Lab: Using Hadoop Client Commands

|  |  |
| --- | --- |
| **Objective:** | To become familiar running HDFS commands and how to view the HDFS file system. |
| **Successful Outcome:** | You will have added and deleted several files and folders in HDFS. |
| **Before You Begin:** | Go into terminal on hadoop-master. |

1. View the hadoop fs Command
   1. From the command line, enter the following command to view the usage of hadoop fs:

# hadoop fs

Notice the usage contains options for performing file system tasks in HDFS, like copying files from a local folder into HDFS, retrieving a file from HDFS, copying and moving files around, and making and removing directories.

* 1. If its valid, enter the following command:

# hdfs dfs

Notice you may get the same usage list as the hadoop fs command. In case you don’t it means that Hadoop isn’t up to the release that successfully has the alias yet.

**NOTE**: The hadoop command is a more generic command that has fewer options than the hdfs command. However, notice hdfs dfs is just an alias for hadoop fs.

1. Understanding the Default Folders in HDFS
   1. Enter the following -ls command to view the contents of the user’s root directory in HDFS, which is /user/root:

# hadoop fs -ls

You do have not a /user/root directory yet, so no output is displayed:

ls: `.': No such file or directory

* 1. View the contents of the /user directory in HDFS:

# hadoop fs -ls /user

Found 4 items

drwxrwx--- - ambari-qa hdfs 0 /user/ambari-qa

drwxr-xr-x - hcat hdfs 0 /user/hcat

drwx------ - hive hdfs 0 /user/hive

drwxrwxr-x - oozie hdfs 0 /user/oozie

*Notice HDFS may have more folders than what we have here.*

* 1. Run the -ls command again, but this time specify the root HDFS folder:

# hadoop fs -ls /

The output should look like:

Found 6 items

drwxrwxrwt - yarn hdfs 0 2013-08-20 13:59 /app-logs

drwxr-xr-x - hdfs hdfs 0 2013-08-20 13:53 /apps

drwxr-xr-x - mapred hdfs 0 2013-08-20 13:57 /mapred

drwxr-xr-x - hdfs hdfs 0 2013-08-20 13:58 /mr-history

drwxrwxrwx - hdfs hdfs 0 2013-08-28 22:03 /tmp

drwxr-xr-x - hdfs hdfs 0 2013-08-28 22:03 /user

* 1. Which user is the owner of the /user folder? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**IMPORTANT**: Notice how adding the / in the -ls command caused the contents of the root folder to display, but leaving off the / attempted to list the contents of /user/root. If you do not specify an absolute path, then all hadoop commands are relative to the user’s default home folder.

1. Create a User Directory in HDFS
   1. Enter the following mkdir command:

# hadoop fs -mkdir /user/root

Notice the root user does not have permission to create this folder.

* 1. Change the permissions to make root the owner of the directory:

$ hadoop fs -chown root:root /user/root

* 1. Verify the folder was created successfully and root is the owner:

$ hadoop fs -ls /user

...

drwxr-xr-x - root root 0 /user/root

* 1. Switch back to the root user:

$ exit

logout

[root@sandbox ~]#

* 1. Now view the contents of /user/root using the following command again:

# hadoop fs -ls

The directory is empty, but notice this time the command worked.

1. Create Directories in HDFS
   1. Enter the following command to create a directory named test in HDFS:

# hadoop fs -mkdir test

* 1. Verify the folder was created successfully:

# hadoop fs -ls

Found 1 items

drwxr-xr-x - root root 0 test

* 1. Create a couple of subdirectories of test:

# hadoop fs -mkdir test/test1

# hadoop fs -mkdir test/test2

# hadoop fs -mkdir test/test2/test3

* 1. Use the -ls command to view the contents of /user/root:

# hadoop fs -ls

Notice you only see the test directory. To recursively view the contents of a folder, use: -ls -R:

# hadoop fs -ls -R

The output should look like:

drwxr-xr-x - root root 0 test

drwxr-xr-x - root root 0 test/test1

drwxr-xr-x - root root 0 test/test2

drwxr-xr-x - root root 0 test/test2/test3

1. Delete a Directory
   1. Delete the test2 folder (and recursively its subcontents) using the –rmr command:

# hadoop fs -rmr test/test2

* 1. Now run the -ls -R command:

# hadoop fs -ls -R

The directory structure of the output should look like:

.Trash

.Trash/Current

.Trash/Current/user

.Trash/Current/user/root

.Trash/Current/user/root/test

.Trash/Current/user/root/test/test2

.Trash/Current/user/root/test/test2/test3

test

test/test1

**NOTE**: Notice Hadoop created a .Trash folder for the root user and moved the deleted content there. The .Trash folder empties automatically after a configured amount of time.

1. Upload a File to HDFS
   1. Now let’s put a file into the test folder. Change directories to /var/log/hadoop/hdfs:

# cd /var/log/hadoop/hdfs

* 1. Notice this folder contains a file named hdfs-audit.log:

# tail hdfs-audit.log

* 1. Run the following -put command to copy hdfs-audit.log into the test folder in HDFS:

# hadoop fs -put hdfs-audit.log test/

* 1. Verify the file is in HDFS by listing the contents of test:

# hadoop fs -ls test

Found 2 items

-rw-r--r-- 3 root root 3744098 test/hdfs-audit.log

drwxr-xr-x - root root 0 test/test1

1. Copy a File in HDFS
   1. Now copy the hdfs-audit.log file in test to another folder in HDFS:

# hadoop fs -cp test/hdfs-audit.log test/test1/copy.log

* 1. Verify the file is in both places by using the -ls -R command on test. The output should look like the following:

# hadoop fs -ls -R test

-rw-r--r-- 3 root root 3744098 test/hdfs-audit.log

drwxr-xr-x - root root 0 test/test1

-rw-r--r-- 3 root root 3744098 test/test1/copy.log

* 1. Now delete the copy.log file using the -rm command:

# hadoop fs -rm test/test1/copy.log

* 1. Verify the copy.log file is in the .Trash folder.

1. View the Contents of a File in HDFS
   1. You can use the -cat command to view text files in HDFS. Enter the following command to view the contents of data.txt:

# hadoop fs -cat test/hdfs-audit.log

* 1. You can also use the -tail command to view the end of a file:

# hadoop fs -tail test/hdfs-audit.log

Notice the output this time is only the last 20 rows of hdfs-audit.log.

1. Getting a File from HDFS
   1. See if you can figure out how to use the get command to copy test/hdfs-audit.log into your local /tmp folder.
2. The getmerge Command
   1. Put the file /var/log/hadoop/hdfs/hadoop-hdfs-namenode-hadoop-master.log into the test folder in HDFS. You should now have two files in test: hdfs-audit.log and hadoop-hdfs-namenode-hadoop-master.log:

# hadoop fs -ls test

Found 3 items

-rw-r--r-- 3 root root 1033038 test/hadoop-hdfs-namenode-hadoop-master.log

-rw-r--r-- 3 root root 3744098 test/hdfs-audit.log

drwxr-xr-x - root root 0 test/test1

* 1. Run the following getmerge command:

# hadoop fs -getmerge test /tmp/merged.txt

* 1. What did the previous command do? Compare the file size of merged.txt with the two log files from the test folder.

1. Specify the Block Size of a File
   1. Change directories to /root/materials

Notice this folder contains a shakespeare.JAR file that is about 4.7MB.

* 1. Put the JAR file into /user/root in HDFS with the name, and assign it a blocksize of 1048576 bytes. **HINT**: The blocksize is defined using the dfs.blocksize property on the command line.
  2. Run the following fsck command on hbase.jar:

# hdfs fsck /user/root/hbase.jar

* 1. How many blocks did this file get broken down in to? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**RESULT**: You should now be comfortable with executing the various HDFS commands, including creating directories, putting files into HDFS, copying files out of HDFS, and deleting files and folders.

**ANSWERS:**

Step 2.4: hdfs

Step 9.1:

# hadoop fs -get test/hdfs-audit.log /tmp

Step 10.3: The two files that were in the test folder in HDFS were merged into a single file and stored on the local file system.

Step 11.2:

hadoop fs -D dfs.blocksize=1048576 -put hbase-0.94.3-bimota-1.2.0.21+HBASE-7644.jar hbase.jar

Step 11.4: The file should be broken down into 5 blocks.