Cluster Setting

Big Data Programming

Lee Hae Joon

Prerequisites

- Ubuntu 14.04
- Hadoop-2.6.5
- JAVA 8
- SSH

Setup of multi node cluster in Hadoop

We have three machines (master and slaves) with IP

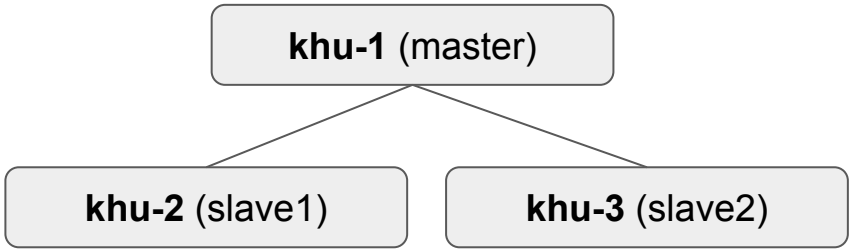
- Master IP: ${master}
- Slaves IPs: ${slave1}, ${slave2}

Open hosts file to add master and data node their respective IP addresses (FINAL STEP)

hosts

```
192.168.0.35 khu-1
192.168.0.36 khu-2
192.168.0.37 khu-3

127.0.0.1 localhost
```

Before doing this, you have to decide which node is responsible for master and slaves



Create the SSH key in master and slaves nodes

We have three machines (master and slaves) with IP

>> ssh-keygen -t rsa -P “”

>> cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys

Copy the master key to slave's authorized_keys

>> scp ~/.ssh/id_rsa.pub %{slave1}:/home/hadoop/.ssh/authorized_keys

>> scp ~/.ssh/id_rsa.pub %{slave2}:/home/hadoop/.ssh/authorized_keys

Copy each slaves key to master's authorized_keys

>> scp ~/.ssh/id_rsa.pub %{master}:/home/hadoop/.ssh/authorized_keys

Prof. already installed programs

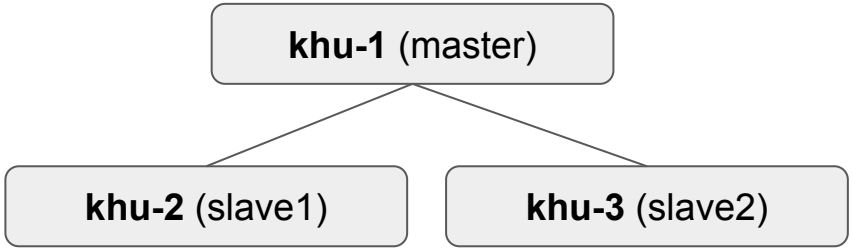
- hadoop-2.6.5 // hadoop version
- java 1.8 // java -version
- git
- zsh // zsh
- ssh // ssh localhost

Configuration setting in hadoop-2.6.5 in master and slaves

hadoop-env.sh

```
# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-8-oracle
export HADOOP_HOME="/home/hadoop/hadoop-2.6.5"
# The jsvc implementation to use. Jsvc is required to run secure
# that bind to privileged ports to provide authentication of dat
# protocol.  Jsvc is not required if SASL is configured for auth
# data transfer protocol using non-privileged ports.
#export JSVC_HOME=${JSVC_HOME}

export HADOOP_CONF_DIR=${HADOOP_CONF_DIR:-"/etc/hadoop"}

# Extra Java CLASSPATH elements.  Automatically insert capacity-
for f in $HADOOP_HOME/contrib/capacity-scheduler/*.jar; do
  if [ "$HADOOP_CLASSPATH" ]; then
    export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$f
  else
    export HADOOP_CLASSPATH=$f
  fi
done
```

Configuration setting in hadoop-2.6.5 in master and slaves



Configuration setting in hadoop-2.6.5 in master and slaves

mapred-site.xml

```
<?xml version="1.0"?>
<?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>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
</configuration>
```

Configuration setting in hadoop-2.6.5 in master and slaves

core-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>fs.defaultFS</name>
        <value>hdfs://${master}:9000</value>
    </property>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>/home/hadoop/hadoop-2.6.5/tmp</value>
    </property>
</configuration>
```

Configuration setting in hadoop-2.6.5

in master

hdfs-site.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>


<configuration>
        <property>
                <name>dfs.replication</name>
                <value>2</value>
        </property>
        <property>
                <name>dfs.permissions</name>
                <value>false</value>
        </property>
        <property>
                <name>dfs.namenode.name.dir</name>
                <value>/home/hadoop/hadoop-2.6.5/data/namenode</value>
        </property>
        <property>
                <name>dfs.datanode.data.dir</name>
                <value>/home/hadoop/hadoop-2.6.5/data/datanode</value>
        </property>

</configuration>
```

in slaves

hdfs-site.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>


<configuration>
        <property>
                <name>dfs.replication</name>
                <value>2</value>
        </property>
        <property>
                <name>dfs.permissions</name>
                <value>false</value>
        </property>
        <property>
                <name>dfs.datanode.data.dir</name>
                <value>/home/hadoop/hadoop-2.6.5/data/datanode</value>
        </property>

</configuration>
```

Configuration setting in hadoop-2.6.5 in master and slaves

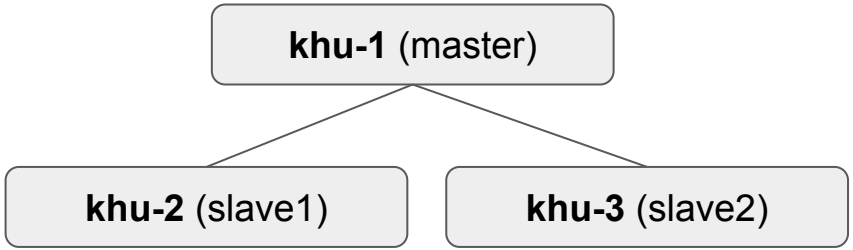
yarn-site.xml

```
<?xml version="1.0"?>

<configuration>

<!-- Site specific YARN configuration properties -->
        <property>
                <name>yarn.nodemanager.aux-services</name>
                <value>mapreduce_shuffle</value>
        </property>
        <property>
                <name>
                yarn.nodemanager.auxservices.mapreduce.shuffle.class</
                name>
                <value>org.apache.hadoop.mapred.ShuffleHandler</value>
        </property>
        <property>
                <name>yarn.resourcemanager.hostname</name>
                <value>${master}</value>
        </property>

</configuration>
```

Open hosts file to add master and data node their respective IP addresses (FINAL STEP)

hosts

```
192.168.0.35 khu-1
192.168.0.36 khu-2
192.168.0.37 khu-3

127.0.0.1 localhost
```



Now, you almost set hadoop cluster.

Next step?

```
hadoop@stud-1:~/hadoop-2.7.4$ jps
3012 ResourceManager
2661 DataNode
3160 NodeManager
3881 Jps
2493 NameNode
2861 SecondaryNameNode
hadoop@stud-1:~/hadoop-2.7.4$ sh remote_jps.sh ip_list
SENDING 192.168.0.58
2101 Jps
1736 DataNode
1883 NodeManager
SENDING 192.168.0.57
2530 Jps
2312 NodeManager
2169 DataNode
hadoop@stud-1:~/hadoop-2.7.4$
```

>> hadoop namenode -format

>> ./sbin/start-all.sh

>> jps // check it in each node !

>> hdfs fsck // check your cluster is healthy

Trouble-shooting

If something is wrong, you should deal with it by yourself.

Please refer to each log of ‘resource manager, node manager, namenode, and datanode’. These logs are stored in each node.

If everything is fine, test wordcount in your cluster.

You already know how to do it

1. **upload the arbitrary file to HDFS**
2. **hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.6.5.jar wordcount input output**

If a previous task is done,
now you have to do a practice from KHUHUB.