

# GRASSHOPPER

## GETTING STARTED

The screenshot shows a Windows desktop with a browser window open to [linuxjourney.com/lesson/linux-history](https://linuxjourney.com/lesson/linux-history). The page title is "Getting Started". On the left, a sidebar menu lists "1. History" (selected), "2. Choosing a Linux Distribution", and other options like "Debian", "Ubuntu", etc. The main content area is titled "1. History". It contains text about the history of Linux, mentioning Ken Thompson, Dennis Ritchie, and Linus Torvalds. It also discusses the creation of the GNU kernel and the transition from MINIX to Linux. A sidebar on the right is titled "Ceremonias" and lists links to "GNU", "Ken Thompson", "Richard Stallman", and "Linus Torvalds". At the bottom right of the content area, there's a "Revisar respuesta" (Review answer) button.

The screenshot shows a Windows desktop with a browser window open to [linuxjourney.com/lesson/choosing-a-linux-distribution](https://linuxjourney.com/lesson/choosing-a-linux-distribution). The page title is "Getting Started". The sidebar menu now shows "2. Choosing a Linux Distribution" as selected. The main content area is titled "2. Choosing a Linux Distribution". It discusses the components of a Linux system: hardware, kernel, and user space. It also mentions the choice of distribution based on popularity. A sidebar on the right is titled "Ceremonias" and states "No hay ejercicios para esta lección". At the bottom right of the content area, there's a "Revisar respuesta" (Review answer) button.

**Getting Started**

- 1. History
- 2. Choosing a Linux Distribution
- 3. Debian**
- 4. Red Hat Enterprise Linux
- 5. Ubuntu
- 6. Fedora
- 7. Linux Mint
- 8. Gentoo
- 9. Arch Linux
- 10. openSUSE

## 3. Debian

### Introducción

Debian es un sistema operativo compuesto completamente de software gratuito y de código libre. Es conocido de forma amplia y ha estado en desarrollo por más de 20 años. Existen tres ramas que puedes utilizar, Stable, Testing y Unstable.

### Stable

Stable es la versión estable y en general una buena rama para utilizar. Testing y Unstable (Pruebas e Inestable) son las liberaciones continuas. Esto significa que cualquier cambio incremental en dichas ramas eventualmente será estable. Por ejemplo, si quisieras obtener la siguiente actualización de Windows 8 a Windows 10, necesitarías realizar una instalación completa de Windows 10. De cualquier forma al pertenecer a la liberación Testing obtendrás actualizaciones automáticas hasta que se convierta la siguiente liberación del sistema operativo sin tener que realizar una instalación completa.

### Administración de paquetes

Debian además utiliza las herramientas de administración de paquetes Debian. Cada distribución de Linux instala y administra los paquetes de forma diferente y utilizan diferentes herramientas para administrarlos. Nos adentraremos en estos temas en un curso posterior.

### Configurabilidad

Debian puede no tener las actualizaciones más recientes, pero es extremadamente estable. Si quieras un buen sistema operativo principal, este es el indicado.

### Usos

Debian en general es un gran sistema operativo para todas las plataformas.

### Ceremonias

Si estás interesado en tener Debian como tu sistema operativo, dirígete a la sección de instalación y dale una oportunidad: <https://www.debian.org/>

### Examen

¿Qué tipo de liberaciones utilizan las ramas Testing y Unstable?

Respuesta correcta!

continua

Revisar respuesta

Activar Windows  
Ve a Configuración para activar Windows.

Continuar

**Getting Started**

- 1. History
- 2. Choosing a Linux Distribution
- 3. Debian
- 4. Red Hat Enterprise Linux**
- 5. Ubuntu
- 6. Fedora
- 7. Linux Mint
- 8. Gentoo
- 9. Arch Linux
- 10. openSUSE

## 4. Red Hat Enterprise Linux

### Introducción

Red Hat Enterprise Linux, referido comúnmente como RHEL es desarrollado por Red Hat. RHEL tiene reglas estrictas para restringir la libre redistribución, de cualquier forma provee el código fuente de manera gratuita.

### Administración de paquetes

RHEL utiliza un administrador de paquetes diferente a Debian, el RPM, del cual aprenderemos también eventualmente.

### Configurabilidad

Los sistemas operativos basados en RHEL difieren ligeramente con los sistemas operativos basados en Debian, notablemente en la administración de paquetes. Si te decides por RHEL es probablemente mejor si sabes que le dedicarás trabajo.

### Usos

Como lo describe el nombre es usado principalmente de forma empresarial, así que si necesitas un sistema operativo sólido para un servidor, este sería el indicado.

### Ceremonias

Si estás interesado en tener RHEL como tu sistema operativo, dirígete a la sección de instalación y dale una oportunidad: <https://www.redhat.com/rhel/>

### Examen

¿Qué administrador de paquetes utiliza RHEL?

Respuesta correcta!

RPM

Revisar respuesta

Activar Windows  
Ve a Configuración para activar Windows.

Continuar

Getting Started

- 1. History
- 2. Choosing a Linux Distribution
- 3. Debian
- 4. Red Hat Enterprise Linux
- 5. Ubuntu**
- 6. Fedora
- 7. Linux Mint
- 8. Gentoo
- 9. Arch Linux
- 10. openSUSE

## 5. Ubuntu

### Introducción

Una de las distribuciones Linux más populares para las computadoras personales es Ubuntu. Ubuntu también incorpora su propio entorno de escritorio Unity por defecto.

### Administración de paquetes

Ubuntu es un sistema operativo basado en Debian y desarrollado por Canonical. Así que utiliza el sistema administrador de paquetes Debian.

### Configurabilidad

Ubuntu es una gran elección para un principiante que quiera adentrarse en Linux. Ubuntu ofrece gran facilidad de uso y una buena experiencia en interfaces de usuario lo cual ha propiciado su amplia adopción. Es ampliamente utilizado y soportado, además de estar a la par con otros sistemas operativos como OSX y Windows en términos de usabilidad.

### Usos

Genial para cualquier plataforma, de escritorio, laptop y servidor.

**Ceremonias**

Si estás interesado en tener Ubuntu como tu sistema operativo, dirígete a la sección de instalación y dale una oportunidad: <http://www.ubuntu.com/>

**Examen**

¿En cuál sistema operativo está basado Ubuntu?

✓ ¡Respuesta correcta!

debian Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

Buscar 17:07 6/6/2025

Getting Started

- 1. History
- 2. Choosing a Linux Distribution
- 3. Debian
- 4. Red Hat Enterprise Linux
- 5. Ubuntu
- 6. Fedora**
- 7. Linux Mint
- 8. Gentoo
- 9. Arch Linux
- 10. openSUSE

## 6. Fedora

### Introducción

Respaldoado por Red Hat, el proyecto Fedora es desarrollado por la comunidad y contiene software gratuito y de código abierto. Red Hat Enterprise Linux se desprende de Fedora, así que puedes pensar en Fedora como el sistema operativo padre de RHEL. Eventualmente RHEL obtendrá las actualizaciones de Fedora a través de pruebas y control de calidad exhaustivo. Piensa en que Fedora es el equivalente a Ubuntu solo que se basa en Red Hat en vez de Debian.

### Administración de paquetes

Utiliza el mismo administrador de paquetes que Red Hat.

### Configurabilidad

Si querés un sistema basado en Red Hat, ésta es la versión accesible.

### Usos

Fedora es genial si querés un sistema operativo basado en Red Hat sin el precio comercial. Recomendado para computadoras de escritorio y laptops.

**Ceremonias**

Si estás interesado en tener Fedora como tu sistema operativo, dirígete a la sección de instalación y dale una oportunidad: <https://getfedora.org/>

**Examen**

¿De cuál sistema operativo se desprendió RHEL?

✓ ¡Respuesta correcta!

Fedora Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

Buscar 17:08 6/6/2025

The screenshot shows a Windows desktop environment with a taskbar at the bottom containing icons for File Explorer, Mail, Photos, Videos, Task View, Edge, Google Chrome, and File Explorer. The main window is a web browser displaying the URL [linuxjourney.com/lesson/linux-mint](https://linuxjourney.com/lesson/linux-mint). The page has a dark blue header with the title "Getting Started". A sidebar on the left lists "1. History", "2. Choosing a Linux Distribution", "3. Debian", "4. Red Hat Enterprise Linux", "5. Ubuntu", "6. Fedora", "7. Linux Mint" (which is highlighted in green), "8. Gentoo", "9. Arch Linux", and "10. openSUSE". The main content area is titled "7. Linux Mint" and includes sections for "Introducción", "Administración de paquetes", and "Usos". A sidebar on the right is titled "Ceremonias" and contains a message about Linux Mint being based on Ubuntu. Another sidebar titled "Examen" asks "¿En cuál sistema operativo está basado Linux Mint?" and shows a correct answer of "Ubuntu" with a "Revisar respuesta" button. A "Continuar" button is also present.

This screenshot shows the same Windows desktop environment and taskbar as the previous one. The web browser displays the URL [linuxjourney.com/lesson/gentoo](https://linuxjourney.com/lesson/gentoo). The sidebar on the left shows the same list of Linux distributions, with "8. Gentoo" highlighted in green. The main content area is titled "8. Gentoo" and includes sections for "Introducción", "Administración de paquetes", and "Usos". A sidebar on the right is titled "Ceremonias" and contains a message about Gentoo's flexibility. Another sidebar titled "Examen" asks "¿Qué administrador de paquetes utiliza Gentoo?" and shows a correct answer of "Portage" with a "Revisar respuesta" button. A "Continuar" button is also present.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "9. Arch Linux" from the "Getting Started" section of the Linux Journey website. The page content includes sections on Arch's introduction, package management, configurability, and usage. A sidebar on the right contains a "Ceremonias" section with a note about ArchLinux.org and an "Examen" section asking about the package manager, with "Pacman" being the correct answer. The browser interface includes a navigation bar, a search bar, and a taskbar at the bottom.

Getting Started

1. History

2. Choosing a Linux Distribution

3. Debian

4. Red Hat Enterprise Linux

5. Ubuntu

6. Fedora

7. Linux Mint

8. Gentoo

9. Arch Linux

10. openSUSE

9. Arch Linux

**Introducción**

Arch es una distribución ligera y flexible desarrollada al 100% por la comunidad. Similar a Debian utiliza un modelo de liberaciones continuas que eventualmente se convierten en la liberación Stable. En realidad necesitas ensuciarte las manos para comprender el sistema y sus funciones, pero a cambio obtienes el total control de tu sistema.

**Administración de paquetes**

Utiliza su propio administrador de paquetes, Pacman, para instalar, actualizar y administrar los paquetes.

**Configurabilidad**

Si quieras un sistema operativo ligero y realmentequieres comprender Linux utiliza Arch. Realmente tienen una curva de aprendizaje alta, pero para los usuarios de Linux intensivos, esta es una gran alternativa.

**Usos**

Genial para computadoras de escritorio y laptops. Si tienes un dispositivo como Raspberry Pi y necesitas instalarle un sistema operativo ligero, definitivamente Arch es tu opción.

Ceremonias

Si estas interesado en tener Arch como tu sistema operativo, dirígete a la sección de instalación y dale una oportunidad: <https://www.archlinux.org/>

Examen

¿Qué administrador de paquetes utiliza Arch Linux?

✓ ¡Respuesta correcta!

Pacman

Revisar respuesta

Activar Windows

Continuar

Buscar

17:10

ESP

6/6/2025

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "10. openSUSE" from the "Empezando" section of the Linux Journey website. The page content includes sections on openSUSE's general description, package management, configurability, and usage. A sidebar on the right contains a "Ceremonias" section with a note about openSUSE.org and an "Examen" section asking about the YAST tool, with "YaST" being the correct answer. The browser interface includes a navigation bar, a search bar, and a taskbar at the bottom.

Empezando

1. Historia

2. Elección de una distribución de Linux

3. Debian

4. Red Hat Enterprise Linux

5. Ubuntu

6. Fedora

7. Linux Mint

8. Gentoo

9. Arch Linux

10. openSUSE

10. openSUSE

**Descripción general:**

openSUSE Linux fue creado por el Proyecto openSUSE. Una comunidad que promueve el uso de Linux en todas partes, colaborando de forma abierta, transparente y amigable como parte de la comunidad mundial de software libre y de código abierto. openSUSE es la segunda distribución de Linux más antigua que aún sigue en funcionamiento y comparte el sistema base con los galardonados productos SUSE Linux Enterprise de SUSE.

**Gestión de paquetes**

Utiliza el administrador de paquetes RPM.

**Configurabilidad:**

openSUSE es una excelente opción para nuevos usuarios de Linux. Ofrece una aplicación gráfica de instalación/administración ([YaST](#)) fácil de usar y un sistema base ordenado y fácil de usar. openSUSE incluye todo lo necesario para disfrutar de Internet sin preocupaciones de virus ni spyware y dar rienda suelta a tu creatividad, ya sea con tus fotos, videos, música o código.

**Usos**

openSUSE Leap es totalmente capaz de utilizarse en una PC de escritorio y una portátil.

Ceremonias

Si está interesado en tener openSUSE como su sistema operativo, diríjase a la página de descarga y pruébelo: [software.opensuse.org](http://software.opensuse.org)

Examen

¿Cuál es el nombre de la herramienta de administración/installación de openSUSE?

✓ ¡Respuesta correcta!

YaST

Revisar respuesta

Activar Windows

Devolver

Buscar

17:11

ESP

6/6/2025

# COMMAND LINE

**1. La concha**

El mundo es tuyo, o mejor dicho, tu concha es tu ostra. ¿Qué es la concha? Es básicamente un programa que recibe tus comandos del teclado y los envía al sistema operativo para que los execute. Si alguna vez has usado una interfaz gráfica de usuario (GUI), probablemente hayas visto programas como "Terminal" o "Consola". Estos son simplemente programas que inician una shell. A lo largo de este curso, aprenderemos sobre las maravillas de la shell.

En este curso, usaremos el programa de shell bash (Bourne Again Shell). Casi todas las distribuciones de Linux lo usan por defecto. Existen otros shells disponibles, como ksh, zsh y tsch, pero no los analizaremos en detalle.

¡Comencemos! Dependiendo de la distribución, el prompt de la shell podría cambiar, pero en general debería seguir el siguiente formato:

```
nombre de usuario@nombre de host:directorio_actual
pete@icebox:/home/pete $
```

¿Has notado el símbolo \$ al final del prompt? Cada shell tiene diferentes prompts. En nuestro caso, el símbolo \$ es para un usuario normal que usa Bash, Bourne o Korn. No se agrega el símbolo al escribir el comando, simplemente se sabe que está ahí.

Comencemos con un comando simple: echo. El comando echo simplemente muestra los argumentos de texto en la pantalla.

```
$ echo Hola mundo
```

**2. pwd (Print Working Directory)**

Everything in Linux is a file, as you journey deeper into Linux you'll understand this, but for now just keep that in mind. Every file is organized in a hierarchical directory tree. The first directory in the filesystem is aptly named the root directory. The root directory has many folders and files which you can store more folders and files, etc. Here is an example of what the directory tree looks like:

```
/  
|-- bin  
|   |-- file1  
|   |-- file2  
|-- etc  
|   |-- file3  
|   '-- directory1  
|       |-- file4  
|       '-- file5  
|-- home  
|-- var
```

The location of these files and directories are referred to as paths. If you had a folder named home with a folder inside of it named pete and another folder in that folder called Movies, that path would look like this: /home/pete/Movies, pretty simple huh?

Navigation of the filesystem, much like real life is helpful if you know where you are and where you are going. To see where you are, you can use the pwd command, this command means "print working directory" and it just shows you

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/change-directory-cd-command". The left sidebar lists 14 commands: 1. The Shell, 2. pwd (Print Working Directory), 3. cd (Change Directory) (which is selected and highlighted in green), 4. ls (List Directories), 5. touch, 6. file, 7. cat, 8. less, 9. history, 10. cp (Copy), 11. mv (Move), 12. mkdir (Make Directory), 13. rm (Remove), and 14. find.

**3. cd (Change Directory)**

Now that you know where you are, let's see if we can move around the filesystem a bit. Remember we'll need to navigate our way using paths. There are two different ways to specify a path, with absolute and relative paths.

- Absolute path: This is the path from the root directory. The root is the head honcha. The root directory is commonly shown as a slash. Every time your path starts with / it means you are starting from the root directory. For example, /home/pete/Desktop.
- Relative path: This is the path from where you are currently in filesystem. If I was in location /home/pete/Documents and wanted to get to a directory inside Documents called taxes, I don't have to specify the whole path from root like /home/pete/Documents/taxes, I can just go to taxes/ instead.

Now that you know how paths work, we just need something to help us change to the directory we want to. Luckily, we have cd or "change directory" to do that.

```
$ cd /home/pete/Pictures
```

So now I've changed my directory location to /home/pete/Pictures.

Now from this directory I have a folder inside called Hawaii, I can navigate to that folder with:

```
$ cd Hawaii
```

Notice how I just used the name of the folder? It's because I was already in /home/pete/Pictures.

It can get pretty tiring navigating with absolute and relative paths all the time, luckily there are some shortcuts to help you out.

**Ceremonias**

1. Run the cd command without any flags, where does it take you?

**Examen**

If you are in /home/pete/Pictures and wanted to go to /home/pete, what's a good shortcut to use?

Respuesta correcta

cd .. Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

17:17 6/6/2025

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/list-directories-ls-command". The left sidebar lists 14 commands: 1. The Shell, 2. pwd (Print Working Directory), 3. cd (Change Directory), 4. ls (List Directories) (which is selected and highlighted in green), 5. touch, 6. file, 7. cat, 8. less, 9. history, 10. cp (Copy), 11. mv (Move), 12. mkdir (Make Directory), 13. rm (Remove), and 14. find.

**4. ls (List Directories)**

Now that we know how to move around the system, how do we figure out what is available to us? Right now it's like we are moving around in the dark. Well, we can use the wonderful ls command to list directory contents. The ls command will list directories and files in the current directory by default, however you can specify which path you want to list the directories of.

```
$ ls  
$ ls /home/pete
```

ls is a quite useful tool, it also shows you detailed information about the files and directories you are looking at.

Also note that not all files in a directory will be visible. Filenames that start with . are hidden, you can view them however with the ls command and pass the -a flag to it (-a for all).

```
$ ls -a
```

There is also one more useful ls flag, -l for long, this shows a detailed list of files in a long format. This will show you detailed information, starting from the left: file permissions, number of links, owner name, owner group, file size, timestamp of last modification, and file/directory name.

```
$ ls -l
```

```
pete@icebox:~$ ls -l  
total 80  
drwxr-x--- 7 pete penguingroup 4096 Nov 20 16:37 Desktop
```

**Ceremonias**

Run ls with different flags and see the output you receive.

- ls -R: recursively list directory contents
- ls -r: reverse order while sorting
- ls -t: sort by modification time, newest first

**Examen**

What command would you use to see hidden files?

Respuesta correcta

ls -a Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

17:18 6/6/2025

linuxjourney.com/lesson/touch-command

Línea de comandos

- 1. La concha
- 2. pwd (Directorio de trabajo de impresión)
- 3. cd (Cambiar directorio)
- 4. ls (Lista de directorios)
- 5. tocar**
- 6. archivo
- 7. gato
- 8. menos
- 9. historia
- 10. cp (Copiar)
- 11. mv (Movimiento)
- 12. mkdir (Crear directorio)
- 13. rm (Quitar)

**5. tocar**

Aprendamos a crear archivos. Una forma muy sencilla es usar el comando táctil. Este comando permite crear nuevos archivos vacíos.

```
$ touch mysuperduperfile
```

¡Y bum, nuevo archivo!

Touch también se usa para cambiar las marcas de tiempo de archivos y directorios existentes. Pruébalo ejecutando un comando ls -l en un archivo y anota la marca de tiempo. Luego, toca ese archivo para actualizarla.

Hay muchas otras formas de crear archivos que involucran otras cosas como redirección y editores de texto, pero llegaremos a eso en el curso de Manipulación de texto.

**Ceremonias**

1. Crear un nuevo archivo
2. Tenga en cuenta la marca de tiempo
3. Toque el archivo y verifique la marca de tiempo una vez más

**Examen**

¿Cómo se crea un archivo llamado myfile?

✓ ¡Respuesta correcta!

```
touch myfile
```

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

17:20 6/6/2025

linuxjourney.com/lesson/file-command

Command Line

- 1. The Shell
- 2. pwd (Print Working Directory)
- 3. cd (Change Directory)
- 4. ls (List Directories)
- 5. touch
- 6. file**
- 7. cat
- 8. less
- 9. history
- 10. cp (Copy)
- 11. mv (Move)
- 12. mkdir (Make Directory)
- 13. rm (Remove)
- 14. find

**6. file**

In the previous lesson we learned about touch, let's go back to that for a bit. Did you notice that the filename didn't conform to standard naming like you've probably seen with other operating systems like Windows? Normally you would expect a file called banana.jpeg and expect a JPEG picture file.

In Linux, filenames aren't required to represent the contents of the file. You can create a file called funny.gif that isn't actually a GIF.

To find out what kind of file a file is, you can use the file command. It will show you a description of the file's contents.

```
$ file banana.jpg
```

**Ceremonias**

Run the file command on a few different directories and files and note the output.

**Examen**

What command can you use to find the file type of a file?

✓ ¡Respuesta correcta!

```
file
```

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

17:21 6/6/2025

**Command Line**

1. The Shell
2. pwd (Print Working Directory)
3. cd (Change Directory)
4. ls (List Directories)
5. touch
6. file
- 7. cat**
8. less
9. history
10. cp (Copy)
11. mv (Move)
12. mkdir (Make Directory)
13. rm (Remove)
14. find

**7. cat**

We're almost done navigating files, but first let's learn how to read a file. A simple command to use is the cat command, short for concatenate, it not only displays file contents but it can combine multiple files and show you the output of them.

```
$ cat dogfile birdfile
```

It's not great for viewing large files and it's only meant for short content. There are many other tools that we use to view larger text files that we'll discuss in the next lesson.

**Ceremonias**

Run cat on different files and directories. Then try to cat multiple files.

**Examen**

What's a good way to see the contents of a file?

✓ ¡Respuesta correcta!

cat  Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

**Línea de comandos**

1. La concha
2. pwd (Directorio de trabajo de impresión)
3. cd (Cambiar directorio)
4. ls (Lista de directorios)
5. tocar
6. archivo
7. gato
- 8. menos**
9. historia
10. cp (Copiar)
11. mv (Movimiento)
12. mkdir (Crear directorio)
13. rm (Quitar)

**8. menos**

Si visualiza archivos de texto más grandes que una salida simple, menos es más. (De hecho, existe un comando llamado more que hace algo similar, lo cual resulta irónico). El texto se muestra paginado, lo que permite navegar por un archivo de texto página por página.

Continúe y revise el contenido de un archivo con "less". Una vez que esté en el comando "less", podrá usar otros comandos de teclado para navegar por el archivo.

```
$ menos /home/pete/Dокументos/texto1
```

Utilice el siguiente comando para navegar por less:

- q - Se utiliza para salir de less y volver a tu shell.
- Re Pág. Av Pág. Arriba y Abajo: navegue utilizando las teclas de flecha y las teclas de página.
- g - Se mueve al principio del archivo de texto.
- G - Se mueve al final del archivo de texto.
- /search - Puedes buscar texto específico dentro del documento. Precede las palabras que deseas buscar con /.
- h - Si necesitas un poco de ayuda sobre cómo usar menos mientras estás en menos, busca ayuda.

**Ceremonias**

Ejecute menos en un archivo, luego avance página y navegue por él. Intente buscar una palabra específica. Navegue rápidamente al principio o al final del archivo.

**Examen**

¿Cómo salir de un comando menor?

✓ ¡Respuesta correcta!

q  Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/history-command". The left sidebar lists numbered commands from 1 to 14. Command 9, "history", is highlighted with a green background. The main content area starts with a section titled "9. history". It explains that the history command lists previous commands and includes examples of using it and navigating through its output. A code input field contains "\$ history". Below this, another code input field contains "\$ clear", with a note that it looks better. The right panel contains two sections: "Ceremonias" (with instructions on navigating history) and "Examen" (a question about clearing the terminal). A green button indicates a correct answer ("¡Respuesta correcta!").

This screenshot shows the same browser window on a different page, "linuxjourney.com/lesson/copy-cp-command". The sidebar highlights command 10, "cp (Copiar)". The main content area discusses the cp command for copying files and directories. It covers wildcards (\*, ?, []), recursive copying (-r), and interactive mode (-i). Examples include "\$ cp \*.jpg /home/pete/Imágenes" and "\$ cp -r Pumpkin/ /home/pete/Dокументos". The right panel's "Examen" section asks about specifying the -r flag for recursive copying. A green button indicates a correct answer ("¡Respuesta correcta!").

linuxjourney.com/lesson/move-mv-command

**Command Line**

- 1. The Shell
- 2. pwd (Print Working Directory)
- 3. cd (Change Directory)
- 4. ls (List Directories)
- 5. touch
- 6. file
- 7. cat
- 8. less
- 9. history
- 10. cp (Copy)
- 11. mv (Move)**
- 12. mkdir (Make Directory)
- 13. rm (Remove)
- 14. find

**11. mv (Move)**

Used for moving files and also renaming them. Quite similar to the cp command in terms of flags and functionality.

You can rename files like this:

```
$ mv oldfile newfile
```

Or you can actually move a file to a different directory:

```
$ mv file2 /home/pete/Documents
```

And move more than one file:

```
$ mv file_1 file_2 /somedirectory
```

You can rename directories as well:

```
$ mv directory1 directory2
```

Like cp, if you mv a file or directory it will overwrite anything in the same directory. So you can use the -i flag to prompt you before overwriting anything.

```
$ mv -i directory1 directory2
```

Let's say you did want to mv a file to overwrite the previous one. You can also make a backup of that file and it will just rename the old version with a ~.

```
$ mv -b directory1 directory2
```

**Ceremonias**

Rename a file, then move that file to a different directory.

**Examen**

How do you rename a file called cat to dog?

Respuesta correcta!

mv cat dog

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

17:36 6/6/2025

linuxjourney.com/lesson/make-directory-mkdir-command

**Command Line**

- 1. The Shell
- 2. pwd (Print Working Directory)
- 3. cd (Change Directory)
- 4. ls (List Directories)
- 5. touch
- 6. file
- 7. cat
- 8. less
- 9. history
- 10. cp (Copy)
- 11. mv (Move)
- 12. mkdir (Make Directory)**
- 13. rm (Remove)
- 14. find

**12. mkdir (Make Directory)**

We're gonna need some directories to store all these files we've been working on. The mkdir command (Make Directory) is useful for that, it will create a directory if it doesn't already exist. You can even make multiple directories at the same time.

```
$ mkdir books paintings
```

You can also create subdirectories at the same time with the -p (parent flag).

```
$ mkdir -p books/hemingway/favorites
```

**Ceremonias**

Make a couple of directories and move some files into that directory.

**Examen**

What command is used to make a directory?

Respuesta correcta!

mkdir

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

17:37 6/6/2025

**Command Line**

- 1. The Shell
- 2. pwd (Print Working Directory)
- 3. cd (Change Directory)
- 4. ls (List Directories)
- 5. touch
- 6. file
- 7. cat
- 8. less
- 9. history
- 10. cp (Copy)
- 11. mv (Move)
- 12. mkdir (Make Directory)
- 13. rm (Remove)**
- 14. find

**13. rm (Remove)**

Now I think we have too many files, let's remove some files. To remove files you can use the rm command. The rm (remove) command is used to delete files and directories.

```
$ rm file1
```

Take caution when using rm, there is no magical trash can that you can fish out removed files. Once they are gone, they are gone for good, so be careful.

Fortunately there are some safety measures put into place, so the average joe can't just remove a bunch of important files. Write-protected files will prompt you for confirmation before deleting them. If a directory is write-protected it will also not be easily removed.

Now if you don't care about any of that, you can absolutely remove a bunch of files.

```
$ rm -f file1
```

-f or force option tells rm to remove all files, whether they are write protected or not, without prompting the user (as long as you have the appropriate permissions).

```
$ rm -i file
```

Adding the -i flag like many of the other commands, will give you a prompt on whether you want to actually remove the files or directories.

```
$ rm -r directory
```

**Ceremonias**

1. Create a file called -file (don't forget the dash!).
2. Remove that file.

**Examen**

How do you remove a file called myfile?

¡Respuesta correcta!

rm myfile Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

**Línea de comandos**

- 1. La concha
- 2. pwd (Directorio de trabajo de impresión)
- 3. cd (Cambiar directorio)
- 4. ls (Lista de directorios)
- 5. tocar
- 6. archivo
- 7. gato
- 8. menos
- 9. historia
- 10. cp (Copiar)
- 11. mv (Movimiento)
- 12. mkdir (Crear directorio)
- 13. rm (Quitar)
- 14. encontrar**

**14. encontrar**

Con todos estos archivos en el sistema, puede resultar un poco complicado encontrar uno específico. Para ello, existe un comando: "find".

```
$ find /home -name cachorros.jpg
```

Con find tendrás que especificar el directorio en el que buscarás, qué estás buscando, en este caso estamos tratando de encontrar un archivo con el nombre de puppies.jpg.

Puede especificar qué tipo de archivo está intentando encontrar.

```
$ find /home -type d -name MiCarpeta
```

Puedes ver que configuré el tipo de archivo que estoy tratando de encontrar como (d) para el directorio y todavía estoy buscando por el nombre de MyFolder.

Una cosa interesante a tener en cuenta es que la función de búsqueda no se detiene en el directorio que estás buscando, sino que también buscará dentro de cualquier subdirectorio que ese directorio pueda tener.

**Ceremonias**

1. Busque un archivo en el directorio raíz que contenga la palabra net.

**Examen**

¿Qué opción debo especificar para buscar si quiero buscar por nombre?

¡Respuesta correcta!

-name Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

**15. help**

Linux has some great built-in tools to help you how to use a command or check what flags are available for a command. One tool, help, is a built-in bash command that provides help for other bash commands (echo, logout, pwd, etc).

```
$ help echo
```

This will give you a description and the options you can use when you want to run echo. For other executable programs, it's convention to have an option called --help or something similar.

```
$ echo --help
```

Not all developers who ship out executables will conform to this standard, but it's probably your best shot to find some help on a program.

**Ceremonias**

Run help on the echo command, logout command and pwd command.

**Examen**

How do you get quick command line help for built-in bash commands?

✓ ¡Respuesta correcta!

help  Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

**16. man**

Gee I wish some of these programs had a manual so we can see some more information about them. Well luckily they do! Aptly named man pages, you can see the manuals for a command with the man command.

```
$ man ls
```

Man pages are manuals that are by default built into most Linux operating systems. They provide documentation about commands and other aspects of the system. Try it out on a few commands to get more information about them.

**Ceremonias**

Run the man command on the ls command.

**Examen**

How do you see the manuals for a command?

✓ ¡Respuesta correcta!

man  Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/whatis-command". The main content area displays a "Command Line" sidebar on the left containing numbered commands from 1 to 14. The main content area shows a section titled "17. whatis" with the following text:

Whew, we've learned quite a bit of commands so far, if you are ever feeling doubtful about what a command does, you can use the whatis command. The whatis command provides a brief description of command line programs.

```
$ whatis cat
```

The description gets sourced from the manual page of each command. If you ran whatis cat, you'd see there is a small blurb with a short description.

To the right of the main content area, there are two sections: "Ceremonias" and "Examen". The "Ceremonias" section contains the text "Run the whatis command on the less command." The "Examen" section asks, "What command can you use to see a small description of a command?" It shows a green feedback box with "¡Respuesta correcta!" and a text input field containing "whatis". A blue button labeled "Revisar respuesta" is visible, along with a note to "Activar Windows" and a "Continuar" button.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/alias-command". The main content area displays a "Línea de comandos" sidebar on the left containing numbered commands from 1 to 13. The main content area shows a section titled "18. alias" with the following text:

A veces, escribir comandos puede resultar muy repetitivo, o si necesitas escribir un comando largo muchas veces, es mejor tener un alias que puedas usar para ello. Para crear un alias para un comando, simplemente especifica un nombre de alias y asignale el nombre al comando.

```
$ alias foobar='ls -la'
```

Ahora, en lugar de escribir ls -la, puedes escribir foobar y se ejecutará ese comando, ¡una gran idea! Ten en cuenta que este comando no guardará tu alias después de reiniciar, así que tendrás que añadir un alias permanente en:

```
~/.bashrc
```

o archivos similares si deseas que persista después de reiniciar.

Puedes eliminar alias con el comando unalias:

```
$ unalias foobar
```

To the right of the main content area, there are two sections: "Ceremonias" and "Examen". The "Ceremonias" section contains the text "Crea un par de alias y luego eliminalos." The "Examen" section asks, "¿Qué comando se utiliza para crear un alias?" It shows a green feedback box with "¡Respuesta correcta!" and a text input field containing "alias". A blue button labeled "Revisar respuesta" is visible, along with a note to "Activar Windows" and a "Continuar" button.

The screenshot shows a Windows desktop environment. A browser window is open to [linuxjourney.com/lesson/exit-command](https://linuxjourney.com/lesson/exit-command). The page displays a lesson titled "19. exit". The sidebar on the left lists "Command Line" topics numbered 1 through 14. The main content area shows the following text and code snippets:

Well, you sure did a good job getting through the basics. We've only scratched the surface, now that you've learned to crawl, in the next set of courses, I'm gonna teach how to walk.

For now, you can pat yourself on the back and take a break. To exit from the shell, you can use the `exit` command

```
$ exit
```

Or the logout command:

```
$ logout
```

Or if you are working out of a terminal GUI, you can just close the terminal, see you in the next course!

**Ceremonias**

Exit out of the shell and see what happens. Make sure you don't need to do anymore work in that shell.

**Examen**

How can you exit from the shell?

Respuesta correcta!

exit

**Revisar respuesta**

**Activar Windows**

Ve a Configuración para activar Windows.

The taskbar at the bottom of the screen shows several pinned icons, including OneDrive, Mail, Photos, File Explorer, Edge, Google Chrome, and others. The system tray shows the date and time as 6/6/2025 17:42.

# TEXT-FU

The screenshot shows a web browser window with multiple tabs open. The main content area displays a lesson titled "1. stdout (Salida estándar)". It includes a sidebar with a navigation menu and a main panel with text and code examples. A sidebar on the right contains sections for "Ceremonias" and "Examen". The "Examen" section includes a question, a text input field, and a "Revisar respuesta" button.

**Ceremonias:**  
Pruebe un par de comandos:  
`$ ls -l /var/log > myoutput.txt  
$ echo Hola mundo > rm  
$ > algúnarchivo.txt`

**Examen:**  
¿Qué redirector utilizas para agregar salida a un archivo?  
¡Respuesta correcta!  
Revisar respuesta  
Continuar

The screenshot shows a web browser window with multiple tabs open. The main content area displays a lesson titled "2. stdin (Standard In)". It includes a sidebar with a navigation menu and a main panel with text and code examples. A sidebar on the right contains sections for "Ceremonias" and "Examen". The "Examen" section includes a question, a text input field, and a "Revisar respuesta" button.

**Ceremonias:**  
Try out a couple of commands:  
`$ echo < peanuts.txt > banana.txt  
$ ls < peanuts.txt > banana.txt  
$ pwd < peanuts.txt > banana.txt`

**Examen:**  
What redirector do you use to redirect stdin?  
¡Respuesta correcta!  
Revisar respuesta  
Continuar

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/stderr-standard-error-redirect". The page content is a lesson on standard error redirection. On the left, a sidebar lists numbered items from 1 to 14, with item 3 highlighted in green: "3. stderr (Error estándar)". The main content area contains text explaining that stderr (standard error) has descriptor 2, and shows examples of redirecting stderr to a file or /dev/null. A sidebar on the right contains sections for "Ceremonias" and "Examen".

**Ceremonias**

¿Qué hace el siguiente comando?

```
$ ls /fake/directorio >> cacahuetes.txt
```

Deberías ver solo los mensajes stderr en peanuts.txt.

¿Y si quisiera ver tanto la salida estándar como la salida estándar en el archivo "peanuts.txt"? También es posible hacerlo con descriptores de archivo:

```
$ ls /fake/directorio > cacahuetes.txt 2>&1
```

Esto envía los resultados de ls /fake/directorio al archivo "peanuts.txt" y luego redirige la salida estándar a la salida estándar mediante "2>&1". El orden de las operaciones es importante: "2>&1" envía la salida estándar a la salida estándar. En este caso, la salida estándar apunta a un archivo, por lo que "2>&1" también envía la salida estándar a un archivo. Por lo tanto, si abre el archivo "peanuts.txt", debería ver tanto la salida estándar como la salida estándar. En nuestro caso, el comando anterior solo genera la salida estándar.

Hay una forma más corta de redirigir tanto stdout como stderr a un archivo:

```
$ ls /fake/directorio >& cacahuetes.txt
```

¿Y si no quiero nada de eso y quiero eliminar por completo los mensajes de stderr? También puedes redirigir la salida a un archivo especial llamado /dev/null, que descartará cualquier entrada.

```
$ ls /fake/directorio >> /dev/null
```

**Examen**

¿Cuál es el redirector para stderr?

✓ ¡Respuesta correcta!

2>

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/env-environment". The page content is a lesson on environment variables. On the left, a sidebar lists numbered items from 1 to 14, with item 5 highlighted in green: "5. env (Environment)". The main content area contains text about the env command and its output. A sidebar on the right contains sections for "Ceremonias" and "Examen".

**5. env (Environment)**

Run the following command:

```
$ echo $HOME
```

You should see the path to your home directory, mine looks like /home/pete.

What about this command?

```
$ echo $USER
```

You should see your username!

Where is this information coming from? It's coming from your environment variables. You can view these by typing

```
$ env
```

This outputs a whole lot of information about the environment variables you currently have set. These variables contain useful information that the shell and other processes can use.

Here is a short example:

```
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/bin  
PWD=/home/user  
USER=pete
```

One particularly important variable is the PATH Variable. You can access these variables by sticking a \$ in front of the variable name like so:

**Ceremonias**

What does the following output? Why?

```
$ echo $HOME
```

**Examen**

How do you see your environment variables?

✓ ¡Respuesta correcta!

env

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linujourney.com/lesson/cut-command". The page content is for lesson 6, titled "6. cortar". A sidebar on the left lists various Linux commands with numbers 1 through 14. "6. cortar" is highlighted with a teal background. The main content area contains text explaining the "cut" command, several code examples in a code editor-like box, and a "Ceremonias" section with a question about the command's purpose. Below that is an "Examen" section with a question, a green "¡Respuesta correcta!" message, and a text input field containing "cut -c 1".

This screenshot shows the same browser window after navigating to the next lesson. The active tab is now "linujourney.com/lesson/paste-command". The content is for lesson 7, titled "7. paste". The sidebar shows "7. paste" is selected. The main content area explains the "paste" command, provides a sample file "sample2.txt" with text, and shows how to combine multiple lines into one using the command "\$ paste -s sample2.txt". It also discusses changing the delimiter from TAB to space. The "Ceremonias" and "Examen" sections are present, with the exam question asking what flag to use with "paste" to make everything go on one line, and the response input field containing "-s".

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/head-command". On the left, a sidebar titled "Text-Fu" lists various Linux commands. The "head" command is selected and highlighted in green. The main content area is titled "8. head". It contains text explaining that the command shows the first 10 lines of a file by default, and examples of using it with and without the -c flag. A section titled "Ceremonias" asks what the command does, and a section titled "Examen" asks for the flag to change the number of lines. The status bar at the bottom shows the date and time as 6/6/2025 17:55.

This screenshot shows the same browser window after navigating to the "tail" command lesson. The sidebar now highlights the "tail" command. The main content area is titled "9. tail". It explains that the command shows the last 10 lines by default, and how to change the number of lines with the -n flag. It also mentions the -f flag for following files. A section titled "Ceremonias" asks about the man page for tail, and a section titled "Examen" asks for the flag to follow a file. The status bar at the bottom shows the date and time as 6/6/2025 17:56.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/expand-unexpand-command". The page content is a lesson titled "10. expand and unexpand". It includes a sidebar with a list of 14 topics, where "10. expand and unexpand" is currently selected. The main content area contains text explaining the `expand` command, two code snippets, and a section for a test or exam.

**10. expand and unexpand**

In our lesson on the cut command, we had our sample.txt file that contained a tab. Normally TABs would usually show a noticeable difference but some text files don't show that well enough. Having TABs in a text file may not be the desired spacing you want. To change your TABs to spaces, use the expand command.

```
$ expand sample.txt
```

The command above will print output with each TAB converted into a group of spaces. To save this output in a file, use output redirection like below.

```
$ expand sample.txt > result.txt
```

Opposite to expand, we can convert back each group of spaces to a TAB with the unexpand command:

```
$ unexpand -a result.txt
```

**Ceremonias**

What happens if you just type expand with no file input?

**Examen**

What command is used to convert TABs to spaces?

✓ ¡Respuesta correcta!

expand  Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/join-split-command". The page content is a lesson titled "11. join and split". It includes a sidebar with a list of 14 topics, where "11. join and split" is currently selected. The main content area contains text explaining the `join` command, two code snippets, and a section for a test or exam.

**11. join and split**

The join command allows you to join multiple files together by a common field: Let's say I had two files that I wanted to join together:

```
file1.txt
1 John
2 Jane
3 Mary
```

```
file2.txt
1 Doe
2 Doe
3 Sue
```

```
$ join file1.txt file2.txt
1 John Doe
2 Jane Doe
3 Mary Sue
```

See how it joined together my files? They are joined together by the first field by default and the fields have to be identical, if they are not you can sort them, so in this case the files are joined via 1, 2, 3.

How would we join the following files?

```
file1.txt
```

**Ceremonias**

Join two files with different number of lines in each file, what happens?

**Examen**

What command would you use to join files named cat dog cow?

✓ ¡Respuesta correcta!

join cat dog cow  Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

The screenshot shows a Windows desktop with a browser window open to [linuxjourney.com/lesson/sort-command](https://linuxjourney.com/lesson/sort-command). The browser has several tabs open in the background, including 'Your Repositories', '(366) WhatsApp', 'OverTheWire: Wargar...', 'Learning | Linux Jour...', 'Solve Linux Shell | H...', 'Welcome to Hacker...', and others.

The main content area is titled '12. sort'. It explains that the sort command is useful for sorting lines. It includes two code snippets:

```
file1.txt
dog
cow
cat
elephant
bird

$ sort file1.txt
bird
cat
cow
dog
elephant
```

You can also do a reverse sort:

```
$ sort -r file1.txt
elephant
dog
cow
cat
```

**Ceremonias**

The real power of sort comes with its ability to be combined with other commands, try the following command and see what happens?

```
$ ls /etc | sort -rn
```

**Examen**

What flag do you use to do a reverse sort?

Respuesta correcta!

-r

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

At the bottom right of the screen, there is a system tray with icons for battery, signal, and time (18:02, 6/6/2025).

The screenshot shows a Windows desktop with a browser window open to [linuxjourney.com/lesson/tr-translate-command](https://linuxjourney.com/lesson/tr-translate-command). The browser has several tabs open in the background, including 'Your Repositories', '(366) WhatsApp', 'OverTheWire: Wargar...', 'Learning | Linux Jour...', 'Solve Linux Shell | H...', 'Welcome to Hacker...', and others.

The main content area is titled '13. tr (Translate)'. It explains that the tr (translate) command allows you to translate a set of characters into another set of characters. It includes a code snippet:

```
$ tr a-z A-Z
hello
HELLO
```

As you can see we made the ranges of a-z into A-Z and all text we type that is lowercase gets uppercase.

**Ceremonias**

Try the following command what happens?

```
$ tr -d ello
hello
```

**Examen**

What command is used to translate characters?

Respuesta correcta!

tl

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

At the bottom right of the screen, there is a system tray with icons for battery, signal, and time (18:02, 6/6/2025).

The screenshot shows a Windows desktop with a taskbar at the bottom containing icons for File Explorer, WhatsApp, OverTheWire: Wargar, Learning | Linux Journal, Solve Linux Shell | HackTheBox, and Welcome to HackerIt. The main window is a web browser displaying the URL [linuxjourney.com/lesson/uniq-unique-command](https://linuxjourney.com/lesson/uniq-unique-command). The page title is "Text-Fu" and the current lesson is "14. uniq (Unique)". The content area includes a sidebar with numbered links from 1 to 14, the lesson title, a brief description, and several code snippets. A sidebar on the right contains sections for "Ceremonias" and "Examen". The "Examen" section asks what command would remove duplicates in a file, with a text input field containing "uniq" and a "Revisar respuesta" button.

This screenshot shows the same Windows desktop and taskbar as the previous one. The main browser window now displays the URL [linuxjourney.com/lesson/grep-command](https://linuxjourney.com/lesson/grep-command). The sidebar shows the "Text-Fu" menu and the current lesson is "16. grep". The content area contains a sidebar with numbered links, the lesson title, a brief description, and several code snippets. A sidebar on the right contains sections for "Ceremonias" and "Examen". The "Examen" section asks what command finds a certain pattern, with a text input field containing "grep" and a "Revisar respuesta" button.

## ADVANCED TEXT-FU

linuxjourney.com/lesson/regular-expressions-regex

Texto avanzado-Fu

1. regex (expresiones regulares)

2. Editores de texto

3. Vim (Vi mejorado)

4. Patrones de búsqueda de Vim

5. Navegación de Vim

6. Vim añadiendo texto

7. Edición de Vim

8. Guardar y salir de Vim

9. Emacs

10. Emacs manipula archivos

11. Navegación por el búfer de Emacs

12. Edición de Emacs

13. Salida y ayuda de Emacs

Ceremonias

Intente combinar expresiones regulares con grep y busque en algunos archivos.

grep [expresión regular aquí] [archivo]

Examen

¿Qué expresión regular usarías para hacer coincidir un solo carácter?

✓ ¡Respuesta correcta!

Continuar

Activar Windows

Ve a Configuración para activar Windows.

18:09 6/6/2025

linuxjourney.com/lesson/text-editors-vim-or-emacs

Advanced Text-Fu

1. regex (Regular Expressions)

2. Text Editors

3. Vim (Vi Improved)

4. Vim Search Patterns

5. Vim Navigation

6. Vim Appending Text

7. Vim Editing

8. Vim Saving and Exiting

9. Emacs

10. Emacs Manipulate Files

11. Emacs Buffer Navigation

12. Emacs Editing

13. Emacs Exiting and Help

Ceremonias

Tome un tour de vim y emacs:

Vim  
emacs

Examen

No hay preguntas más allá.

✓ ¡Respuesta correcta!

Continuar

Activar Windows

Ve a Configuración para activar Windows.

18:09 6/6/2025

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/vim-text-editor". The left sidebar lists "Advanced Text-Fu" topics: 1. regex (Regular Expressions), 2. Text Editors, 3. Vim (Vi Improved) (highlighted in green), 4. Vim Search Patterns, 5. Vim Navigation, 6. Vim Appending Text, 7. Vim Editing, 8. Vim Saving and Exiting, 9. Emacs, 10. Emacs Manipulate Files, 11. Emacs Buffer Navigation, 12. Emacs Editing, and 13. Emacs Exiting and Help. The main content area is titled "3. Vim (Vi Improved)". It contains text explaining that Vim stands for vi (Improved) and is a super lightweight text editor. A search bar at the bottom has "vim" typed into it. To the right, there are two sections: "Ceremonias" (No exercises for this lesson) and "Examen" (No questions move along!). The "Examen" section includes a green button for "¡Respuesta correcta!", a text input field, a blue "Revisar respuesta" button, and an orange "Continuar" button. The status bar at the bottom shows "18:10 6/6/2025".

This screenshot shows the same browser window with a different lesson page. The active tab is now "linuxjourney.com/lesson/vim-search-patterns". The left sidebar lists "Advanced Text-Fu" topics, with "4. Patrones de búsqueda de Vim" highlighted in green. The main content area is titled "4. Patrones de búsqueda de Vim". It explains how to search for a pattern in Vim by pressing / followed by the pattern and Enter. It shows examples of searching for "/bonito" and "?bonito" in a file containing "Mi bonito archivo es muy bonito.". Below this, it says "El comando de búsqueda ? buscará en el archivo de texto hacia atrás, por lo que en el ejemplo anterior, el último texto aparecería primero." To the right, there are "Ceremonias" and "Examen" sections. The "Examen" section asks "¿Qué tecla se utiliza para buscar en vim?". It includes a green "¡Respuesta correcta!" button, a text input field with a question mark, a blue "Revisar respuesta" button, and an orange "Continuar" button. The status bar at the bottom shows "18:11 6/6/2025".

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/vim-navigation". The left sidebar of the website lists various lessons: 1. regex (Regular Expressions), 2. Text Editors, 3. Vim (Vi Improved), 4. Vim Search Patterns, 5. Vim Navigation (which is highlighted in green), 6. Vim Appending Text, 7. Vim Editing, 8. Vim Saving and Exiting, 9. Emacs, 10. Emacs Manipulate Files, 11. Emacs Buffer Navigation, 12. Emacs Editing, and 13. Emacs Exiting and Help.

**5. Vim Navigation**

Now you may notice, the mouse is nowhere is use here. To navigate a text document in vim, use the following keys:

- h or the left arrow - will move you left one character
- k or the up arrow - will move you up one line
- j or the down arrow - will move you down one line
- l or the right arrow - will move you right one character

**Ceremonias**

No exercises for this lesson.

**Examen**

What letter is used to move down?

i Revisar respuesta

Continuar join Activar Windows  
Ve a Configuración para activar Windows.

At the bottom of the browser window, there is a taskbar with icons for various applications like WhatsApp, OverTheWire, Learning, Solve Linux Shell, and Welcome to HackerIt. The system tray shows the date and time as 6/6/2025 18:12.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/vim-inserting-appending-text". The left sidebar of the website lists various lessons: 1. regex (Regular Expressions), 2. Text Editors, 3. Vim (Vi Improved), 4. Vim Search Patterns, 5. Vim Navigation, 6. Vim Appending Text (which is highlighted in green), 7. Vim Editing, 8. Vim Saving and Exiting, 9. Emacs, 10. Emacs Manipulate Files, 11. Emacs Buffer Navigation, 12. Emacs Editing, and 13. Emacs Exiting and Help.

**6. Vim Appending Text**

Now you may have noticed if you tried to type something you wouldn't be able to. That's because you are in command mode. This can get pretty confusing especially if you just want to open a file and enter text. The command mode is used for when you enter commands like h,j,k,l etc. To insert text you'll need to enter insert mode first.

- i - insert text before the cursor
- o - insert text on the previous line
- o - insert text on the next line
- a - append text after the cursor
- A - append text at the end of the line

Notice how when you type any of these insertion modes, you'll see that vim has entered insert mode at the bottom of the shell. To exit insert mode and go back to command mode, just hit the Esc key.

**Ceremonias**

Play around with entering and exiting insertion mode.

**Examen**

What key is used to insert text before the cursor?

i Revisar respuesta

Continuar Activar Windows  
Ve a Configuración para activar Windows.

At the bottom of the browser window, there is a taskbar with icons for various applications like WhatsApp, OverTheWire, Learning, Solve Linux Shell, and Welcome to HackerIt. The system tray shows the date and time as 6/6/2025 18:13.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/vim-editing". The left sidebar lists various lessons: 1. regex (Regular Expressions), 2. Text Editors, 3. Vim (Vi Improved), 4. Vim Search Patterns, 5. Vim Navigation, 6. Vim Appending Text, 7. Vim Editing (which is highlighted in green), 8. Vim Saving and Exiting, 9. Emacs, 10. Emacs Manipulate Files, 11. Emacs Buffer Navigation, 12. Emacs Editing, and 13. Emacs Exiting and Help. The main content area is titled "7. Vim Editing" and contains the following text:

Now that we have a couple of lines written, let's edit it a bit more and remove some cruft.

- x - used to cut the selected text also used for deleting characters
- dd - used to delete the current line
- y - yank or copy whatever is selected
- yy - yank or copy the current line
- p - paste the copied text before the cursor

## Ceremonias

I know this lesson added some oddballs, open up a text editor and play around with these.

## Examen

What character is used to delete an entire line?

✓ |Respuesta correcta!

Revisar respuesta

Activar Windows  
Ve a Configuración para activar Windows.

Continuar

18:13  
6/6/2025

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/vim-saving-and-exiting". The left sidebar lists various lessons: 1. regex (Regular Expressions), 2. Text Editors, 3. Vim (Vi Improved), 4. Vim Search Patterns, 5. Vim Navigation, 6. Vim Appending Text, 7. Vim Editing, 8. Vim Saving and Exiting (which is highlighted in green), 9. Emacs, 10. Emacs Manipulate Files, 11. Emacs Buffer Navigation, 12. Emacs Editing, and 13. Emacs Exiting and Help. The main content area is titled "8. Vim Saving and Exiting" and contains the following text:

Now that you've done your editing it's time to actually save and quit out of vim:

- :w - writes or saves the file
- :q - quit out of vim
- :wq - write and then quit
- :q! - quit out of vim without saving the file
- ZZ - equivalent of :wq, but one character faster
- u - undo your last action
- Ctrl-r - redo your last action

You may not think ZZ is necessary, but you'll eventually see that your fingers may tend to lean towards this rather than :wq.

Whew that was a lot of information to take about Vim. Now that you know some basic commands and navigation, you can start editing some text files. There are many more options you can use in vim to increase your ability to master this text editor, head on to Vim's online guide to take a look.

## Ceremonias

No exercises for this lesson.

## Examen

How do you quit out of vim without saving?

✓ |Respuesta correcta!

Revisar respuesta

Activar Windows  
Ve a Configuración para activar Windows.

Continuar

18:14  
6/6/2025

**9. Emacs**

Emacs is for users who want an extremely powerful text editor, which may be an understatement because you essentially live in emacs. You can do all your code editing, file manipulation, etc all within emacs. It's a bit slower to load up and the learning curve is a bit steeper than vim, but if you want a powerful editor that is extremely extensible, this is the one for you. When I say extensible, I literally mean you can write up scripts for emacs that extend its functionality.

To start emacs just use:

```
emacs
```

You should be greeted with the default welcome buffer.

Buffers in emacs is what your text resides in. So if you open up a file, a buffer is used to store that file's content. You can have multiple buffers open at the same time and you can easily switch between buffers.

**10. Emacs Manipulate Files**

In a lot (if not all) of Emacs documentation, you will see the syntax C-[letter]. This just means hit the Ctrl-letter, but for shorthand purposes, we'll call Ctrl with C. If you see syntax such as M-[letter], that means use the Meta key, most commonly the Alt key.

**Saving files**

```
C-x C-s - Save a file  
C-x C-w - Save file as  
C-x s - Save all
```

The save file options will prompt you if you want to save each file.

**Opening a file**

```
C-x C-f
```

This will prompt you to type a filename to open. If you do not have a file that already exists, it will create a new file. You can load up a directory as well.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "11. Emacs Buffer Navigation" from "linuxjourney.com". The page content includes a sidebar with a navigation menu and the main content area which details buffer navigation commands like C-x b, C-x right arrow, C-x left arrow, C-x k, and C-x 2.

**Advanced Text-Fu**

- 1. regex (Regular Expressions)
- 2. Text Editors
- 3. Vim (Vi Improved)
- 4. Vim Search Patterns
- 5. Vim Navigation
- 6. Vim Appending Text
- 7. Vim Editing
- 8. Vim Saving and Exiting
- 9. Emacs
- 10. Emacs Manipulate Files
- 11. Emacs Buffer Navigation**
- 12. Emacs Editing
- 13. Emacs Exiting and Help

**11. Emacs Buffer Navigation**

To move around buffers (or files you're visiting) use the following commands:

**Switch buffers**

```
C-x b - switch buffer  
C-x right arrow - right-cycle through buffer  
C-x left arrow - left-cycle through buffer
```

**Close the buffer**

```
C-x k
```

**Split the current buffer**

```
C-x 2
```

This allows you see multiple buffers on one screen. To move between these buffers use: C-x o

**Set a single buffer as the current screen**

```
C-x 1
```

If you ever used a terminal multiplexer like screen and tmux, the buffer commands will feel very familiar.

**Ceremonias**  
Play around with buffers.

**Examen**  
How do you kill a buffer?

Respuesta correcta!

C-x k |

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

18:16 6/6/2025

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "12. Emacs Editing" from "linuxjourney.com". The page content includes a sidebar with a navigation menu and the main content area which details text navigation keys like C-up arrow, C-down arrow, C-left arrow, C-right arrow, and M->.

**Advanced Text-Fu**

- 1. regex (Regular Expressions)
- 2. Text Editors
- 3. Vim (Vi Improved)
- 4. Vim Search Patterns
- 5. Vim Navigation
- 6. Vim Appending Text
- 7. Vim Editing
- 8. Vim Saving and Exiting
- 9. Emacs
- 10. Emacs Manipulate Files
- 11. Emacs Buffer Navigation
- 12. Emacs Editing**
- 13. Emacs Exiting and Help

**12. Emacs Editing**

**Text Navigation**

```
C-up arrow : move up one paragraph  
C-down arrow: move down one paragraph  
C-left arrow: move one word left  
C-right arrow: move one word right  
M-> : move to the end of the buffer
```

With text navigation, your regular text buttons work as they should, home, end, page up, page down and the arrow keys, etc.

**Cutting and Pasting**

To cut (kill) or paste (yank) in Emacs you'll need to be able to select text first. To select text, move your cursor to where you want to cut or paste and hit

```
C-space key
```

then you can use the navigation keys to select the text you want. Now you can do the cut and paste like so:

```
C-w : cut  
C-y : yank
```

**Ceremonias**  
Play around with text navigation.

**Examen**  
How do you move to the end of the buffer?

Respuesta correcta!

M-> |

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

18:17 6/6/2025

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "13. Emacs Exiting and Help" from "linuxjourney.com/lesson/emacs-exiting-and-help". The page content includes a sidebar with navigation links for Advanced Text-Fu, a main section with text and code snippets, and a sidebar with sections for Ceremonias and Examen.

**Advanced Text-Fu**

- 1. regex (Regular Expressions)
- 2. Text Editors
- 3. Vim (Vi Improved)
- 4. Vim Search Patterns
- 5. Vim Navigation
- 6. Vim Appending Text
- 7. Vim Editing
- 8. Vim Saving and Exiting
- 9. Emacs
- 10. Emacs Manipulate Files
- 11. Emacs Buffer Navigation
- 12. Emacs Editing
- 13. Emacs Exiting and Help**

**13. Emacs Exiting and Help**

To close out of emacs  
C-x C-c

If you have any open buffers, it will ask you to save it before closing out of emacs.

Confused?  
C-h C-h : help menu

Undo  
C-x u

As you can see Emacs has more moving parts, so the learning curve is a little steeper. In exchange though, you get a very powerful text editor.

**Ceremonias**

Visit the Emacs site to learn about more commands. [Emacs](#)

**Examen**

How do you access the help menu?

Respuesta correcta

C-h C-h

Activar Windows  
Ve a Configuración para activar Windows.

## USER MANAGEMENT

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "1. Users and Groups" from "linuxjourney.com/lesson/users-and-groups". The page content includes a sidebar with navigation links for User Management, a main section with text and code snippets, and a sidebar with sections for Ceremonias and Examen.

**User Management**

- 1. Users and Groups**
- 2. root
- 3. ./etc/passwd
- 4. ./etc/shadow
- 5. ./etc/group
- 6. User Management Tools

**1. Users and Groups**

In any traditional operating system, there are users and groups. They exist solely for access and permissions. When running a process, it will run as the owner of that process whether that is Jane or Bob. File access and ownership is also permission dependent. You wouldn't want Jane to see Bob's documents and vice versa.

Each user has their own home directory where their user specific files get stored, this is usually located in /home/username, but can vary in different distributions.

The system uses user ids (UID) to manage users, usernames are the friendly way to associate users with identification, but the system identifies users by their UID. The system also uses groups to manage permissions, groups are just sets of users with permission set by that group, they are identified by the system with their group ID (GID).

In Linux, you'll have users in addition to the normal humans that use the system. Sometimes these users are system daemons that continuously run processes to keep the system functioning. One of the most important users is root or superuser, root is the most powerful user on the system, root can access any file and start and terminate any process. For that reason, it can be dangerous to operate as root all the time, you could potentially remove system critical files. Luckily, if root access is needed and a user has root access, they can run a command as root instead with the sudo command. The sudo command (superuser do) is used to run a command with root access, we'll go more in depth on how a user receives root access in a later lesson.

Go ahead and try to view a protected file like /etc/shadow:

\$ cat /etc/shadow

Respuesta correcta

sudo

Activar Windows  
Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/root-user". The page content is a lesson titled "2. raíz". It discusses how to obtain root access using the "sudo" command. A code input field contains "\$ su". Below it, a text block explains the disadvantages of using "su" instead of "sudo". Another text block asks how to identify who has root access. A sidebar on the right is titled "Ceremonias" and "Examen". The "Examen" section asks which file lists users with sudo access, with the answer "/etc/sudoers" being correct.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/etc-passwd-file". The page content is a lesson titled "3. /etc/passwd". It explains that user names are not real identifications and that the system uses a user ID (UID) to identify users. A code input field contains "\$ cat /etc/passwd". Below it, a text block shows an example of the /etc/passwd file format. A sidebar on the right is titled "Ceremonias" and "Examen". The "Examen" section asks what is denoted by a blank password field in /etc/passwd, with the answer "1" being correct.

User Management

- 1. Users and Groups
- 2. root
- 3. /etc/passwd
- 4. /etc/shadow**
- 5. /etc/group
- 6. User Management Tools

**4. /etc/shadow**

The /etc/shadow file is used to store information about user authentication. It requires superuser read permissions.

```
$ sudo cat /etc/shadow
root:MyEPTEa$6Nonsense:15000:0:99999:7:::
```

You'll notice that it looks very similar to the contents of /etc/passwd, however in the password field you'll see an encrypted password. The fields are separated by colons as follows:

1. Username
2. Encrypted password
3. Date of last password changed - expressed as the number of days since Jan 1, 1970. If there is a 0 that means the user should change their password the next time they login
4. Minimum password age - Days that a user will have to wait before being able to change their password again
5. Maximum password age - Maximum number of days before a user has to change their password
6. Password warning period - Number of days before a password is going to expire
7. Password inactivity period - Number of days after a password has expired to allow login with their password
8. Account expiration date - date that user will not be able to login
9. Reserved field for future use

**Ceremonias**  
Take a look at the /etc/shadow file

**Examen**  
No questions move along!

**Activar Windows**  
Ve a Configuración para activar Windows.

User Management

- 1. Users and Groups
- 2. root
- 3. /etc/passwd
- 4. /etc/shadow
- 5. /etc/group**
- 6. User Management Tools

**5. /etc/group**

Another file that is used in user management is the /etc/group file. This file allows for different groups with different permissions.

```
$ cat /etc/group
root:*:0:pete
```

Very similar to the /etc/password field, the /etc/group fields are as follows:

1. Group name
2. Group password - there isn't a need to set a group password, using an elevated privilege like sudo is standard. A \*\*\* will be put in place as the default value.
3. Group ID (GID)
4. List of users - you can manually specify users you want in a specific group

**Ceremonias**  
Run the command **groups**. What do you see?

**Examen**  
What is the GID of root?

**Activar Windows**  
Ve a Configuración para activar Windows.

User Management

1. Users and Groups

2. root

3. /etc/passwd

4. /etc/shadow

5. /etc/group

6. User Management Tools

## 6. User Management Tools

Most enterprise environments are using management systems to manage users, accounts and passwords. However, on a single machine computer there are useful commands to run to manage users.

### Adding Users

You can use the adduser or the useradd command. The adduser command contains more helpful features such as making a home directory and more. There are configuration files for adding new users that can be customized depending on what you want to allocate to a default user.

```
$ sudo useradd bob
```

You'll see that the above command creates an entry in /etc/passwd for bob, sets up default groups and adds an entry to the /etc/shadow file.

### Removing Users

To remove a user, you can use the userdel command.

```
$ sudo userdel bob
```

This basically does its best to undo the file changes by useradd.

### Changing Passwords

```
$ passwd bob
```

This will allow you to change the password of yourself or another user (if you are root).

## Ceremonias

Create a new user then change their password and login as the new user.

## Examen

What command is used to change a password?

✓ ¡Respuesta correcta!

passwd

Revisar respuesta

Return

Activar Windows  
Ve a Configuración para activar Windows.

# PERMISSIONS

Permissions

1. File Permissions

2. Modifying Permissions

3. Ownership Permissions

4. Umask

5. Setuid

6. Selgid

7. Process Permissions

8. The Sticky Bit

## 1. File Permissions

As we learned previously, files have different permissions or file modes. Let's look at an example:

```
$ ls -l Desktop/
drwxr-xr-x 2 pete penguins 4096 Dec 1 11:45 .
```

There are four parts to a file's permissions. The first part is the filetype, which is denoted by the first character in the permissions, in our case since we are looking at a directory it shows d for the filetype. Most commonly you will see a - for a regular file.

The next three parts of the file mode are the actual permissions. The permissions are grouped into 3 bits each. The first 3 bits are user permissions, then group permissions and then other permissions. I've added the pipe to make it easier to differentiate.

```
d | rwx | r-x | r-x
```

Each character represents a different permission:

- r: readable
- w: writable
- x: executable (basically an executable program)
- : empty

So in the above example, we see that the user pete has read, write and execute permissions on the file. The group penguins has read and execute permissions. And finally, the other users (everyone else) has read and execute permissions.

## Ceremonias

Use the ls -l command on multiple files and recite their permissions, user and group.

## Examen

What permission bit is used for executable?

✓ ¡Respuesta correcta!

x

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

**Permissions**

- 1. File Permissions
- 2. Modifying Permissions**
- 3. Ownership Permissions
- 4. Umask
- 5. Setuid
- 6. Setgid
- 7. Process Permissions
- 8. The Sticky Bit

**2. Modifying Permissions**

Changing permissions can easily be done with the `chmod` command.

First, pick which permission set you want to change, user, group or other. You can add or remove permissions with a + or -, let's look at some examples.

**Adding permission bit on a file**

```
$ chmod u+x myfile
```

The above command reads like this: change permission on myfile by adding executable permission bit on the user set. So now the user has executable permission on this file!

**Removing permission bit on a file**

```
$ chmod u-x myfile
```

**Adding multiple permission bits on a file**

```
$ chmod ug+w
```

There is another way to change permissions using numerical format. This method allows you to change permissions all at once. Instead of using r, w, or x to represent permissions, you'll use a numerical representation for a single permission set. So no need to specify the group with g or the user with u.

The numerical representations are seen below:

- 4: read permission
- 2: write permission
- 1: execute permission

**Ceremonias**

Change some basic text file permissions and see the bits changing as you do an ls -l.

**Examen**

What number represents the read permission when using numerical format?

Respuesta correcta!

4

**Continuar**

Activar Windows  
Ve a Configuración para activar Windows.

**Permissions**

- 1. File Permissions
- 2. Modifying Permissions
- 3. Ownership Permissions**
- 4. Umask
- 5. Setuid
- 6. Setgid
- 7. Process Permissions
- 8. The Sticky Bit

**3. Ownership Permissions**

In addition to modifying permissions on files, you can also modify the group and user ownership of the file as well.

**Modify user ownership**

```
$ sudo chown patty myfile
```

This command will set the owner of myfile to patty.

**Modify group ownership**

```
$ sudo chgrp whales myfile
```

This command will set the group of myfile to whales.

**Modify both user and group ownership at the same time**

If you add a colon and groupname after the user you can set both the user and group at the same time.

```
$ sudo chown patty:whales myfile
```

**Ceremonias**

Modify the group and user of some test files. Afterwards take a look at the permissions with ls -l.

**Examen**

What command do you use to change user ownership?

Respuesta correcta!

chown

**Continuar**

Activar Windows  
Ve a Configuración para activar Windows.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "4. Umask" from "linuxjourney.com/lesson/umask". The page content discusses the umask command, explaining that it removes permissions from files created by the user. It includes a code block showing the command \$ umask 022 and a note about its default value of 022. A sidebar on the right contains sections for "Ceremonias" and "Examen". The "Ceremonias" section lists three tasks: creating a file, modifying the umask, and checking permissions again. The "Examen" section asks what command changes default file permissions, with a correct answer of "umask" and a button to "Revisar respuesta". The browser interface includes a navigation bar with icons for search, refresh, and back/forward.

This screenshot shows the same browser window on a different lesson page, "5. Setuid", from "linuxjourney.com/lesson/setuid-set-user-id". The page content explains the Set User ID (SUID) permission bit. It includes a code block for the passwd command and a file listing for /etc/shadow. A sidebar on the right contains sections for "Ceremonias" and "Examen". The "Ceremonias" section asks to look at the permission for /etc/passwd. The "Examen" section asks what number represents SUID, with a correct answer of "4" and a button to "Revisar respuesta". The browser interface is identical to the previous screenshot.

**Permissions**

- 1. File Permissions
- 2. Modifying Permissions
- 3. Ownership Permissions
- 4. Umask
- 5. Setuid
- 6. Setgid**
- 7. Process Permissions
- 8. The Sticky Bit

**6. Setgid**

Similar to the set user ID permission bit, there is a set group ID (SGID) permission bit. This bit allows a program to run as if it was a member of that group.

Let's look at one example:

```
$ ls -l /usr/bin/wall  
-rwxr-sr-x 1 root tty 19024 Dec 14 11:45 /usr/bin/wall
```

We can see now that the permission bit is in the group permission set.

**Modifying SGID**

```
$ sudo chmod g+s myfile  
$ sudo chmod 2555 myfile
```

The numerical representation for SGID is 2.

**Ceremonias**

No exercises for this lesson.

**Examen**

What number represents the SGID?

2 Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

**Permisos**

- 1. Permisos de archivos
- 2. Modificación de permisos
- 3. Permisos de propiedad
- 4. Umask
- 5. Setuid
- 6. Setgid
- 7. Permisos de proceso**
- 8. La parte pegajosa

**7. Permisos de proceso**

Hablemos un poco sobre los permisos de proceso. ¿Recuerdas que le dije que al ejecutar el comando `passwd` con el bit de permiso SUID habilitado, el programa se ejecuta como root? Es cierto, pero ¿significa que, al ser root temporalmente, puedes modificar las contraseñas de otros usuarios? ¡Afortunadamente no!

Esto se debe a la gran cantidad de UID que implementa Linux. Hay tres UID asociados a cada proceso:

Al iniciar un proceso, este se ejecuta con los mismos permisos que el usuario o grupo que lo ejecutó, lo que se conoce como **ID de usuario efectivo**. Este UID se utiliza para otorgar derechos de acceso a un proceso. Por lo tanto, si Bob ejecuta el comando fácil, el proceso se ejecutará bajo su nombre y cualquier archivo que cree será de su propiedad.

Existe otro UID, denominado **ID de usuario real**, que es el ID del usuario que inició el proceso. Este se utiliza para identificar al usuario que lo inició.

Un último UID es el **ID de usuario guardado**. Esto permite que un proceso alterne entre el UID efectivo y el real, y viceversa. Esto es útil porque no queremos que nuestro proceso se ejecute con privilegios elevados constantemente; simplemente es recomendable usar privilegios especiales en momentos específicos.

Ahora vamos a unir todo esto mirando el comando `passwd` una vez más.

Al ejecutar el comando `passwd`, su UID efectivo es su ID de usuario; por ahora, digamos 500. Pero recuerde, el comando `passwd` tiene habilitado el permiso SUID. Por lo tanto, al ejecutarlo, su UID efectivo es 0 (0 es el UID del usuario root). Ahora, este programa puede acceder a los archivos como root.

**Ceremonias**

Aún no hemos discutido los procesos, aún podemos echar un vistazo a este cambio que ocurre en tiempo real:

1. Abra una ventana de terminal y ejecute el comando: `watch -n 1 "ps aux | grep passwd"`. Esto supervisará el proceso `passwd`.
2. Abra una segunda ventana de terminal y ejecute: `passwd`
3. Observa la primera ventana de terminal; verás que se inicia un proceso para `passwd`. La primera columna de la tabla de procesos es el ID de usuario efectivo; ¡es el usuario root!

**Examen**

¿Qué UID decide qué acceso conceder?

Effective Revisar respuesta

Continuar Activar Windows Ve a Configuración para activar Windows.

The screenshot shows a Windows desktop environment. A browser window is open to a Linux journey lesson page. The left sidebar lists various Linux concepts, with '8. The Sticky Bit' selected. The main content area shows a section titled '8. The Sticky Bit' with text explaining the sticky bit and its numerical representation. Below this are two code snippets:

```
$ ls -ld /tmp  
drwxrwxrwt 6 root root 4096 Dec 15 11:45 /tmp
```

```
$ sudo chmod +t mydir  
$ sudo chmod 1755 mydir
```

The text states that the numerical representation for the sticky bit is 1. To the right of the main content is a 'Ceremonias' section asking what other files/directories have a sticky bit enabled. Below it is an 'Examen' section with a question about symbols and a dropdown menu with 'touch', 'tee', and 'tr'. A green checkmark indicates a correct response. Buttons for 'Revisar respuesta' (Review answer) and 'Continuar' (Continue) are visible.

# PROCESSES

The screenshot shows a Windows desktop environment. A browser window is open to a Linux journey lesson page. The left sidebar lists various Linux concepts, with '1. ps (Processes)' selected. The main content area shows a section titled '1. ps (Processes)' with text explaining the ps command. It lists fields like USER, PID, %CPU, %MEM, VSZ, RSS, TTY, STAT, START, TIME, and COMMAND. It also notes that the ps command can get messy and suggests using flags like u and top. Below this is a code snippet:

```
$ top
```

To the right of the main content is a 'Ceremonias' section asking what ps flag is used to view detailed information about processes. Below it is an 'Examen' section with a question about the ps flag and a dropdown menu with 'Ubuntu' and 'uniq'. A green checkmark indicates a correct response. Buttons for 'Revisar respuesta' (Review answer) and 'Continuar' (Continue) are visible.

The screenshot shows a Windows desktop environment with a web browser open to [linuxjourney.com/lesson/controlling-terminal](https://linuxjourney.com/lesson/controlling-terminal). The browser has multiple tabs open, including 'Your Repositories', '(365) WhatsApp', 'OverTheWire: Wargar...', 'Learning | Linux Jour...', 'Solve Linux Shell | H...', 'Welcome to Hacke...', and the main content tab.

The main content area displays a lesson titled '2. Controlling Terminal'. The sidebar on the left lists numbered steps: 1. ps (Processes), 2. Controlling Terminal (which is selected and highlighted in green), 3. Process Details, 4. Process Creation, 5. Process Termination, 6. Signals, 7. kill (Terminate), 8. nice ness, 9. Process States, 10. /proc filesystem, and 11. Job Control.

The lesson content discusses TTY fields in ps output and the difference between regular terminal devices and pseudoterminal devices. It includes code snippets and instructions for interacting with the terminal.

On the right side, there are two sections: 'Ceremonias' (with a note to look at ps output and list unique TTY values) and 'Examen' (a question about processes without a controlling terminal, with a correct answer of 'kernel' and a 'Continuar' button).

This screenshot shows the same Windows desktop environment with the browser now on the 'Process Details' lesson page at [linuxjourney.com/lesson/process-details](https://linuxjourney.com/lesson/process-details).

The sidebar and overall layout are identical to the previous screenshot, showing the same numbered steps and main content area.

The main content area discusses what processes are and how they work, mentioning the kernel's role in managing resources. It includes a bulleted list of what processes do:

- The status of the process
- The resources the process is using and receives
- The process owner
- Signal handling (more on that later)
- And basically everything else

The right side 'Examen' section asks 'What manages and controls processes?' with a correct answer of 'kernel' and a 'Continuar' button.

The screenshot shows a Microsoft Edge browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/process-creation". The page content is a lesson on process creation, with the fourth item in the sidebar ("4. Process Creation") highlighted in green. The main text discusses how a new process is created by cloning the existing process using the fork system call. It also mentions the parent process ID (PPID) and the original process becoming its parent. A terminal window shows the command \$ ps 1. Below the text, there's a question about what system call creates a new process, with a correct answer of "fork" and a "Revisar respuesta" button.

The screenshot shows a Microsoft Edge browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/process-termination". The page content is a lesson on process termination, with the fifth item in the sidebar ("5. Process Termination") highlighted in green. The main text explains the \_exit system call and the termination status of 0. It also discusses orphan processes and zombie processes. A terminal window shows the command 0. Below the text, there's a question about the most common termination status, with a correct answer of "0" and a "Revisar respuesta" button.

The screenshot shows a web browser window with the URL [linuxjourney.com/lesson/process-signals](https://linuxjourney.com/lesson/process-signals). The page is titled "Process Signals". On the left, a sidebar lists numbered steps: 1. ps (Processes), 2. Controlling Terminal, 3. Process Details, 4. Process Creation, 5. Process Termination, 6. Signals (which is selected and highlighted in green), 7. kill (Terminate), 8. nice ness, 9. Process States, 10. /proc filesystem, and 11. Job Control. The main content area starts with a section titled "Signal process" which explains how signals are delivered to processes. It then moves to "Common signals", listing several signal types with their meanings. Below this, it notes that numbers can be used instead of symbolic names. A note about SIGKILL follows, stating it's unblockable. The right side of the page features a "Ceremonias" section with a message "No exercises for this lesson." and an "Examen" section asking "What signal is unblockable?" with a correct answer of "SIGKILL". There are "Revisar respuesta" and "Continuar" buttons.

This screenshot shows the same web browser window at the URL [linuxjourney.com/lesson/killing-processes](https://linuxjourney.com/lesson/killing-processes). The sidebar now highlights step 7. kill (Terminate). The main content area is titled "7. kill (Terminate)". It explains that you can send signals to terminate processes, with the kill command being the most common. It shows examples of using kill with a PID (e.g., \$ kill 12445) and with a signal number (e.g., \$ kill -9 12445). It also discusses the differences between various signals like SIGHUP, SIGINT, SIGTERM, SIGKILL, and SIGSTOP. The right side has a "Ceremonias" section with the message "Kill some processes using different signals." and an "Examen" section asking "What is the signal name for the default kill command?" with a correct answer of "SIGTERM". There are "Revisar respuesta" and "Continuar" buttons.

The screenshot shows a Microsoft Edge browser window with the URL [linuxjourney.com/lesson/process-niceness](https://linuxjourney.com/lesson/process-niceness). The page content is as follows:

**Processes**

- 1. ps (Processes)
- 2. Controlling Terminal
- 3. Process Details
- 4. Process Creation
- 5. Process Termination
- 6. Signals
- 7. kill (Terminate)
- 8. niceness**
- 9. Process States
- 10. /proc filesystem
- 11. Job Control

**When you run multiple things on your computer, like perhaps Chrome, Microsoft Word or Photoshop at the same time, it may seem like these processes are running at the same time, but that isn't quite true.**

Processes use the CPU for a small amount of time called a time slice. Then they pause for milliseconds and another process gets a little time slice. By default, process scheduling happens in this round-robin fashion. Every process gets enough time slices until it's finished processing. The kernel handles all of these switching of processes and it does a pretty good job at it most of the time.

Processes aren't able to decide when and how long they get CPU time, if all processes behaved normally they would each (roughly) get an equal amount of CPU time. However, there is a way to influence the kernel's process scheduling algorithm with a nice value. Niceness is a pretty weird name, but what it means is that processes have a number to determine their priority for the CPU. A high number means the process is nice and has a lower priority for the CPU and a low or negative number means the process is not very nice and it wants to get as much of the CPU as possible.

**\$ top**

You can see a column for NI right now, that is the niceness level of a process.

To change the niceness level you can use the nice and renice commands:

**\$ nice -n 5 apt upgrade**

The nice command is used to set priority for a new process. The renice command is used to set priority on an existing process.

**\$ renice 10 -p 3245**

**Ceremonias**

What processes aren't very nice and why?

**Examen**

If I want a process to get more CPU priority, do I use a lower or higher nice number?

Respuesta correcta

Lower

**Revisar respuesta**

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

18:56 6/6/2025

The screenshot shows a Microsoft Edge browser window with the URL [linuxjourney.com/lesson/process-states](https://linuxjourney.com/lesson/process-states). The page content is as follows:

**Processes**

- 1. ps (Processes)
- 2. Controlling Terminal
- 3. Process Details
- 4. Process Creation
- 5. Process Termination
- 6. Signals
- 7. kill (Terminate)
- 8. niceness
- 9. Process States**
- 10. /proc filesystem
- 11. Job Control

**9. Process States**

Let's take a look at the ps aux command again:

**\$ ps aux**

In the STAT column, you'll see lots of values. A Linux process can be in a number of different states. The most common state codes you'll see are described below:

- R: running or runnable, it is just waiting for the CPU to process it
- S: interruptible sleep, waiting for an event to complete, such as input from the terminal
- D: uninterruptible sleep, processes that cannot be killed or interrupted with a signal, usually to make them go away you have to reboot or fix the issue
- Z: Zombie, we discussed in a previous lesson that zombies are terminated processes that are waiting to have their statuses collected
- T: Stopped, a process that has been suspended/stopped

**Ceremonias**

Take a look at the running processes on your system and check out their process states.

**Examen**

What STAT code is used to represent an uninterruptible sleep?

Respuesta correcta

dd

**Revisar respuesta**

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

18:56 6/6/2025

**Ceremonias**  
No exercises for this lesson.

**Examen**  
What filesystem stores process information?  
✓ |Respuesta correcta!  
/proc Revisar respuesta  
Continuar Activar Windows  
Ve a Configuración para activar Windows.

**Ceremonias**  
Move some jobs between the background and the foreground

**Examen**  
What command is used to list background jobs?  
✓ |Respuesta correcta!  
jobs Revisar respuesta  
Return Activar Windows  
Ve a Configuración para activar Windows.

# PACKAGES

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "linuxjourney.com/lesson/software-distribution". The page content is a lesson titled "1. Software Distribution". The sidebar on the left lists topics: 1. Software Distribution (selected), 2. Package Repositories, 3. tar and gzip, 4. Package Dependencies, 5. rpm and dpkg, 6. yum and apt, and 7. Compile Source Code. The main content area contains a section titled "1. Software Distribution" with text explaining what packages are and how they are installed via package managers like apt or yum. To the right of the main content are two boxes: "Ceremonias" (which says "No exercises for this lesson") and "Examen" (which asks "Where is the sources file in a Debian system?"). A green button labeled "Revisar respuesta" is visible. The browser interface includes a search bar, a taskbar with various icons, and a status bar at the bottom.

This screenshot shows the same web browser window with the URL "linuxjourney.com/lesson/package-repositories". The active tab is now titled "linuxjourney.com/lesson/package-repositories". The sidebar still lists the same topics, but "2. Package Repositories" is selected. The main content area contains a section titled "2. Package Repositories" with text explaining what package repositories are and how they work. It mentions the /etc/apt/sources.list file. To the right are the "Ceremonias" and "Examen" boxes. The "Examen" box asks "Where is the sources file in a Debian system?" and has a green "Revisar respuesta" button. The browser interface remains consistent with the first screenshot.

**Ceremonias**

Familiarize yourself with the tar documentation and look at the other options available in the manpage.

**Examen**

What tar flag is  
c-x k  
cd ..  
cat  
clear  
cp  
c-w : cut

¡Respuesta correcta!

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

**Ceremonias**

No exercises for this lesson.

**Examen**

No questions, move along!

¡Respuesta correcta!

Revisar respuesta

Continuar

Activar Windows  
Ve a Configuración para activar Windows.

**Ceremonias**

Find a program that you want to install on your system like Google Chrome and install it using one of these commands.

**Examen**

What is the package management tool for .deb files?

dpkg

Activar Windows  
Ve a Configuración para activar Windows.

**Ceremonias**

Installing package dependencies. Two of the most popular management systems is **yum** and **apt**. Yum is exclusive to the Red Hat family and apt is exclusively to the Debian family.

**Examen**

What command is used to show package information on a Debian system?

apt show

Activar Windows  
Ve a Configuración para activar Windows.

Your Repositories (365) WhatsApp OverTheWire: Wargame Learning | Linux Journal Solve Linux Shell | Hackers Welcome to HackerRank

linuxjourney.com/lesson/compile-source-code

Packages

1. Software Distribution
2. Package Repositories
3. tar and gzip
4. Package Dependencies
5. rpm and dpkg
6. yum and apt
7. Compile Source Code

error and you'll need to fix those dependencies.

```
$ ./configure
```

The `J` allows you to execute a script in the current directory.

```
$ make
```

Inside of the package contents, there is a file called `Makefile` that contains rules to building the software. When you run the `make` command, it looks at this file to build the software.

```
$ sudo make install
```

This command actually installs the package. It will copy the correct files to the correct locations on your computer.

If you want to uninstall the package, use:

```
$ sudo make uninstall
```

Be wary when using `make install`, you may not realize how much is actually going on in the background. If you decide to remove this package, you may not actually remove everything because you didn't realize what was added to your system. Instead forget everything about `make install` that I just explained to you and use the `checkinstall` command. This command will make a `.deb` file for you that you can easily install and uninstall.

```
$ sudo checkinstall
```

This command will essentially "make install" and build a `.deb` package and install it. This makes it easier to remove the package later on.

## Ceremonias

Find a source code program (from a trusted site) and install from source.

## Examen

What should you use instead of `make install` ALWAYS?

✓ Respuesta correcta!

checkinstall Revisar respuesta

Activar Windows  
Ve a la Configuración para activar Windows.

Return

Buscar

19:04 6/6/2025

