HDFS – Hadoop File System

HDFS adalah sebuah sistem penyimpanan data terdistribusi yang memungkinkan untuk menyimpan data berukuran sangat besar. Sistem file ini dikembangkan berdasarkan konsep dari GFS. HDFS mempunyai karakteristik sebagai sistem terdistribusi yang memiliki kapasitas berskala besar dan handal, namun demikian proses instalasinya dikenal mudah dan pengoperasiannya cukup sederhana. HDFS dapat dijalankan pada mode pseudo-distributed yang berarti dapat digunakan hanya dalam satu node saja maupun pada mode fully distributed yang berarti untuk digunakan dalam beberapa node. Saat ini semakin banyak organisasi skala menengah maupun besar yang sudah mengadopsi, seperti yang sudah dilakukan oleh Yahoo!, IBM, Facebook, Twitter, Rakuten, Amazon, dan NTT Docomo. HDFS bukan database, sehingga tidak cocok pada kondisi dimana ada tuntutan akan latency yang rendah dari proses membaca atau menulis dan terdapat banyak file yang berukuran kecil. HDFS tidak memiliki fitur pengindeksan, tidak ada akses file secara acak, dan tidak mendukung SQL. Sehingga apabila diperlukan kemampuan tambahan seperti layaknya database maka perlu menggunakan HBase.

Perintah-perintah HDFS:

1. appendToFile

Usage: hdfs dfs -appendToFile <localsrc> ... <dst>

Append single src, or multiple srcs from local file system to the destination file system. Also reads input from stdin and appends to destination file system.

2. cat

Usage: hdfs dfs -cat URI [URI ...]

Copies source paths to stdout.

3. chgrp

Usage: hdfs dfs -chgrp [-R] GROUP URI [URI ...]

Change group association of files. The user must be the owner of files, or else a super-user. Additional information is in the Permissions Guide.

4. chmod

Usage: hdfs dfs -chmod [-R] <MODE[,MODE]... | OCTALMODE> URI [URI ...]

Change the permissions of files. With -R, make the change recursively through the directory structure. The user must be the owner of the file, or else a super-user. Additional information is in the Permissions Guide.

5. chown

Usage: hdfs dfs -chown [-R] [OWNER][:[GROUP]] URI [URI]

Change the owner of files. The user must be a super-user. Additional information is in the Permissions Guide.

6. copyFromLocal

Usage: hdfs dfs -copyFromLocal <localsrc> URI

7. copyToLocal

Usage: hdfs dfs -copyToLocal [-ignorecrc] [-crc] URI <localdst>

Similar to get command, except that the destination is restricted to a local file reference.

8. count

Usage: hdfs dfs -count [-q] <paths>

Count the number of directories, files and bytes under the paths that match the specified file pattern. The output columns with -count are: DIR_COUNT, FILE_COUNT, CONTENT_SIZE FILE_NAME

9. cp

Usage: hdfs dfs -cp [-f] URI [URI ...] <dest>

Copy files from source to destination. This command allows multiple sources as well in which case the destination must be a directory.

10. du

Usage: hdfs dfs -du [-s] [-h] URI [URI ...]

Displays sizes of files and directories contained in the given directory or the length of a file in case its just a file.

11. dus

Usage: hdfs dfs -dus <args>

Displays a summary of file lengths. This is an alternate form of hdfs dfs -du -s.

12. expunge

Usage: hdfs dfs -expunge

Empty the Trash. Refer to the HDFS Architecture Guide for more information on the Trash feature.

13. get

Usage: hdfs dfs -get [-ignorecrc] [-crc] <src> <localdst>

Copy files to the local file system. Files that fail the CRC check may be copied with the -ignorecrc option. Files and CRCs may be copied using the -crc option.

14. getfacl

Usage: hdfs dfs -getfacl [-R] <path>

Displays the Access Control Lists (ACLs) of files and directories. If a directory has a default ACL, then getfacl also displays the default ACL.

15. getmerge

Usage: hdfs dfs -getmerge <src> <localdst> [addnl]

Takes a source directory and a destination file as input and concatenates files in src into the destination local file. Optionally addnl can be set to enable adding a newline character at the end of each file.

16. ls

Usage: hdfs dfs -ls <args>

For a file returns stat on the file with the following format:

permissions number_of_replicas userid groupid filesize modification_date modification_time filename
For a directory it returns list of its direct children as in Unix. A directory is listed as:

permissions userid groupid modification_date modification_time dirname

17. lsr

Usage: hdfs dfs -lsr <args>

18. mkdir

Usage: hdfs dfs -mkdir [-p] <paths>

Takes path uri's as argument and creates directories.

19. moveFromLocal

Usage: dfs -moveFromLocal <localsrc> <dst>

Similar to put command, except that the source localsrc is deleted after it's copied.

20. moveToLocal

Usage: hdfs dfs -moveToLocal [-crc] <src> <dst>

Displays a "Not implemented yet" message.

21. mv

Usage: hdfs dfs -mv URI [URI ...] <dest>

Moves files from source to destination. This command allows multiple sources as well in which case the destination needs to be a directory. Moving files across file systems is not permitted.

22. put

Usage: hdfs dfs -put <localsrc> ... <dst>

Copy single src, or multiple srcs from local file system to the destination file system. Also reads input from stdin and writes to destination file system.

rm
Usage: hdfs dfs -rm [-skipTrash] URI [URI ...]

Delete files specified as args. Only deletes non empty directory and files. If the -skipTrash option is specified, the trash, if enabled, will be bypassed and the specified file(s) deleted immediately. This can be useful when it is necessary to delete files from an over-quota directory. Refer to rmr for recursive deletes.

rmr
Usage: hdfs dfs -rmr [-skipTrash] URI [URI ...]

Recursive version of delete. If the -skipTrash option is specified, the trash, if enabled, will be bypassed and the specified file(s) deleted immediately. This can be useful when it is necessary to delete files from an over-quota directory.

setfacl
Usage: hdfs dfs -setfacl [-R] [-b|-k -m|-x <acl_spec> <path>]|[--set <acl_spec> <path>]

Sets Access Control Lists (ACLs) of files and directories.

setrep
Usage: hdfs dfs -setrep [-R] [-w] <numReplicas> <path>

Changes the replication factor of a file. If path is a directory then the command recursively changes the replication factor of all files under the directory tree rooted at path.

Options:

The -w flag requests that the command wait for the replication to complete. This can potentially take a very long time.
The -R flag is accepted for backwards compatibility. It has no effect.
Example:

hdfs dfs -setrep -w 3 /user/hadoop/dir1
Exit Code:

Returns 0 on success and -1 on error.

stat
Usage: hdfs dfs -stat URI [URI ...]

Returns the stat information on the path.

23. tail

Usage: hdfs dfs -tail [-f] URI

Displays last kilobyte of the file to stdout.

24. test

Usage: hdfs dfs -test -[ezd] URI

25. text

Usage: hdfs dfs -text <src>

Takes a source file and outputs the file in text format. The allowed formats are zip and TextRecordInputStream.

26. touchz

Usage: hdfs dfs -touchz URI [URI ...]

Create a file of zero length.
