
Lab: Using HDFS Commands

Exploring HDFS Operations from the CLI and UI

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/rtlabs/datasets
Successful outcome:	Have successfully added and deleted several files and folders in HDFS
Before you begin	n/a
Related lesson:	HDFS Architecture

Switch to the Appropriate User

The cluster being used is the publicly available HDP Sandbox and this lab will be completed as user `maria_dev`. Ensure you are logged into the sandbox as this user. The following steps assume you started a new Terminal.

```
[root@ip-172-30-0-164 ~]# ssh -p 2222 root@127.0.0.1
root@127.0.0.1's password:
Last login: Thu Jun  1 20:58:25 2017 from 172.17.0.1
[root@sandbox ~]# su - maria_dev
[maria_dev@sandbox ~]$ pwd
/home/maria_dev
[maria_dev@sandbox ~]$
```

Lab Steps

- 1) View the `hdfs dfs` Command
 - a. Open a Terminal in your VM.
 - 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

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

```
# hdfs dfs -ls
```

You should have a single file from the earlier lab in `/user/maria_dev`.

```
[maria_dev@sandbox ~]$ hdfs dfs -ls
Found 1 items
-rw-r--r--  1 maria_dev hdfs      185 2017-06-01 20:23 hosts.txt
```

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/maria_dev`, which is the default prefix if you leave off the leading `/` on any of the hadoop commands (assuming the command is run by the “maria_dev” user).

- e. Enter the following command to create a directory named `test` in HDFS:

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

- f. Verify that the folder was created successfully:

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

-
- g. 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`.

- h. Use the `-ls` command to view the contents of `/user/maria_dev`:

```
# 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

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

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

- j. 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/maria_dev
.Trash/Current/user/maria_dev/test
.Trash/Current/user/maria_dev/test/test2
.Trash/Current/user/maria_dev/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

- k. Now let's put a file into the test folder. Change directories to `/root/rtlabs/datasets`:

```
# cd /root/rtlabs/datasets/
```

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

```
# tail data.txt
```

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

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

- n. 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

- o. 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
```

- p. 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
```

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

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

- r. Verify that the `data2.txt` file is in the `.Trash` folder.

6) View the Contents of a File in HDFS

- s. 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
```

- t. 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

- u. 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

- v. Put the file `/root/rtlabs/datasets/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/rtlabs/datasets/small_blocks.txt test/
```

- w. Run the following `getmerge` command:

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

- x. 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

- y. Put `/root/rtlabs/datasets/data.txt` into `/user/studentN` 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
```

- z. Run the following `fsck` command on `data.txt`:

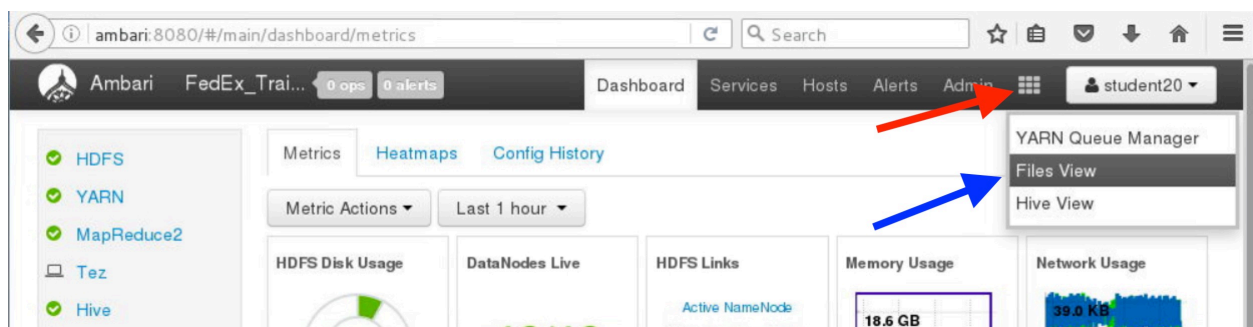
```
# hdfs fsck /user/maria_dev/data.txt
```

- aa. How many blocks are there for this file?

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

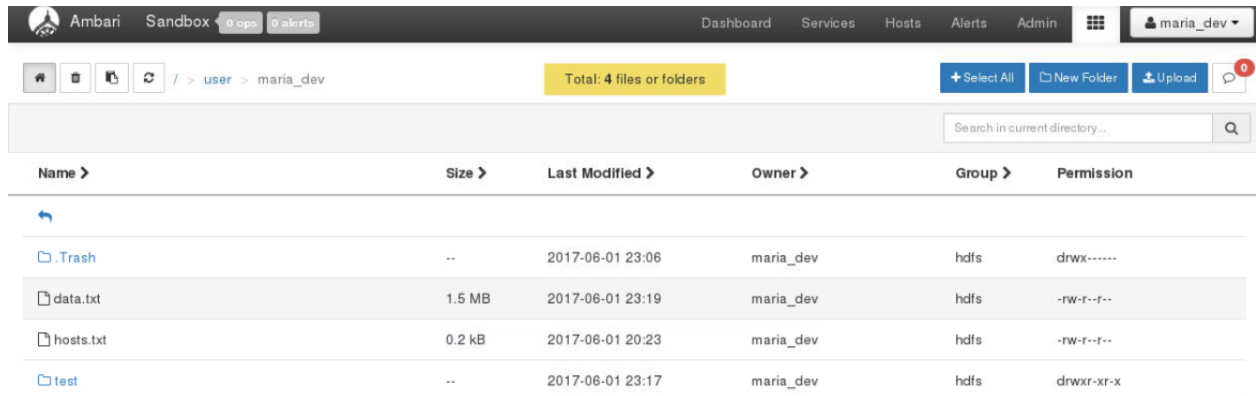
Explore the Ambari Files View

Open up the Firefox browser in your student VM and navigate to `http://127.0.0.1:8001` and log in with user `maria_dev` and password `maria_dev`.



This will log you into the main Dashboard view that presents metrics on how the cluster is running as well as details on all installed components. To launch the Ambari Files View, hover of the icon identified by the red arrow above and then select Files View as identified by the blue arrow.

Navigate to your `/user/maria_dev` home directory.



The screenshot shows the Ambari Files View interface. The top navigation bar includes 'Ambari', 'Sandbox', and 'Alerts' tabs. The main header shows 'Dashboard', 'Services', 'Hosts', 'Alerts', and 'Admin' links. The user 'maria_dev' is logged in. The breadcrumb path is '/ > user > maria_dev'. A yellow box indicates 'Total: 4 files or folders'. There are buttons for '+ Select All', 'New Folder', and 'Upload'. A search bar is present with the text 'Search in current directory...'. Below the search bar is a table with columns: Name, Size, Last Modified, Owner, Group, and Permission. The table lists four items: '.Trash', 'data.txt', 'hosts.txt', and 'test'.

Name	Size	Last Modified	Owner	Group	Permission
.Trash	--	2017-06-01 23:06	maria_dev	hdfs	drwx-----
data.txt	1.5 MB	2017-06-01 23:19	maria_dev	hdfs	-rw-r--r--
hosts.txt	0.2 kB	2017-06-01 20:23	maria_dev	hdfs	-rw-r--r--
test	--	2017-06-01 23:17	maria_dev	hdfs	drwxr-xr-x

Explore the functionality provided in this web-based UI.

Summary

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.