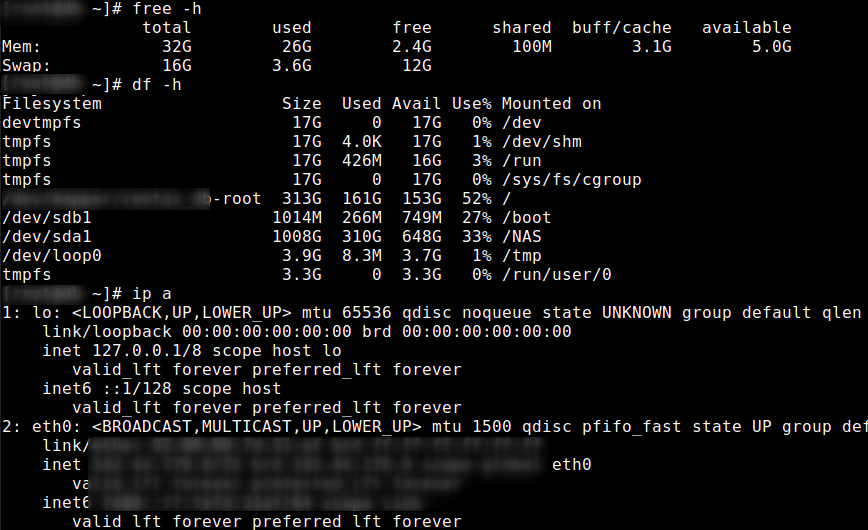
**[Linux Commands frequently used by Linux Sysadmins – Part 1](https://haydenjames.io/linux-commands-frequently-used-by-linux-sysadmins-part-1/" \o "Linux Commands frequently used by Linux Sysadmins – Part 1)**

Some people consider Linux as a being a complicated operating system, geared toward expert users only. However, as a free and open-source operating system, Linux is actually geared toward all users. Allowing both end-users and admins, access to understanding as much or as little as they desire.

### Whether you are completely new to Linux or an experienced admin,you’ll find yourself using these commands frequently. (Part 1 of 5)

As such, the commands listed below should help you better navigate, manage, and search Linux systems. The Linux commands listed are also useful in grabbing more information when troubleshooting. These command-line tips apply to all Linux systems and distros, both on [virtual and physical machines](https://haydenjames.io/home-lab-beginners-guide-hardware/).



## 1. List and show all IP addresses associated with all network interfaces.

You may know this as the much longer command ip address show.

ip a

Example output:

[root@web ~]# ip a

...

2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

...

inet xxx.xx.xxx.xx/32 brd xxx.xx.xxx.xx scope global eth0

...

3: eth1: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

...

inet 192.168.0.2/24 brd 192.168.0.255 scope global eth1

## 2. List non-hidden files and subfolders in the current directory.

Use -R for recursive, -a to include hidden files or -l to use per-line listing format. The cd command is discussed in [Part 2](https://haydenjames.io/linux-commands-frequently-used-by-linux-sysadmins-part-2/).

ls

example output:

[root@web /]# ls -l

total 36

drwx--x--x. 5 root root 76 Aug 11 03:28 backup

lrwxrwxrwx. 1 root root 7 Oct 30 2019 bin -> usr/bin

dr-xr-xr-x. 5 root root 4096 Jun 26 05:45 boot

drwxr-xr-x. 20 root root 3120 Jun 6 06:07 dev

drwxr-xr-x. 99 root root 12288 Aug 12 07:40 etc

drwxr-xr-x. 8 root root 146 Feb 17 00:04 home

...

## 3. Display disk space usage.

Use -i to list inode information instead of block usage. Use -h to print sizes in powers of 1024 (e.g., 1023M).

df -h

Example output:

[user@server ~]# df -h

Filesystem Size Used Avail Use% Mounted on

devtmpfs 17G 0 17G 0% /dev

/dev/mapper/root 313G 161G 153G 52% /

/dev/sdb1 1014M 266M 749M 27% /boot

...

Also, popular is the du command. Used to estimate file space usage under a particular directory or files on the system.

## 4. Display memory usage.

Use -h to show all output fields automatically scaled to the shortest three digit unit and display the units of print out. Or use -m to display the amount of memory in mebibytes.

free -m

Example output:

[root@web /]# free -h

total used free shared buff/cache available

Mem: 32G 2.0G 24G 1.6G 6.1G 28G

Swap: 16G 64M 16G

[root@web /]# free -m

total used free shared buff/cache available

Mem: 33016 2021 24746 1640 6248 28957

Swap: 16639 64 16575

## 5. Run multiple commands in one line using ; .

;

Example:

sudo apt update ; apt upgrade

Then optionally you can add the final command it to a bash script.

## 6. Find large files.

Or install [ncdu](https://dev.yorhel.nl/ncdu" \t "_blank) and execute from the command line. Also, see [the locate command in part 3](https://haydenjames.io/linux-commands-frequently-used-by-linux-sysadmins-part-3/).

find [directory] -size [set minimum size]

Example:

find /home/ -size +1000000k

## 7.  Display a tree of processes.

Add -P to show PIDs.  PIDs are shown as decimal numbers in parentheses after each process name.

pstree -P

Example output:

xxx@host:~$ pstree

systemd─┬─accounts-daemon───2\*[{accounts-daemon}]

├─agetty

├─apache2───3\*[apache2───31\*[{apache2}]]

├─atd

├─cron

...

├─networkd-dispat───{networkd-dispat}

├─php-fpm7.4───5\*[php-fpm7.4]

...

Also, see the [ps command in part 3](https://haydenjames.io/linux-commands-frequently-used-by-linux-sysadmins-part-3/" \t "_blank).

## 8. Show listing of last logged in users.

last

Example output:

[root@server ~]# last

root pts/0 xxx.xxx.xxx.xxx Wed Aug 12 08:29 still logged in

root pts/0 xxx.xx.xxx.xx Wed Jul 29 10:52 - 12:13 (01:21)

root pts/0 xxx.xx.xx.xx Mon Jul 27 23:11 - 00:10 (00:58)

root pts/0 xxx.xxx.xx.xx Wed Jul 15 23:46 - 00:01 (00:15)

## 9. Show list of currently logged in user sessions.

w

Example output:

root@host:~$ w

13:08:25 up 72 days, 8:00, 1 user, load average: 0.01, 0.08, 0.08

USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT

xxxx-user pts/0 xxx.xx.xxx.xx 13:04 0.00s 0.00s 0.00s w

## 10. Search a file for a pattern of characters, then display all matching lines.

grep

Example:

grep [options] pattern [files]

Example, grep directory recursively:

grep -r "texthere" /home/

Example, grep the word printf:

grep printf /path/filename.txt

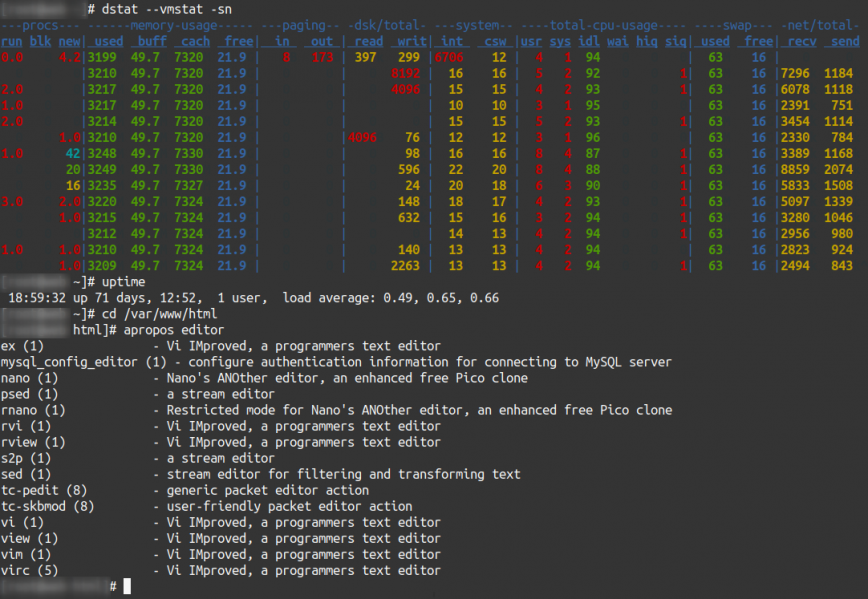
Example, find previously used commands which include systemctl

history | grep systemctl

Example, find the last login(s) for username:

last | grep username

Let’s jump into another set of commands and command-line tools often used for analysis, troubleshooting, and administration of Linux systems. Listed below are some of the commands I’ve used over the past week. In this and future series, I will group related commands together, as much as possible, and also add a table of contents to interlink all five parts.



## 1. Command-line Linux system monitoring.

Very often it’s useful to view Linux system performance. He’s a shortlist for quickly accessing system performance details via the command line.

* uptime – shows system uptime and load average.
* [top](https://haydenjames.io/linux-top-customize-it/) – shows an overall system view.
* vmstat – shows system memory, processes, interrupts, paging, block I/O, and CPU info.
* [htop](https://haydenjames.io/htop-quick-guide-customization/) – interactive process viewer and manager.
* dstat – view processes, memory, paging, I/O, CPU, etc., in real-time. All-in-one for vmstat, iostat, netstat, and ifstat.
* iftop – network traffic viewer. Ex: iftop -i eth0
* nethogs – network traffic analyzer.
* iotop – interactive I/O viewer. Get an overview of storage r/w activity.
* iostat – for storage I/O statistics.
* netstat – for network statistics.
* ss – utility to investigate sockets.

Here are some related articles I’ve written in the past covering command-line system monitoring tools:

* [top – Here’s how to customize it](https://haydenjames.io/linux-top-customize-it/).
* [atop – For Linux server performance analysis](https://haydenjames.io/use-atop-linux-server-performance-analysis/).
* [Glances](https://haydenjames.io/alternatives-top-htop/) and [nmon –](https://haydenjames.io/alternatives-top-htop/" \o "htop and top Alternatives: Glances, nmon)[htop and top Alternatives:](https://haydenjames.io/alternatives-top-htop/)
* [htop – Quick Guide & Customization](https://haydenjames.io/htop-quick-guide-customization/).
* [bashtop – the ‘cool’ top alternative](https://haydenjames.io/bashtop/).

## 2. ssh – secure command-line access to remote Linux systems.

Other than my [Homelab](https://haydenjames.io/home-lab-beginners-guide-hardware/), most of my [Sysadmin related work](https://haydenjames.io/web-server-services/) is performed remotely. With clients’ server locations in the Americas, Europe, Asia, and Australia. Access command-line on a remote server, you can [use the ssh command](https://www.ssh.com/ssh/command/).

The ssh command provides secure encrypted connections between local and remote hosts. ssh is used for terminal access, file transfers ([sftp](https://www.ssh.com/ssh/sftp/)), and [tunneling](https://www.ssh.com/ssh/tunneling/) other applications.

ssh -i [key\_file] [user]@[hostname/ip] -p [port]

-i identity\_file, selects a file from which the identity (private key) for public key authentication is read.  
-p port, is used to select which port to connect to on the remote host.

For example:

ssh -i path/to/key\_file root@ip\_address -p 2222

SSH can also be run securely on a graphical desktop using terminal emulators such as [Terminator](https://gnome-terminator.org/).

Also, take a look at [telnet](https://wiki.archlinux.org/index.php/Telnet) and [scp – Securely Copy Files Using SCP with examples](https://haydenjames.io/linux-securely-copy-files-using-scp/" \o "SCP Linux – Securely Copy Files Using SCP examples).

In **Part 3**, we will take a look at using [rsync](https://rsync.samba.org/" \t "_blank).

## 3. sudo – execute commands with administrative privilege.

The sudo (**S**uper **U**ser **DO**) command in Linux is regularly used as a prefix to commands which only the superuser (root) has permission to run. If you are not logged in with the root user, you can use the sudo  command to preface a command in order to run that command with root privileges. For example:

$ sudo apt install terminator

Once you run a sudo command, as a security precaution, you will be prompted to enter the password of the current non-root user session. (not the root password)

See also [Red Hat’s Gaining Privileges documentation](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/system_administrators_guide/chap-gaining_privileges).

## 4. cd – Directory navigation.

The cd command allows you to change directories. To navigate into the root directory, use:

cd /

Navigate to your home directory use:

cd

or

cd ~

Navigate up one directory level:

cd ..

Navigate to the previous directory (or back):

cd -

To navigate to a multi-level directory use:

cd /path/to/directory/

The pwd command shows your current directory location. pwd stands for “print working directory”. Also, see the [ls command from part 1](https://haydenjames.io/linux-commands-frequently-used-by-linux-sysadmins-part-1/).

Example:

hydn@ubuntu:~$ pwd

/home/hydn

## 5. cp – Copying files and folders

The cp command will create a copy of a file for you. For example:

cp file1 file1\_backup

This command will create a copy of “file1” named “file1\_backup”, and “file1” will remain unmodified.

To copy a directory and it’s contents use the command with -r recursively:.

cp -r directory1 directory1\_copy

Recursively means copying the directory and all its files and subdirectories and all their files and subdirectories and so on.

## 6. mv – moving files and folders.

The mv command is used to move a file from its location to a different location or will rename a file. For example:

mv file1 file0

This will rename the “file1” to “file0”.

Another example:

mv file1 /home/user/Desktop/

will move “file1” to your Desktop directory, but it will not rename it.

To move AND rename you simply specify a new file. For example:

mv file1 /home/user/Desktop/file0

You can substitute *~* in place of /home/user/ which is faster for home directory related commands. For example:

mv file1 ~/Desktop/

When using the sudo command or logged in as root, then ~ will be the home directory of root.

## 7. Removing files and folders.

Use the rm command to remove/delete a file in a directory. For example:

rm file1

Use the rmdir command to delete an empty directory.

rmdir /path/to/empty/dir/

Use rm -r to delete a directory and all of its contents recursively:

rm -r /path/to/dir/

You can also create an empty file with touch command or cat to echo, can’t list them all:

touch filename

Or create a new directory you can use the mkdir command:

mkdir newdir/

In Part 3, we will look at creating and editing files with vi and nano.

## 8. man – for reading system reference manuals.

The man command displays a manual for any terminal commands we need help or info. All man pages include but are not limited to the following standard sections: Name, Synopsis, Description, Examples, and See Also.

man [command]

Example:

man ssh

## 9. apropos – Search man page names and descriptions.

The apropos command searches the man page names and descriptions. It can be used to find what commands to use for a certain job.

apropos [keyword]

For example, search for available commands related to firewall:

apropos firewall

## 10. Search your package manager for available packages.

If you’d like to search for packages not already installed on your system, you can use the following commands to search for packages using your Linux distro’s package manager:

Ubuntu/Debian:

apt search [keyword]

or

apt-cache search [keyword]

For example:

hydn@ubuntu:~$ apt-cache search "nginx web"

nginx-core - nginx web/proxy server (standard version)

fcgiwrap - simple server to run CGI applications over FastCGI

libnginx-mod-http-dav-ext - WebDAV missing commands support for Nginx

libnginx-mod-http-ndk - Nginx Development Kit module

nginx-extras - nginx web/proxy server (extended version)

nginx-full - nginx web/proxy server (standard version)

nginx-light - nginx web/proxy server (basic version)

CentOS/Fedora/RHEL:

dnf search [keyword]

OpenSuse:

zypper se [keyword]