Hadoop2OnWindows

Build and Install Hadoop 2.x or newer on Windows

Introduction

Hadoop version 2.2 onwards includes native support for Windows. The official Apache Hadoop releases do not include Windows binaries (yet, as of January 2014). However building a Windows package from the sources is fairly straightforward.

Hadoop is a complex system with many components. Some familiarity at a high level is helpful before attempting to build or install it or the first time. Familiarity with Java is necessary in case you need to troubleshoot.

Building Hadoop Core for Windows

Choose target OS version

The Hadoop developers have used **Windows Server 2008** and **Windows Server 2008 R2** during development and testing. **Windows Vista** and **Windows 7** are also likely to work because of the Win32 API similarities with the respective server SKUs. We have **not** tested on Windows XP or any earlier versions of Windows and these are not likely to work. Any issues reported on Windows XP or earlier will be closed as *Invalid*.

Do not attempt to run the installation from within *Cygwin*. Cygwin is neither required nor supported.

Choose Java Version and set JAVA_HOME

Oracle JDK versions *1.7* and *1.6* have been tested by the Hadoop developers and are known to work.

Make sure that *JAVA_HOME* is set in your environment and does not contain any spaces. If your default Java installation directory has spaces then you must use the Windows 8.3 Pathname instead e.g. *c:\Progra~1\Java\...* instead of *c:\Program Files\Java\...*.

Getting Hadoop sources

The current stable release as of August 2014 is 2.5. The source distribution can be retrieved from the ASF download server or using subversion or git.

- From the ASF Hadoop download page or a mirror.
- Subversion URL: _https://svn.apache.org/repos/asf/hadoop/common/branches/branch-2.5_
- Git repository URL: *git://git.apache.org/hadoop-common.git*. After downloading the sources via git, switch to the stable 2.5 using ***git checkout branch-2.5***, or use the appropriate branch name if you are targeting a newer version.

Installing Dependencies and Setting up Environment for Building

The BUILDING.txt file in the root of the source tree has detailed information on the list of requirements and how to install them. It also includes information on setting up the environment and a few quirks that are specific to Windows. It is strongly recommended that you read and understand it before proceeding.

A few words on Native IO support

Hadoop on Linux includes optional Native IO support. However Native IO is mandatory on Windows and without it you will not be able to get your installation working. You must follow all the instructions from BUILDING.txt to ensure that Native IO support is built correctly.

Build and Copy the Package files

To build a binary distribution run the following command from the root of the source tree.

```
mvn package -Pdist,native-win -DskipTests -Dtar
```

Note that this command must be run from a *Windows SDK command prompt* as documented in BUILDING.txt. A successful build generates a binary hadoop *.tar.gz* package in _hadoop-dist\target_.

The Hadoop version is present in the package file name. If you are targeting a different version then the package name will be different.

Installation

Pick a target directory for installing the package. We use *c:\deploy* as an example. Extract the tar.gz file (e.g.*hadoop-2.5.0.tar.gz*) under *c:\deploy*. This will yield a directory structure like the following. If installing a multi-node cluster, then repeat this step on every node.

```
C:\deploy>dir
 Volume in drive C has no label.
 Volume Serial Number is 9D1F-7BAC

 Directory of C:\deploy

01/18/2014  08:11 AM    <DIR>          .
01/18/2014  08:11 AM    <DIR>          ..
01/18/2014  08:28 AM    <DIR>          bin
01/18/2014  08:28 AM    <DIR>          etc
01/18/2014  08:28 AM    <DIR>          include
01/18/2014  08:28 AM    <DIR>          libexec
01/18/2014  08:28 AM    <DIR>          sbin
01/18/2014  08:28 AM    <DIR>          share
               0 File(s)              0 bytes
```

Starting a Single Node (pseudo-distributed) Cluster

This section describes the absolute minimum configuration required to start a Single Node (pseudo-distributed) cluster and also run an example MapReduce job.

Example HDFS Configuration

Before you can start the Hadoop Daemons you will need to make a few edits to configuration files. The configuration file templates will all be found in *c:\deploy\etc\hadoop*, assuming your installation directory is *c:\deploy*.

First edit the file **hadoop-env.cmd** to add the following lines near the end of the file.

```
set HADOOP_PREFIX=c:\deploy
set HADOOP_CONF_DIR=%HADOOP_PREFIX%\etc\hadoop
set YARN_CONF_DIR=%HADOOP_CONF_DIR%
set PATH=%PATH%;%HADOOP_PREFIX%\bin
```

Edit or create the file **core-site.xml** and make sure it has the following configuration key:

```
<configuration>
  <property>
    <name>fs.default.name</name>
    <value>hdfs://0.0.0.0:19000</value>
  </property>
</configuration>
```

Edit or create the file **hdfs-site.xml** and add the following configuration key:

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

Finally, edit or create the file **slaves** and make sure it has the following entry:

```
localhost
```

The default configuration puts the HDFS metadata and data files under **\tmp** on the current drive. In the above example this would be **c:\tmp**. For your first test setup you can just leave it at the default.

Example YARN Configuration

Edit or create **mapred-site.xml** under **%HADOOP_PREFIX%\etc\hadoop** and add the following entries, replacing *%USERNAME%* with your Windows user name.

```
<configuration>

   <property>
     <name>mapreduce.job.user.name</name>
     <value>%USERNAME%</value>
   </property>

   <property>
     <name>mapreduce.framework.name</name>
     <value>yarn</value>
   </property>

  <property>
    <name>yarn.apps.stagingDir</name>
    <value>/user/%USERNAME%/staging</value>
  </property>

  <property>
    <name>mapreduce.jobtracker.address</name>
    <value>local</value>
  </property>

</configuration>
```

Finally, edit or create **yarn-site.xml** and add the following entries:

```
<configuration>
  <property>
    <name>yarn.server.resourcemanager.address</name>
    <value>0.0.0.0:8020</value>
  </property>

  <property>
    <name>yarn.server.resourcemanager.application.expiry.interval</name>
    <value>60000</value>
  </property>

  <property>
    <name>yarn.server.nodemanager.address</name>
    <value>0.0.0.0:45454</value>
  </property>

  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>

  <property>
    <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
  </property>

  <property>
    <name>yarn.server.nodemanager.remote-app-log-dir</name>
    <value>/app-logs</value>
  </property>

  <property>
    <name>yarn.nodemanager.log-dirs</name>
    <value>/dep/logs/userlogs</value>
  </property>

  <property>
    <name>yarn.server.mapreduce-appmanager.attempt-listener.bindAddress</name>
    <value>0.0.0.0</value>
  </property>

  <property>
    <name>yarn.server.mapreduce-appmanager.client-service.bindAddress</name>
    <value>0.0.0.0</value>
  </property>

  <property>
    <name>yarn.log-aggregation-enable</name>
    <value>true</value>
  </property>

  <property>
    <name>yarn.log-aggregation.retain-seconds</name>
    <value>-1</value>
  </property>

  <property>
    <name>yarn.application.classpath</name>
    <value>%HADOOP_CONF_DIR%,%HADOOP_COMMON_HOME%/share/hadoop/common/*,%HADOOP_COMMON_HOME%/share/hadoop/common
/lib/*,%HADOOP_HDFS_HOME%/share/hadoop/hdfs/*,%HADOOP_HDFS_HOME%/share/hadoop/hdfs/lib/*,%HADOOP_MAPRED_HOME%
/share/hadoop/mapreduce/*,%HADOOP_MAPRED_HOME%/share/hadoop/mapreduce/lib/*,%HADOOP_YARN_HOME%/share/hadoop/yarn
/*,%HADOOP_YARN_HOME%/share/hadoop/yarn/lib/*</value>
  </property>
</configuration>
```

Initialize Environment Variables

Run **c:\deploy\etc\hadoop\hadoop-env.cmd** to setup environment variables that will be used by the startup scripts and the daemons.

Format the FileSystem

Format the filesystem with the following command:

```
%HADOOP_PREFIX%\bin\hdfs namenode -format
```

This command will print a number of filesystem parameters. Just look for the following two strings to ensure that the format command succeeded.

```
14/01/18 08:36:23 INFO namenode.FSImage: Saving image file \tmp\hadoop-username\dfs\name\current\fsimage.
ckpt_0000000000000000000 using no compression
14/01/18 08:36:23 INFO namenode.FSImage: Image file \tmp\hadoop-username\dfs\name\current\fsimage.
ckpt_0000000000000000000 of size 200 bytes saved in 0 seconds.
```

Start HDFS Daemons

Run the following command to start the Name{{`Node and Data}}`Node on localhost.

```
%HADOOP_PREFIX%\sbin\start-dfs.cmd
```

To verify that the HDFS daemons are running, try copying a file to HDFS.

```
C:\deploy>%HADOOP_PREFIX%\bin\hdfs dfs -put myfile.txt /

C:\deploy>%HADOOP_PREFIX%\bin\hdfs dfs -ls /
Found 1 items
drwxr-xr-x   - username supergroup          4640 2014-01-18 08:40 /myfile.txt

C:\deploy>
```

Start YARN Daemons and run a YARN job

Finally, start the YARN daemons.

```
%HADOOP_PREFIX%\sbin\start-yarn.cmd
```

The cluster should be up and running! To verify, we can run a simple wordcount job on the text file we just copied to HDFS.

```
%HADOOP_PREFIX%\bin\yarn jar %HADOOP_PREFIX%\share\hadoop\mapreduce\hadoop-mapreduce-example
s-2.5.0.jar wordcount /myfile.txt /out
```

Multi-Node cluster

TODO: Document this

Conclusion

Caveats

The following features are yet to be implemented for Windows.

- Hadoop Security
- Short-circuit reads

Questions?

If you have any questions you can request help from the Hadoop mailing lists. For help with building Hadoop on Windows, send mail to **common-dev@hadoop.apache.org**. For all other questions send email to **user@hadoop.apache.org**. Subscribe/unsubscribe information is included on the linked webpage. Please note that the mailing lists are monitored by volunteers.