Raspberry Pi Commands Cheat Sheet

74 Raspberry Pi Commands

From

RaspberryTips

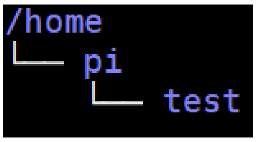
FILES MANAGEMENT

These commands are the basics that every Linux beginner should learn to browse the Linux files tree from a terminal

Reminder:

The Linux files organization is a tree, starting at /
Each subfolder adds a new level under /

For example, on the image you can see the tree for this folder : /home/pi/test



```
/home
└── pi
    └── test
```

CD <FOLDER>

Changes directory, go to the specified folder

Absolute path:	cd /home/pi/test
Relative path:	cd test

NB: "Absolute" is when you use the entire path
For "relative" you only enter the path from your current directory (in the second example, you need to already be in the /home/pi folder)

MKDIR <FOLDER>

Creates a new subfolder in the current or specified path

Current directory:	mkdir test
Specific:	mkdir /home/pi/test

NB: The first example create a folder in your current directory (relative path)
The second one create a new directory in the exact parameter (absolute path)

MV <SRC> <TARGET>

Moves a file or directory to another location (cut/paste)

Move a file:	mv test.txt /home/pi
Move a folder:	mv /home/pi/test /home/pi/test2

NB: The mv command is always in recursive mode

MORE <FILENAME>

Displays the content of the file, page per page, from the beginning

Absolute path:	more test.txt
Relative path:	more /home/pi/test.txt

NB: For long files, you need to press "space" to continue, or "q" to quit

LS (FOLDER)

Lists files and directory, in the current or specified folder

Current directory:	ls
Specific:	ls /home/pi/test

NB: You can use options with ls to get a more detailed view of files and folder, ex: ls -latr /home/pi

CP <SOURCE> <TARGET>

Copies a file or directory to another location (copy/paste)

Copy a file	cp test.txt /home/pi
Recursive copy:	cp -r /home/pi/test /home/user/

NB: Use the recursive option to copy a folder and all its files and folders

CAT <FILENAME>

Displays the content of the file, without pagination

Display on file:	cat test.txt
Use pattern:	cat *.txt

NB: A pattern allows you to display all files content for similar files

TAIL <FILENAME>

Displays the end of the file

Basic usage:	tail test.txt
Lines count:	tail -n20 test.txt
Real-time display:	tail -f test.txt

NB: The -n option allows you to ask for a specific number of lines to display
The -f option refresh the display each time the file is modified (perfect for log files monitoring)

FILES MANAGEMENT (2)

HEAD <FILENAME>

Similar to tail but to display the beginning of the file

Display 10 lines:	head test.xt
With lines count:	head -n20 test.txt

TOUCH <FILENAME>

Create a new empty file

Current directory:	touch test.txt
Specific:	touch /home/pi/test.txt

NB: Most of the time, nano is a better choice to create a file, as you can edit it directly

GREP

Grep is a powerful (and complex) tool to search string in a text or file

Find string in a file	grep "dhcp" /var/log/syslog
Filter a command output	ls -latr | grep ".php"
With a script:	/home/pi/script.sh | grep error

NB: The | option (pipe), allows you to run a command on another one output
You need to use quotes for complex search with space or special characters

There are also advanced usages possible:

Regular expressions:	grep "dhcp\\\|dns" /var/log/syslog
Command options:	grep -A2 -B4 'Fatal error' /var/log/syslog
Inverted search:	grep -v 'Notice' /var/log/syslogi

The | in the regular expressions allows you to use OR (one or more condition)
The -A option also catch X lines "after" the matched condition, -B is for "before"
Finally, the -v option is to filter lines that don't match the condition

NANO <FILENAME>

Opens and edit the specified file. Nano is a powerful text editor in a terminal

Basic usage:	nano /home/pi/test.txt

NB: Nano will create the file if it doesn't exist

RM <FILENAME>

Removes a file or directory

Remove file:	rm test.txt
Remove directory:	rm -rf /home/pi/test

NB: You need to use -rf options to remove a directory even if not empty (recursive + force)

TAR

Tar is the linux way to manage compressed files

Create a new archive:	tar -cvfz archive.tar.gz /home/pi/test
Extract files	tar -xvfz archive.tar.gz

Options:
-c is to Compress, -x to eXtract
-v: verbose mode, -z: use gZip to compress, -f specify the file name
Use "man tar" for more information

ZIP / UNZIP

Zip is similar to tar, but mainly used on Windows systems

Create a new archive:	zip -r archive.zip /home/pi/test
Extract files:	unzip archive.zip

NB: The -r option is to compress all the folder content
You can use the -d option to extract files in a specific folder
Use "man zip" or "man unzip" for all available options

FILES MANAGEMENT (3)

- ## PWD

An easy command to display you current directory

Example:	pwd

- ## TREE

Another tool to get details on your current location, in a tree format

Current directory:	tree
Specific folder:	tree /home/pi/

NB: There are a few options to filter the output, by selecting only directory, managing symbolic links or setting a max depth level

- ## FIND

Find allows you to search files on your Raspberry Pi, there is a lot of options

Find a file name:	find /home/pi -iname test.txt
Filter extensions:	find /home/pi -iname *.php
Find only directories:	find / -type d -iname test

NB: -iname stands for "insensitive case", you can use -name if you prefer
You can use "-type f" to find only files

More advanced options:

File size:	find / -size +10M
Recently modified files:	find /home -mtime -2

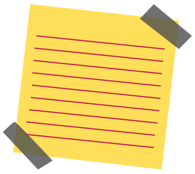
Run command on results:

```
find /var/log -iname *.log.gz -exec rm {} \;
```

The first command display all files over 10M on the disk
The -mtime -2 checks files modified in the last two days
The {} parameter in the last command will be replaced by the file name
Check the "man find" for more information

NETWORK COMMANDS

Here are the main commands to know to manage and use the network on your Raspberry Pi



Reminder:

Recent Raspberry Pi models comes with two interfaces : Ethernet and Wifi
Ethernet is called eth0 and the Wifi one is wlan0

IFCONFIG

Displays your current network configuration (IP Address, Mac Address, ...)

Usage: `ifconfig`

NB: You can add an interface name to display only this one, for example: "ifconfig wlan0"

IWCONFIG

Shows information about the wireless network configuration (SSID, speed, ...)

Usage: `iwconfig`

NB: You can also display a specific interface with iwconfig wlan0

IFUP / IFDOWN

Allows you to enable or disable one specific interface

Enable interface: `sudo ifup eth0`
Disable interface: `sudo ifdown eth0`

NB: It can help to disable the wireless interface while connected by cable

PING <HOST>

Checks if the host is alive

Basic usage: `ping 192.168.1.1`

NB: Read the "man ping" to see all available options

HOSTNAME

Displays or set the Raspberry Pi hostname

Display hostname: `hostname`
Set a new hostname: `sudo hostname RaspberryZero`

WGET <URL>

Download a file with the terminal

Basic usage: `wget http://192.168.1.1/test.txt`
Change file name:
`wget http://192.168.1.1/test.txt -O  target.txt`

SSH <USER>@<IP>

Connects to another Linux system with SSH

Example: `ssh pi@192.168.1.1`

SCP

Copies a file over the network by using SSH

Syntax: `scp <file> <user>@<ip>:<path>`
Example: `scp test.txt pi@192.168.1.1:/home/pi/`

RSYNC

Similar to SCP with more options like delta comparison and some other optimizations

Syntax: `rsync <file> <user>@<ip>:<path>`
Example: `rsync test.txt pi@192.168.1.1:/home/pi/`

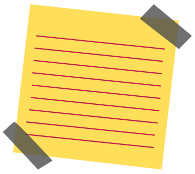
Local copy: `rsync /home/pi/* /media/usb/`
Remote recursive copy: `rsync -auzr /home/pi/Documents/* pi@192.168.1.1:/home/pi/Documents/`

NB: Use "man rsync" to get all possible options

RaspberryTips https://raspberrytips.com

PACKAGES MANAGEMENT

Once you have the network working, you'll probably update your system and install needed package
On this page, you have all the required commands to do this from a terminal



Vocabulary:

On Linux, each software is a **package**, as well as each **dependency**
You are downloading new packages from **repositories** (servers hosting packages)
You need to use a tool called **apt** to search, install and updates packages on Debian/RPI OS
All these commands need root privilege, you have to use sudo before each one

APT UPDATE

Downloads the last packages list from your repositories

Usage:

```
sudo apt update
```

NB: To add a new repository, you can edit the apt configuration in /etc/apt/sources.list, or follow the instructions from the software editor

RPI-UPDATE

Updates everything on your Raspberry Pi, use with precaution

Usage:

```
sudo rpi-update
```

APT REMOVE <PACKAGE>

Uninstall a package from your system

Usage:

```
sudo apt remove vim
```

NB: I give you the command to list currently installed packages in the next line

MANUAL INSTALLATION

Sometimes, you need to install packages manually, if the editor doesn't provide a repository

Download the file with wget:

```
wget https://www.realvnc.com/download/file/viewer.files/VNC-Viewer-6.19.325-Linux-ARM.deb
```

Manual installation:

```
sudo dpkg -i VNC-Viewer-6.19.325-Linux-ARM.deb
```

NB: You can use dpkg -r to remove a package manually, or dpkg-reconfigure to redo the configuration after installation

APT UPGRADE

Downloads and installs the latest version of each package available in the repository

Usage:

```
sudo apt upgrade
```

NB: You need to run apt update before doing this, to get the latest versions
The -y option allows you to automatically accept the installation

APT INSTALL <PACKAGE>

Installs the specified package on your system

Usage:

```
sudo apt install phpmyadmin
```

NB: Use the following search command to know the exact name of a package

APT SEARCH

Very useful to find the exact package name before installing it

Usage:

```
apt search openjdk
```

With grep:

```
apt search openjdk | grep jre
```

NB: You don't need sudo for this one

LIST INSTALLED PACKAGES

Dpkg can also be useful to list currently installed packages

Syntax:

```
dpkg -l
```

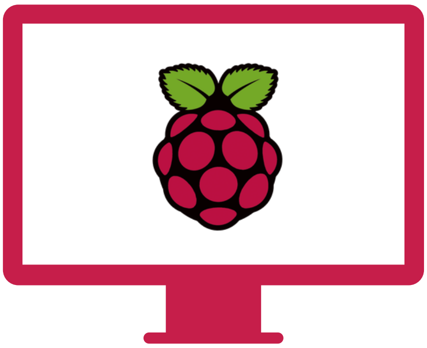
With grep:

```
dpkg -l | grep php
```

NB: Read the "man dpkg" output to get all possible options from this command

SYSTEM MANAGEMENT

Now that you have all packages installed, you may need to learn more advanced commands on how to manage your Raspberry Pi operating system



REBOOT

This command will restart your Raspberry Pi immediately

Usage:	`sudo reboot`

SERVICE

Each daemon has an associated service, you can start or stop it when you want

Start:	`sudo service apache2 start`
Stop:	`sudo service apache2 stop`
Restart:	`sudo service apache2 start`
Reload config:	`sudo service apache2 reload`

NB: Use "service <service>" to list all available options, for example "service apache2"
The tab key will help you to find the service name

PROCESS LIST

Displays all running processes

Basic usage:	`ps aux`
Only by a specific user:	`ps -u pi`

NB: I give you the command to list currently installed packages in the next line

HTOP

A great alternative to top, to display system load and process in an intuitive interface

Usage:	`htop`

NB: htop is not installed by default, install it with "apt install htop"

SHUTDOWN

Stops the Raspberry Pi, now or at a specific time

Stop now:	`sudo shutdown -h now`
At a specific time:	`sudo shutdown -h 20:00`

START SERVICE ON BOOT

Most of the time, services automatically start on boot, but if needed you can do this manually

Start on boot:	`sudo update-rc.d ssh enable`
Don't start on boot:	`sudo update-rc.d -f ssh remove`

NB: To start a script on boot, add it to the /etc/rc.local file

KILL / KILLALL

Immediately stop a specific process or all processes from the same command

Kill:	`kill 12345`
Killall:	`killall php`

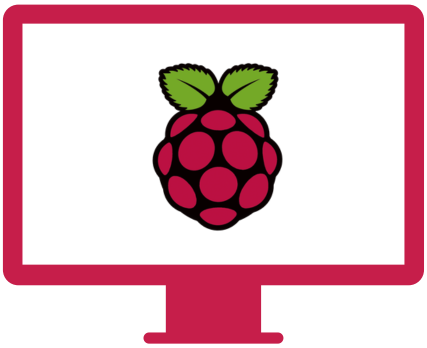
NB: Use the ps command to find the process ID to kill

DF

Displays your partition list, a good way to check the remaining disk space

Basic usage:	`df`
More readable:	`df -h`
Specific partition:	`df -h /media/usb`

SYSTEM MANAGEMENT (2)



DU

Displays the disk space usage in the current or specified folder

Basic usage:	du
Specific folder:	du /home/pi
Summarize:	du --summarize /home/pi
20 biggest files:	du -ak | sort -nr | head -20

NB: There are a lot more options, check the "man du" to find more help about this one

DATE

As the name says, display the current date and time

Full output:	date
Specific format:	date +%m-%d-%Y

NB: The "man date" command gives you the list of all availables options and format

CHOWN

Changes file owner and group

Change file owner:	sudo chown pi /usr/local/bin/script.sh
Change file owner & group:	sudo chown pi:www-data /var/www/html/mysite

CPU TEMPERATURE

It's not an easy command to remember, but very useful while overclocking or running consuming apps

Usage:	vcgencmd measure_temp

NB: vcgencmd is hidden in the libraspberrypi-bin package, you may need to install it manually on Raspberry Pi OS lite: "sudo apt install libraspberrypi-bin"

MOUNT

Mount a new partition (usb key for example)

Mount disk:	sudo mount /dev/sda1 /mnt/usb
Unmount:	sudo umount /mnt/usb

NB: It's a complex command for beginner, but this post will give you all the needed informations https://raspberrytips.com/mount-usb-drive-raspberry-pi/

UPTIME

Displays the current uptime of the Raspberry Pi (how many time on)

Basic usage:	uptime
Last boot date:	uptime -s

CHMOD

Changes file or folder permissions

Digits permissions:	chmod 644 script.sh
Letters permissions:	chmod +x script.sh

NB: Chmod is a complex command for beginner, you can check this tool to know how to read and set permissions correctly: https://chmod-calculator.com/

MAN <COMMAND>

I already give it a lot of times in this document, but man allows finding help on a command

Example:	man find

NB: Press space to go to the next page, and "q" to leave

RASPBERRY PI OS COMMANDS

As a Debian-like operating system, RPI OS use most of the same commands
But you'll find here the specific RPI OS commands

Note:

There are a few commands that only works on Raspberry Pi OS
They are not essentials to use a Raspberry Pi (except the first one probably)
But on most websites you'll not find them as they are not present on other Linux distributions

RASPI-CONFIG

This is the main tool to configure your Raspberry Pi from a terminal

Usage: `sudo raspi-config`

NB: Raspi-config allows you a lot of changes in your Raspberry Pi configuration, like password, network options, boot options, localisation options, interfacing options (ssh), overclocking and other advanced options

LIBCAMERA-VID

It's the same thing but to capture video from your camera

Basic usage: `libcamera-vid -o video.h264 -t 10000`

NB: -t option is for the time you want to capture the video
You'll find all needed information on how to use your camera on this post :
https://raspberrytips.com/camera-raspberry-pi/

RPI-UPDATE

We already saw this command in the system updates section, it'll update everything on your system

Usage: `sudo rpi-update`

LIBCAMERA-STILL

This command allows you to take a picture from the Raspberry Pi camera

Basic usag `libcamera-still -o image.jpg`

NB: You'll find all needed informations on how to use your camera on this post :
https://raspberrytips.com/camera-raspberry-pi/

RASPI-GPIO

Set or get values from your GPIO pins in a terminal

Get value: `sudo raspi-gpio get`
Set value: `sudo raspi-gpio set 20 a5`

NB: It can be a good start to check that your circuit is working, but the best way is to use Python scripts, more info here:
https://raspberrytips.com/raspberry-pi-gpio-pins/

MISCELLANEOUS COMMANDS

In this part, I wanted to give you all others useful commands that doesn't fit into the others

HISTORY

Linux stores any command you type in an archive file, you can read it with "history"

All commands:	history
Last 20:	history | tail -n 20
Clear all history:	history -c
Clear one line:	history -d 123

|

I already show you the pipe in a lot of examples, it allows you to combine multiple commands to find exactly what you want

Syntax:	<command1> | <command2>
Grep example:	cat test.txt | grep error
Double:	du -ak | sort -nr | head -20

!

Run a specific command from the history

Syntax:	!<history_id>
Example:	!123

NB: The history ID changes on each new command you type (including !), make sur to use only once or check the ID again

>

Create a file to store the command output

Syntax:	<command> > <filename>
Example:	cat test.txt | grep error > error.log

NB: The last command put all lines containing "error" in the test.txt file
This command doesn't output anything

CRONTAB

Allows you to schedule tasks on your Raspberry Pi

List current tasks:	crontab -l
Edit tasks:	crontab -e

NB: The crontab syntax is a tough to understand for beginners, use this tool to check your line is correct:
https://crontab.guru/

SCREEN

Run a virtual terminal, to let a session running in background

Start a screen:	screen -S <name>
Exit a screen:	CTRL+A CTRL+D
Resume a screen:	screen -r <name>
Stop a screen:	CTRL+D

!!

Similar to ! but to run the last command again

Usage:	!!

NB: Can be useful to run the same complex commands several times

>>

Add the command output at the end of a file

Usage:	cat test.txt | grep error >> error.log

NB: It's the same usage than >
But in this case, it'll add the lines to the error.log file, and keep the beginning as it was

WARRIORS COMMANDS

And finally, now that you're an expert with a terminal, let's see
some tricky commands to push your limits :)
They can be hard to use, with a lot of options, or hard to analyze

AWK

Awk is close to a programming language
Allows you to search string and transform them to display differently

Syntax:	awk [-F] [-v var=value] 'program' file
Basic example:	awk -F":" '{print $1}' /etc/passwd

NB: The last command displays only the first column
I can't explain to you the awk usage in detail in a few lines
Check this guide to learn more about this:
https://do.co/2VC8mnm

CUT

Another way to transform text in a command line, probably easier to understand

Syntax:	cut <option> <file>
Example:	cut -d : -f 1 /etc/passwd

NB: -d set the delimiter to use, and -f the field to keep
Use "man cut" to learn more about other options

LSOF

Stands for "LiSt Open Files", displays all currently opened files on your Raspberry Pi

Usage:	lsof

NB: Use grep with a pipe to find the file you're looking for

NETSTAT

Monitors your network activity

Listening ports:	netstat -l
Add the process ID:	netstat -lp
Same thing in real-time:	netstat -lpc

NB: There are many other options for netstat, you can check the "man netstat" page to learn more

SED

Similar to awk, but for regular expressions only

Syntax:	sed <option> <script> <file>
Basic example:	sed '/^#/d' /etc/apache2/apache2.conf

NB: The last command remove comments from the configuration
As for awk, you'll need serious tutorials and experience to master this one.

WC

WC stands for "Words Count" and also gets lines count, characters count and file size

Syntax:	wc <options> <file>
Lines count:	wc -l /var/log/syslog

NB: -l is for lines, -w for words and -m for characters
You can also use it after a pipe (to count lines from a grep command for example)

WATCH

Monitors a command output, by running it at each specified interval

Basic usage:	watch date
Specific time:	watch -n10 date

NB: Default refresh time is 2s

DMESG

Shows a log file of every events happening in the last boot sequence

Usage:	dmesg

NB: Most of them are normal
You can use grep to look for errors or a specific thing

Thanks for Reading !

See you soon on RaspberryTips
Patrick

RaspberryTips https://raspberrytips.com