Lab: Using WebHDFS

|  |  |
| --- | --- |
| **Objective:** | To become familiar with the capabilities of WebHDFS and how to use it. |
| **Successful Outcome:** | You will have executed several WebHDFS file commands successfully, include upload, append, list a directory, and download. |
| **Before You Begin:** | Go into terminal on the sandbox. |

1. List a Directory
   1. Using curl, view the contents of the /user/root directory in HDFS:

# curl -i "http://sandbox:50070/webhdfs/v1/user/root?op=LISTSTATUS"

* 1. You should see a 200 OK response, along with a JSON object containing the files and directories in your /user/root folder:

HTTP/1.1 200 OK

Cache-Control: no-cache

Expires: Thu, 14 Nov 2013 14:35:42 GMT

Date: Thu, 14 Nov 2013 14:35:42 GMT

Pragma: no-cache

Expires: Thu, 14 Nov 2013 14:35:42 GMT

Date: Thu, 14 Nov 2013 14:35:42 GMT

Pragma: no-cache

Content-Type: application/json

Transfer-Encoding: chunked

Server: Jetty(6.1.26)

{"FileStatuses":{"FileStatus":[

{"accessTime":0,"blockSize":0,"childrenNum":0,"fileId":17321,"group":"hadoop","length":0,"modificationTime":1384408800076,"owner":"root","pathSuffix":".Trash","permission":"700","replication":0,"type":"DIRECTORY"},

{"accessTime":1384219125588,"blockSize":134217728,"childrenNum":0,"fileId":17331,"group":"hadoop","length":861,"modificationTime":1384219125967,"owner":"root","pathSuffix":"constitution.txt","permission":"644","replication":3,"type":"FILE"},

...

]}}

1. Make a New Directory
   1. Use WebHDFS to make a new subdirectory of /user/root in HDFS named history:

# curl -i -X PUT "http://sandbox:50070/webhdfs/v1/user/root/history?op=MKDIRS"

Verify the file in Hue: <http://localhost:8000/filebrowser/#/user/history>

If you get an AccessControlException, you need to add the user.name property to the URL:

# curl -i -X PUT "http://sandbox:50070/webhdfs/v1/user/root/history?op=MKDIRS&user.name=root"

* 1. Verify the history directory was created successfully:

# hadoop fs -ls

1. Upload a File
   1. The first step to uploading a file is to create a path on the NameNode. As part of the REST contract, the NameNode will respond with a *307 TEMPORARY\_REDIRECT* to the actual DataNode that the blocks should go to:

# curl -i -X PUT "http://sandbox:50070/webhdfs/v1/user/root/history/constitution.txt?op=CREATE&blocksize=1048576"

* 1. Use the temporary redirect URL that the NameNode provides in the response above to submit the file to the DataNode. For example, the command shown here puts the file onto node4, but you should copy-and-paste the URL from the response of the previous step:

# curl -i -PUT -T constitution.txt "http://sandbox:50075/webhdfs/v1/user/root/history/constitution.txt?op=CREATE"

* 1. Verify the file was uploaded successfully:

# hadoop fs -lsr history

Found 1 items

-rwxr-xr-x 3 root hadoop 44841 history/constitution.txt

1. Upload a Large File
   1. In this step, you will upload a larger file, one that spans multiple blocks. Start by changing directories to /root/materials.

You should have a large file in materials. Pick one.

* 1. Ask the NameNode to create a file named big.txt in /user/root:

# curl -i -X PUT "http://sandbox:50070/webhdfs/v1/user/root/big.txt?op=CREATE&blocksize=1048576"

* 1. Using the URL provided by the previous command, upload test\_data into HDFS, and then verify the upload worked successfully.

1. Append to an Existing File
   1. Appending a file is similar to creating a file - it is a two-step process. Using WebHDFS, append the local file constitution.txt to big.txt in HDFS.
2. Retrieve a File
   1. Use WebHDFS to retrieve the file constitution.txt.
   2. Retrieve big.txt from the 1,000,000th byte offset and get 1048576 bytes (1MB). Pipe the result to a local file named big\_partial.txt.

**RESULT**: You have seen how to use WebHDFS to execute a variety of HDFS commands over HTTP using RESTful web Services.

**SOLUTION to 6.2**:

curl -i -L "http://sandbox:50070/webhdfs/v1/user/root/big.txt?op=OPEN&offset=1000000&length=1048576" > big\_partial.txt