Lab: Using HDFS Commands

About this Lab

Objective:	To become familiar with how files are added to and removed from HDFS and how to view files in HDFS.
File locations:	/root/devph/labs/Lab2.1
Successful outcome:	You will have added and deleted several files and folders in HDFS.
Before you begin:	Your HDP 2.3 cluster should be up and running within your VM.
Related lesson:	***The Hadoop Distributed File System (HDFS)***

Lab Steps

1) View the hdfs dfs Command

a. Open a Terminal in your VM and type "ssh sandbox".
b. Enter the following command to view the usage of hdfs dfs:

```
# hdfs dfs
```

c. Notice that the usage contains options for performing filesystem 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. In this lab, you will perform these commands, and many others, to help you become comfortable with working with HDFS.

2) Create a Directory in HDFS

a. Enter the following `-ls` command to view the contents of the user's root directory in HDFS, which is `/user/root`

```
# hdfs dfs -ls
```

You do not have any files in `/user/root` yet, so no output is displayed.
Run the `-ls` command again, but this time specify the root HDFS folder:

```
# hdfs dfs -ls /
```

The output should look similar to:

```
Found 10 items
drwxrwxrwx   - yarn   hadoop          0 2014-12-16 19:06 /app-logs
drwxr-xr-x   - hdfs   hdfs            0 2014-12-16 19:13 /apps
drwxr-xr-x   - hdfs   hdfs            0 2014-12-16 19:48 /demo
drwxr-xr-x   - hdfs   hdfs            0 2014-12-16 19:07 /hdp
drwxr-xr-x   - mapred hdfs            0 2014-12-16 19:06 /mapred
drwxr-xr-x   - hdfs   hdfs            0 2014-12-16 19:06 /mr-
history
drwxr-xr-x   - hdfs   hdfs            0 2014-12-16 19:37 /ranger
drwxr-xr-x   - hdfs   hdfs            0 2014-12-16 19:08 /system
drwxrwxrwx   - hdfs   hdfs            0 2014-12-16 19:29 /tmp
drwxr-xr-x   - hdfs   hdfs            0 2015-01-12 05:34 /user
```

Important

Notice how adding the / in the `-ls` command caused the contents of the root folder to display, but leaving off the / showed the contents of `/user/root`, which is the default prefix if you leave off the leading / on any of the hadoop commands (assuming the command is run by the "root" user).

b. Enter the following command to create a directory named test in HDFS:

```
# hdfs dfs -mkdir test
```

c. Verify that the folder was created successfully:

```
# hdfs dfs -ls
Found 1 items
drwxr-xr-x   - root root  0   test
```

d. Create a couple of subdirectories for test:

```
# hdfs dfs -mkdir test/test1
# hdfs dfs -mkdir -p test/test2/test3
```

Notice how the `-p` command can be used to create multiple directories. The second command above will fail if you omit the `-p`.

e. Use the `-ls` command to view the contents of `/user/root`:

```
# hdfs dfs -ls
```

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

```
# hdfs dfs -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
```

3) Delete a Directory

a. Delete the test2 folder (and recursively, its subcontents) using the `-rm -R` command:

```
# hdfs dfs -rm -R test/test2
```

b. Now run the `-ls -R` command:

```
# hdfs dfs -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.

4) Upload a File to HDFS

a. Now let's put a file into the test folder. Change directories to `/root/devph/labs/Lab2.1`:

```
# cd /root/devph/labs/Lab2.1/
```

b. Notice this folder contains a file named `data.txt`:

```
# tail data.txt
```

c. Run the following `-put` command to copy data.txt into the test folder in HDFS:

```
# hdfs dfs -put data.txt test/
```

d. Verify that the file is in HDFS by listing the contents of test:

```
# hdfs dfs -ls test
```

The output should look like the following:

```
Found 2 items
-rw-r--r--   1 root root 1529355  test/data.txt
drwxr-xr-x   - root root       0  test/test1
```

5) Copy a File in HDFS

a. Now copy the data.txt file in test to another folder in HDFS using the `-cp` command:

```
# hdfs dfs -cp test/data.txt test/test1/data2.txt
```

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

```
# hdfs dfs -ls -R test
-rw-r--r--   1 root root     1529355 test/data.txt
drwxr-xr-x   - root root           0 test/test1
-rw-r--r--   1 root root     1529355 test/test1/data2.txt
```

c. Now delete the data2.txt file using the `-rm` command:

```
# hdfs dfs -rm test/test1/data2.txt
```

d. Verify that the data2.txt file is in the .Trash folder.

6) View the Contents of a File in HDFS

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

```
# hdfs dfs -cat test/data.txt
```

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

```
# hdfs dfs -tail test/data.txt
```

Notice the output this time is only the last 20 rows of data.txt.

7) Getting a File from HDFS

a. See if you can figure out how to use the get command to copy test/data.txt from HDFS into your local `/tmp` folder.

***Answer*:**

```
# hdfs dfs -get test/data.txt /tmp/
# cd /tmp
# ls data*
```

8) The `getmerge` Command

a. Put the file `/root/devph/labs/demos/small_blocks.txt` into the test folder in HDFS. You should now have two files in test: data.txt and small_blocks.txt.

Answer:

```
# hdfs dfs -put /root/devph/labs/demos/small_blocks.txt test/
```

b. Run the following `getmerge` command:

```
# hdfs dfs -getmerge test /tmp/merged.txt
```

c. What did the previous command do? Did you open the file merged.txt to see what happened?

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

9) Specify the Block Size and Replication Factor

a. Put `/root/devph/labs/Lab2.1/data.txt` into `/user/root` in HDFS, giving it a blocksize of 1,048,576 bytes.

Hint

The blocksize is defined using the dfs.blocksize property on the command line.

Answer:

```
# hdfs dfs -D dfs.blocksize=1048576 -put data.txt data.txt
```

b. Run the following `fsck` command on data.txt:

```
# hdfs fsck /user/root/data.txt
```

c. How many blocks are there for this file?

Answer: The file should be broken down into two blocks.

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.