larger than multiple number of slave hosts,
    e.g. for 3 nodes set this to 17
  </description>
</property>

<property>
  <name>mapred.reduce.tasks</name>
  <value>2</value>
  <description>
    This should be a prime number close to a low multiple of slave hosts,
    e.g. for 3 nodes set this to 7
  </description>
</property>

</configuration>
```

Edit hadoop-site.xml

```
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>

<property>
  <name>fs.default.name</name>
  <value>???:9000</value>
  <description>
    The name of the default file system. Either the literal string
    "local" or a host:port for NDFS.
  </description>
</property>

<property>
  <name>mapred.job.tracker</name>
  <value>???:9001</value>
  <description>
    The host and port that the MapReduce job tracker runs at. If
    "local", then jobs are run in-process as a single map and
    reduce task.
  </description>
</property>

<property>
  <name>mapred.tasktracker.tasks.maximum</name>
  <value>2</value>
  <description>
    The maximum number of tasks that will be run simultaneously by
    a task tracker. This should be adjusted according to the heap size
    per task, the amount of RAM available, and CPU consumption of each task.
  </description>
</property>

<property>
  <name>mapred.child.java.opts</name>
  <value>-Xmx200m</value>
  <description>
    You can specify other Java options for each map or reduce task here,
    but most likely you will want to adjust the heap size.
  </description>
</property>

<property>
  <name>dfs.name.dir</name>
  <value>/nutch-0.9/filesystem/name</value>
</property>

<property>
  <name>dfs.data.dir</name>
  <value>/nutch-0.9/filesystem/data</value>
</property>

<property>
  <name>mapred.system.dir</name>
  <value>/nutch-0.9/filesystem/mapreduce/system</value>
</property>

<property>
  <name>mapred.local.dir</name>
  <value>/nutch-0.9/filesystem/mapreduce/local</value>
</property>

<property>
  <name>dfs.replication</name>
  <value>1</value>
</property>

</configuration>
```

Edit nutch-site.xml

Edit the nutch-site.xml file. Take the contents below and fill in the value tags.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
  <name>http.agent.name</name>
  <value></value>
  <description>HTTP 'User-Agent' request header. MUST NOT be empty -
  please set this to a single word uniquely related to your organization.

  NOTE: You should also check other related properties:

  http.robots.agents
  http.agent.description
  http.agent.url
  http.agent.email
  http.agent.version

  and set their values appropriately.

  </description>
</property>

<property>
  <name>http.agent.description</name>
  <value></value>
  <description>Further description of our bot- this text is used in
  the User-Agent header.  It appears in parenthesis after the agent name.
  </description>
</property>

<property>
  <name>http.agent.url</name>
  <value></value>
  <description>A URL to advertise in the User-Agent header.  This will
   appear in parenthesis after the agent name. Custom dictates that this
   should be a URL of a page explaining the purpose and behavior of this
   crawler.
  </description>
</property>

<property>
  <name>http.agent.email</name>
  <value></value>
  <description>An email address to advertise in the HTTP 'From' request
   header and User-Agent header. A good practice is to mangle this
   address (e.g. 'info at example dot com') to avoid spamming.
  </description>
</property>
</configuration>
```

Edit crawl-urlfilter.txt

Edit the crawl-urlfilter.txt file to edit the pattern of the urls that have to be fetched.

```
cd /nutch-0.9.0/search
vi conf/crawl-urlfilter.txt

change the line that reads:   +^http://([a-z0-9]*\.)*MY.DOMAIN.NAME/
to read:                      +^http://([a-z0-9]*\.)*/
```

Finishing the installation

```
# Change the ownership of all files to the nutch user
chown -R nutch:nutch /nutch-0.9

# Log in as nutch
su nutch

# Create ssh keys. These are needed by the hadoop scripts.
ssh-keygen -t rsa
cp /nutch-0.9/home/.ssh/id_rsa.pub /nutch-0.9/home/.ssh/authorized_keys

# Format the name node
cd /nutch-0.9/crawler
bin/hadoop namenode -format
```

Start crawling

To start crawling from a few urls as seeds an url directory is made in which a seed file is put with some seed urls. This file is put into the hdfs, to check if hdfs has stored the directory use the dfs -ls option of hadoop.

```
mkdir urls
echo "http://lucene.apache.org" >> urls/seed
bin/hadoop dfs -put urls urls
bin/hadoop dfs -ls urls
```

Start an initial crawl

```
export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun/
bin/nutch crawl urls -dir crawled -depth 3
```

On the masternode the progress and status can be viewed with a webbrowser. [http://localhost:50030/|http://localhost:50030/]

[Nutch_Hadoop_Lucene_Tutorial_%3a_Setting_up_the_slave_nodes]
[Nutch_Hadoop_Lucene_Tutorial_%3a_Setting_up_the_master_search_node]
[Nutch_Hadoop_Lucene_Tutorial_%3a_Setting_up_the_slave_search_nodes]
[Nutch_Hadoop_Lucene_Tutorial_%3a_Recrawl]
[Nutch_Hadoop_Lucene_Tutorial_%3a_Spliting_up_the_index]