Linux Commands for DevOps Warriors

Quick guide to understand Linux Commands

[Vishweshwaran M J](#)

This article will help in understanding most of the important and majorly used Linux commands that would be required for a DevOps Engineer.

To execute these commands one can either use any Linux machine / virtual machine / online Linux terminal to quickly start working with the commands.

System Info Commands

`hostname` - shows the name of the system host.

`hostid` – shows the host id of the system assigned by the OS.

`date` - shows the current date and time in UTC format.

`whoami` - shows the currently logged-in username of the terminal.

`uptime` - shows the elapsed time duration since the machine logged in.

`uname` - unix name.

`clear` - clears the screen.

`history` - lists all the commands executed until now.

`sudo` - Super User Do.

`echo $?` - shows the exit status of the last executed command (0 - success, 1-255 - error/failure).

`shutdown -r now` - restart the machine immediately (-r restart).

`printenv` - displays all the environment variables of the Linux system.

`last` - shows previous logins in the Linux system.

Directory Commands

`pwd` - shows the present working directory (abbr. Print Working Directory).

`cd` - change directory.

- `cd ..` - changes to its parent directory (i.e) one level up.
- `cd <dirName>` - change to the directory mentioned.

- **`cd ~`** or **`cd`** – changes to the currently logged in user's home directory.
- **`cd ../..`** – changes the directory two levels up.
- **`cd –`** – changes to the last working directory.

`mkdir` – make directory.

- **`mkdir <dirName>`** – creates the directory.
- **`mkdir –p <pathOftheDir>`** – creates directory with its parent directories if it does not exists (-p parent).

`ls` – lists the files & folders of the directory you are in.

- **`ls –a`** – lists all files & folders along with hidden files (-a all).
- **`ls –al`** – lists all files & folders along with hidden files in a formatted manner (-l long listing format).

File Commands

`touch` – creates an empty file or updates timestamp of the existing file.

- **`touch <fileName>`** – creates a single empty file.
- **`touch <file1> <file2>`** – creates file1, file2 empty files.

`cat` – concatenates and displays the contents of files.

- **`cat <fileName>`** – displays the contents of the file.
- **`cat > <fileName>`** – creates a new file, allows to input content interactively and redirects inputted content to the created file (> redirection operator).

`head <fileName>` – displays first 10 lines of the file by default.

- **head –n 5 <fileName>** - displays first 5 lines of the file (-n number)

tail <fileName> - displays the last 10 lines of the file by default.

- **tail –n 5 <fileName>** - displays last 5 lines of the file (-n number).
- **tail –F <fileName>** - displays contents of the file in real-time even when the file is rotated or replaced (used for log file monitoring).

less <fileName> - used to view large files (log files) in a paginated manner.

rm - remove command.

- **rm <fileName>** - removes the file.
- **rm –r <dirName>** - removes files & folders of directory recursively (-r recursive).
- **rm –rf <dirName>** - force remove the files & folders of directory recursively (-f force).

cp - copy command.

- **cp <source> <destination>** - copy the files and folders from source to destination.
- **cp –r <dir1> <dir2>** - copy dir1 directory to dir2 directory recursively (-r recursive).

mv - move or rename command.

- **mv <fileName> <newFileName>** - renames the file to new name.
- **mv <oldFilePath> <newFilePath>** - moves the file to new path.

File Permission Commands

`ls –l <pathOfFileName>` – shows the permissions of the file.

`ls –ld <dirNamePath>` – shows the permissions of the directory.

`chmod <octalNumber> <fileName>` – changes mode/permissions of the file.

`chmod <octalNumber> –R <dirName>` – changes mode/permissions of the directory recursively.

`chown <newUser> <fileName>` – changes the user ownership of a file.

`chown <newUser>:<newGroup> <fileName>` – changes the user & group ownerships of a file.

`chgrp <groupName> <fileName/dirName>` – updates the group name for file/directory.

`getfacl <fileName/dirName>` – shows the file/directory access control list.

`setfacl –m u:<userName>:rwx <fileName/dirName>` - modifies the current acl of the file/directory.

`setfacl –x u:<userName>: <fileName/dirName>` – removes the acl permissions for the file/directory.

`setfacl –m g:<groupName>:rwx <fileName/dirName>` – modifies the group acls for the file/directory.

`setfacl –x g:<groupName>: <fileName/dirName>` – removes the group acl permissions for the file/directory.

File Permission Octal Numbers

read (r) — 4, write (w)- 2, execute (x) — 1

- *Sum the numbers to create an octal number for providing permission to a file/directory.*

User Management Commands

`useradd` - creates a user account.

- `useradd <userName>` - creates user account without home & mail spool directories.
- `useradd —m <userName>` - creates user account with home & mail spool directories.

`passwd <userName>` - creates a password for the user and stores it in /etc/shadow file.

`userdel` - user delete.

- `userdel <userName>` - deletes the user from the system.
- `userdel —r <userName>` - deletes the user from the system along with home and mail spool directories (-r remove).

`/etc/passwd` - stores information about user accounts.

- `cat /etc/passwd` - displays the complete list of users on that machine.

`/etc/shadow` - stores the password for users in an encrypted format.

- `cat /etc/shadow` - displays the complete list of user passwords on that machine.

`su` - substitute user.

- `su <userName>` - switches to the user mentioned.
- `exit` - to logout from that user.

`usermod` - modify user.

- `usermod –aG <groupName> <userName>` - adds the user to another group (-aG append the user to the group without removing from other groups).

`chsh` - change shell.

- `chsh –s /bin/bash <user>` - changes the shell to bash for the user.
- `chsh –s /bin/sh <user>` - changes the shell to sh for the user.

Group Management Commands

`groupadd <groupName>` - creates the group.

`groupdel <groupName>` - delete the group.

`/etc/group` - stores the information of the groups.

- `cat /etc/group` - displays the complete list of groups on that machine.

`gpasswd <groupName>` - creates a password for the group.

- `gpasswd –a <userName> <groupName>` - adds the user to the group.
- `gpasswd –d <userName> <groupName>` - removes the user from the group.
- `gpasswd –M <userName1>,<userName2>,<userName3> <groupName>` - adds multiple users to the group and removes the existing ones of the group.

Searching Commands

`locate` - used to search for files/directories based on names.

- `sudo updatedb` - updates the database so the results are up-to-date.
- `locate <fileName/dirName>` - locates the file/directory and displays the path.

GREP Command — Global Regular Expression Print

- `grep <textToSearch> <fileName>` - used to find text patterns within files.
- `grep –i <textToSearch> <fileName>` - used to find text patterns within the file ignoring the case (-i ignore case).
- `grep –v <textToSearch> <fileName>` - used to find non matching lines of text patterns (-v invert-match).
- `grep –l <textToSearch> <fileNames>` - used to display the matching string file names.

Find Command

- `find . –name <fileName>` - finds the mentioned file if available in the current directory (.(period) represents current directory).
- `find <dirName> –name <fileName>` - finds the mentioned file in the directory.
- `find <dirName> –perm 754` - finds the files in the directory having 754 permission.

Hardware Infomation Commands

`free -h` - shows systems memory information (-h human readable format).

`df -h` - shows the disk space usage of mounted file systems.

`du` - disk usage.

- `du -h` - displays disk usage information (-h human-readable format).
- `du -sh` - displays the total size of the directory instead of individual files in human-readable format (-s summarize).
- `du -sh <fileName/dirName>` - displays the total size of the file/directory.

Network Commands

`ping <hostName>` - tests the reachability & responsiveness of the remote host.

`dig <domainName>` - shows DNS information of the domain.

`wget <url>`- used to retrieve/download files from the internet.

`curl` - client URL.

- `curl <url>` - used to retrieve/download files from the internet.

`ifconfig` - display available network interfaces.

`ip addr` - display and manipulate network interface info.

`curl ifconfig.me` - shows the public ip address of the machine.

`netstat —antp`- shows all tcp open ports (-a all, t-tcp, n-active, p protocol).

`traceroute <url>` - traces the route using packets from source to destination host.

Process Information Commands

`ps` - process status.

- `ps` - shows the currently running process.
- `ps —u <userName>`- shows the process of the username
- `ps —ef` - shows all the processes of the system.
- `top` - shows the real-time, dynamic view of the running processes of a system.
- `kill <pid>` - gracefully terminates the process pid.
- `pgrep <processName>` - shows process id of processes based on name/other criteria.
- `bg` - background, sends the process to the background & continues execution without interruption.
- `fg` - foreground, brings the process to the foreground and makes it an active process.
- `nohup` - no hangup, runs command/script in the background even after the terminal is closed or the user logs out.

Archiving File Commands

`tar` - tape archive.

- `tar —cvf <fileName> <directory>` - creates the tar file with the fileName for the directory mentioned (-c create, -v verbose, f - output

file name).

- `tar -xvf <sourceTarFileName> -C <destinationDir>` - puts the extracted files into the destination directory (-x extract, v verbose, -f source tar file name).

Ubuntu Package related Commands

`apt` - Package Manager for Debian-based Linux distributions Eg: Ubuntu.

- `apt` - a newer version of the package manager with colorized output, progress bar and additional functions.
- `apt-get` - older version and basic package manager.

`apt update` - updates the package list.

`apt list --installed` - lists all the installed packages.

- `apt list --installed <packageName>` - shows the package name if it's installed.

`apt show <packageName>` - shows information about a package mentioned.

`apt search <packageName>` - searches and shows the list of packages.

`apt install <packageName>` - installs the required package.

`apt remove <packageName>` - removes the required package.

`apt purge <packageName>` - removes the required package along with its config files.

Misc Commands

- `echo "sample text" | grep text` - the output of the first command is passed as an input to the second command using the pipe (|) symbol.
- `ls -l | tee file.txt` - redirects the list to the file.txt and simultaneously displays it in the terminal.
- `echo "sample text" > <fileName>` - write the content to the file mentioned by overwriting the existing content (> redirection operator).
- `echo "new sample text" >> <fileName>` - appends the contents to the file mentioned without overwriting the existing content (>> redirection operation).
- `stat <fileName/dirName>` - shows detailed information about the file or directory.
- `cron` - system daemon for managing scheduled tasks.
- `tree` - tree representation of files and directories of a specific directory.

This blog page covers most of the majorly used linux commands for DevOps Warriors and is a growing document. Commands will be added as the days go and when I find some interesting commands on the same.

Do follow me for more such content related to DevOps world.

Cheers, Vish. Happy Learning!!