Linux commands

Pedro José Albarracín Vargas

1. Introduction

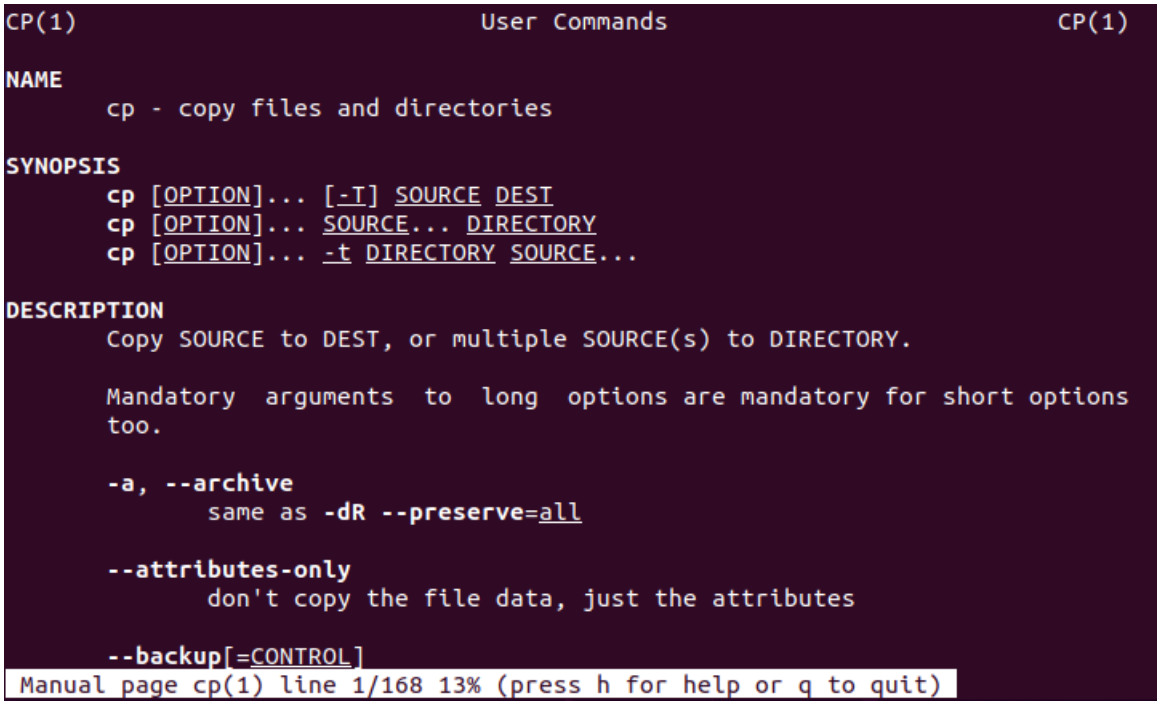
Linux is a free code operating system that will allow us to expand the scanning windows with fewer retrictions that can be handled in another OS, such as Windows or macOS. Command interpreter `shell` exposes an interface for the user to work with the operating system Linux. There are many different types of shells. Four our need focuses on Unix shells, which are commonly found on Linux and macOS computers. It's the `Bash` (also BASH or bash), the **Bourne-again shell**, is a free Unix shell under GPL. Like all script-type command-line interpreters, Bash performs four fundamental operations:

1. It provides a list of commands to operate on the computer (launching programs, copying files, etc.).
2. It allows to group these commands into a single file called a script.
3. It checks the command line when it is run or during a possible verification procedure and returns an error message in case of a syntax error.
4. In case of validation, each command line is interpreted, that is, translated into a language understandable by the operating system, which then executes it.

Next, we will see the main commands that are used in this shell for file manipulation.

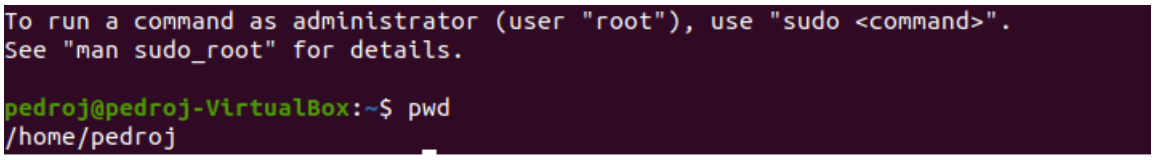
man command

The `man` command is manual gives complete information of a command. For instance, `man cp`:



pwd Command

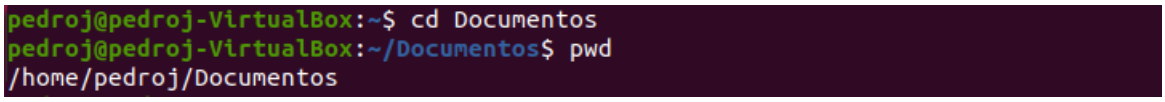
`pwd` means "printed word directory". If you want to know the current location of your directory, use this command:



It will print the path of the current folder.

cd Command

`cd` means "change directory". Once you have a folder, `cd` allows to move to that folder:



using the command `pwd`, verify that you changed the directory. You can use the special path `cd ..` to indicate the main folder.

ls Command

Inside a folder you can view all the files that the folder contains as a list using the `ls` command:

```
pedroj@pedroj-VirtualBox:~$ ls Documentos
cocodrilo-655x368.jpg  Nombres              stockpkg_mm9598_32.jpg
images.jpeg            resolucion_8470.pdf
```

If you want to see the list of a folder without moving from your directory, type the `ls` command followed by the folder name. For instance, `ls Documentos`, where "Documentos" is the folder name.

```
pedroj@pedroj-VirtualBox:~$ ls -al Documentos
total 424
drwxr-xr-x  3 pedroj pedroj   4096 mar 28 09:53 .
drwxr-xr-x 15 pedroj pedroj   4096 mar 28 09:45 ..
-rw-rw-r--  1 pedroj pedroj  60283 mar 28 09:49 cocodrilo-655x368.jpg
-rw-rw-r--  1 pedroj pedroj   7013 mar 28 09:49 images.jpeg
drwxrwxr-x  2 pedroj pedroj   4096 mar 28 09:53 Nombres
-rw-rw-r--  1 pedroj pedroj 188059 mar 28 09:51 resolucion_8470.pdf
-rw-rw-r--  1 pedroj pedroj 160548 mar 28 09:48 stockpkg_mm9598_32.jpg
```

A variation of this command is `ls -al`. This combination of commands gives us much more information than the previous command:

- Permissions of the files.
- Link numbers of the files.
- Owns the file.
- File size in bytes.
- Date and time of last file modification.

Commands for files and folders

Create

To create a text file or empty file, we uses `touch` command. Also, it can to create various files in only command. For instance, `touch Ring Oso Mesa`.
To create a folder, we uses `mkdir -r` command. we can create folders in string using the command `mkdir -p`:

```
pedroj@pedroj-VirtualBox:~/Documentos$ mkdir -p carpeta1/carpeta2/carpeta3
pedroj@pedroj-VirtualBox:~/Documentos$ cd carpeta1/carpeta2/carpeta3/
pedroj@pedroj-VirtualBox:~/Documentos/carpeta1/carpeta2/carpeta3$ pwd
/home/pedroj/Documentos/carpeta1/carpeta2/carpeta3
pedroj@pedroj-VirtualBox:~/Documentos/carpeta1/carpeta2/carpeta3$
```

We create folders *carpeta1*, *carpeta2* and *carpeta3*. Then we see that the folders are organized hierarchically (check by viewing the path). If you want to copy a file, use `cp` command:

```
pedroj@pedroj-VirtualBox:~/Documentos$ cp images.jpeg carpeta1/
pedroj@pedroj-VirtualBox:~/Documentos$ ls carpeta1/
carpeta2  images.jpeg
```

To copy folders, need to add the option `-r`: `cp -r carpeta1 Nombres` (Folder *carpeta1* is pasted in *Nombres*).

Remove

To remove files, type the `rm` command. To remove folders, it needs `rmdir` command.

Display commands

`cat` command shows small size files as text files. `less` command displays the content stored within a file, in a pleasant and interactive user interface:

```
pedroj@pedroj-VirtualBox:~/Documentos$ cat arch.txt
Do Re Mi Fa Sol La Si
pedroj@pedroj-VirtualBox:~/Documentos$ less arch.txt
```

```
Do Re Mi Fa Sol La Si
arch.txt (END)
```

We uses the key `q` for exit. To view the specific file type, we use the `file` command:

```
pedroj@pedroj-VirtualBox:~/Documentos$ file arch.txt
arch.txt: ASCII text
pedroj@pedroj-VirtualBox:~/Documentos$ file images.jpeg
images.jpeg: JPEG image data, JFIF standard 1.01, aspect ratio, density 1x1, seg
ment length 16, baseline, precision 8, 225x225, components 3
```

The `find` command helps to search for a specific file or files with some particular feature:

```
pedroj@pedroj-VirtualBox:~/Documentos$ find  *.jpg
cocodrilo-655x368.jpg
stockpkg_mm9598_32.jpg
```

echo Command

the `echo` command, prints the argument you missed in the output.

```
pedroj@pedroj-VirtualBox:~/Documentos$ echo hello
hello
```

Will print hello on the terminal.

type Command

A command can be executable, a program built into shell, a shell function or an alias. The `type` command can help solve this, in case we want to know.

```
pedroj@pedroj-VirtualBox:~/Documentos$ type pwd
pwd es una orden interna del intérprete de ordenes
pedroj@pedroj-VirtualBox:~/Documentos$ type ls
ls es un alias de `ls --color=auto'
pedroj@pedroj-VirtualBox:~/Documentos$ type cat
cat está asociado (/usr/bin/cat)
pedroj@pedroj-VirtualBox:~/Documentos$ type touch
touch está asociado (/usr/bin/touch)
```

whyami command

The `whyami` command prints the username that is connected to the terminal session.

clear command

To delete all previous commands that were executed on the current terminal, use `clear` command.

history command

The bash can show all the history of executed commands using `history` command:

```
pedroj@pedroj-VirtualBox:~/Documentos$ history
    1  echo hola
    2  pwd
    3  ls
    4  pwd
    5  clear
    6  ls
    7  clear
    8  cd Documentos
    9  pwd
   10  ls
   11  cd ..
   12  pwd
   13  ls Documentos/
   14  pwd
   15  clear
   16  ls Documentos
   17  ls -al Documentos
   18  clear
   19  pwd
   20  cd Documentos/
```

How to compress a file

You can compress Linux files with the open-source compression tool with Gzip or Zip, recognized by most operating systems. To compress with zip, you must do the following:
`zip -r total.zip carpeta1/`

```
pedroj@pedroj-VirtualBox:~/Documentos$ zip -r total.zip carpeta1
  adding: carpeta1/ (stored 0%)
  adding: carpeta1/carpeta2/ (stored 0%)
  adding: carpeta1/carpeta2/carpeta3/ (stored 0%)
  adding: carpeta1/images.jpeg (deflated 0%)
```

For instance, *total* represents new file that you have created and *carpeta1* represents the file that is inside the new file.

```
pedroj@pedroj-VirtualBox:~/Documentos$ unzip total.zip
Archive:  total.zip
   creating: carpeta1/
   creating: carpeta1/carpeta2/
   creating: carpeta1/carpeta2/carpeta3/
  inflating: carpeta1/images.jpeg
```

To extract the contents of the folder, consider `unzip` command as above, and then it obtain *carpeta1* file on the directory. To compress files `.tar`, it follows the order:
`tar -zcvf total.gz carpeta1/`
If you prefer to revert the command, use the following:
`tar -zxvf total.gz/`