

Visita nuestro sitio SoloLinux.es

MAGAZINE SOLO LINUX

Nº
22

Tu revista, la revista de tod@s

NOVIEMBRE 2020



Top 5: las supercomputadoras más potentes del 2020

Extraer archivos en **bash** con una función

Actualizar **Fedora 32** a **Fedora 33** sin problemas

Instalar **GIMP 2.99.2** en Ubuntu y derivados

Cómo instalar **Gravit Designer** en Linux

Probar el rendimiento del disco con **KDiskMark**

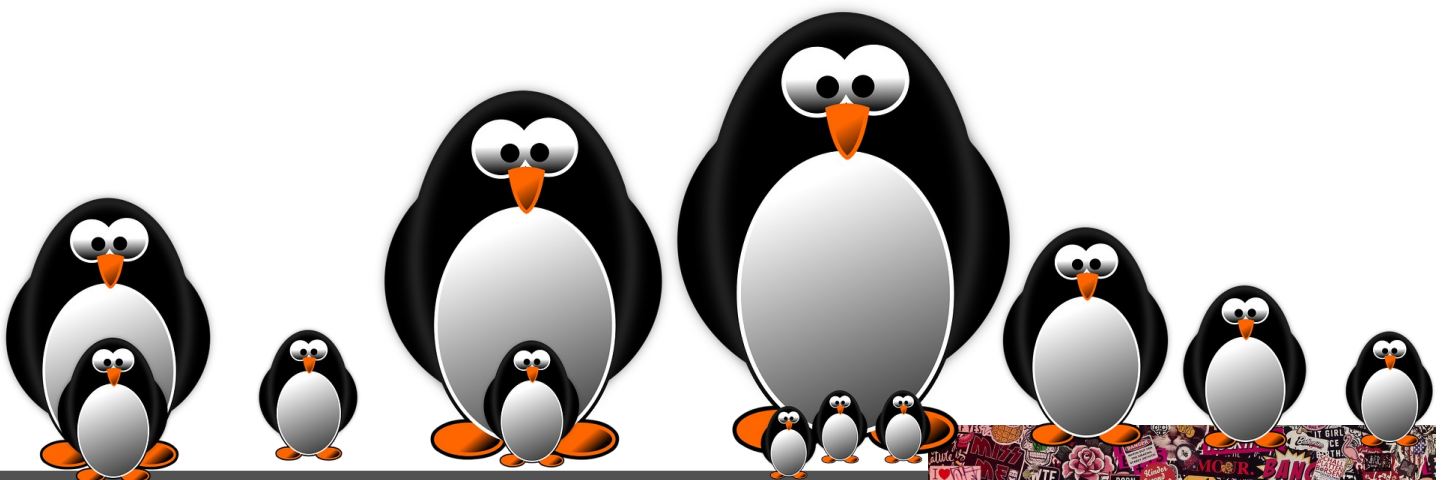
MANUALES, SCRIPTS, SOFTWARE, HARDWARE, DISTROS LINUX, SEGURIDAD, REDES Y MUCHO MAS EN LA WEB...

SoloLinux

Esta revista es de **distribución gratuita**, si lo consideras oportuno puedes ponerle precio.

Tu también puedes ayudar, contamos con la posibilidad de hacer donaciones para la REVISTA, de manera muy simple a través de **PAYPAL**

AYUDANOS A SEGUIR CRECIENDO

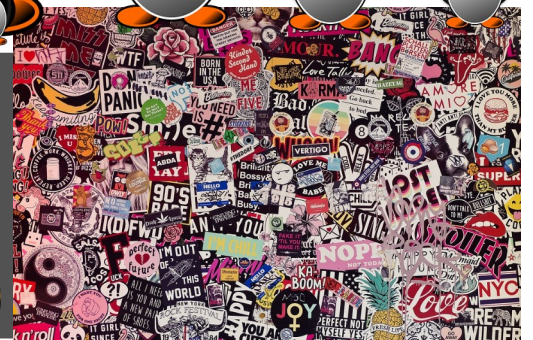


Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es



By: MichaelGaida. Extraída de Pixabay.

Bienvenido a la Revista SOLOLINUX

Buenos días, tardes o noches, dependiendo del lugar del mundo donde se encuentren ahora mismo.

Os presentamos el **número 22 de la Revista SoloLinux**. Quiero agradecer a todos los que hacéis que este proyecto siga adelante. Desde el equipo de SOLOLINUX esperamos que os guste este nuevo número.

Al igual que en número anteriores quiero proponer a nuestros lectores que nos pasen por email su opinión personal sobre **SOFTWARE LIBRE y GNU/LINUX**. Todos los emails recibidos saldrán en la revista en un apartado llamado **LA OPINIÓN DEL LECTOR**. También pueden lanzar preguntas para el resto de los lectores para saber diferentes puntos de vista de distintos temas. Anímense y **envíen sus opiniones**.

Al igual que lo anteriormente mencionado, nos gustaría promover un espacio en la revista sobre los **eventos de Software Libre y GNU/Linux en todo el mundo**. Los organizadores de estos eventos pueden ponerse en contacto con migo a través de correo electrónico, adrian@sololinux.es

Sin mas **quiero agradecer a todos** los que hacéis posible que esta revista siga adelante.

Personalmente agradezco a Sergio todo su trabajo en la multitud de artículos que realiza a lo largo del mes para que esta revista pueda tener suficiente información mes a mes.

Gracias a TOD@S

Compartan esta revista en sus redes sociales o web.
Revista digital **SOLOLINUX MAGAZINE**.

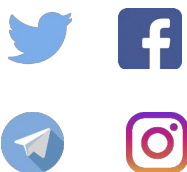
**Tu revista, la revista
de todos.**

Esta revista es de **distribución gratuita**, si lo consideras oportuno puedes ponerle precio.
Tu también puedes ayudar, contamos con la posibilidad de hacer donaciones para la REVISTA, de manera muy simple a través de **PAYPAL**

**AYUDANOS A SEGUIR
CRECIENDO**



Síguenos en las Redes:



La revista SOLOLINUX
esta realizada con Libre
Office Impress 7.0.0.3

Nuestras Webs:

www.sololinux.es
www.solowordpress.es

EDITORIAL

Edición:

- **Adrián A. A.**
adrian@sololinux.es

Redacción:

- **Sergio G. B.**
(Administrador y redactor artículos
SoloLinux)
info@sololinux.es

- **Henry G. R.**
(Redactor artículos SoloWordPress)
info@solowordpress.es

Agradecimientos:

- **Hernán** por dedicarnos un poco
de su tiempo para la realización
de la entrevista.

Diseños de Portada:

- **Karina Fernández**
@karyfernandez.design

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital de software libre y GNU/Linux en ESPAÑOL

CON SOLOLINUX MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a: adrian@sololinux.es

Colabora:

Quieres colaborar en la revista.
Para mayor información escribe un email a: adrian@sololinux.es

La **Revista SOLOLINUX**, se distribuye gratuitamente en forma digital para todo el mundo que quiere disfrutar de ella. Si quieres imprimirla es cosa tuya.
Si os cobran por ella, os están timando. :)

Contacto:

Para cualquier consulta sobre las revistas, publicidad o colaboraciones escribir un email a:
adrian@sololinux.es

NOTICIAS

- 08. Linux Mint agrega Chromium a sus repositorios oficiales
- 09. Top 5 – Las supercomputadoras más potentes del 2020

MANUALES

- 11. Eliminar módulos del kernel con el comando Rmmod
- 12. Extraer archivos en bash con una función
- 13. 20 ejemplos de uso del comando nmap en linux
- 16. Actualizar Fedora 32 a Fedora 33 sin problemas
- 17. Listar los procesos que se ejecutan en linux
- 26. Actualizar Transmission en Ubuntu 20.04 y derivados
- 28. Como crear archivos torrent con Transmission
- 31. 8 formas de verificar las conexiones ssh activas
- 38. Como instalar Nextcloud en Ubuntu 20.04
- 40. Instalar Nextcloud Client en Ubuntu 20.04
- 41. Conectar por SSH incluyendo el password
- 45. 4 formas de buscar la ip de un dominio en terminal
- 46. Como usar el comando Hexdump en linux
- 48. Como usar zip en linux
- 50. Configurar OPCache en Ubuntu 20.04
- 51. Extraer archivos zip con unzip
- 52. Instalar php 8.0 en Ubuntu 20.04
- 56. Ajustar la frecuencia de la cpu con CpuPower-GUI



HARDWARE

- 21. Equilibrar la carga entre la memoria ram y swap
- 25. Probar el rendimiento del disco con KdiskMark
- 34. Configurar el Bluetooth en Linux



SOFTWARE

- 22. Novedad – Instalar el editor Bluefish 2.2.12 en Ubuntu
- 23. Instalar GIMP 2.99.2 en Ubuntu y derivados
- 24. Cómo instalar Gravit Designer en Linux

SEGURIDAD

- 36. Antivirus para linux – ClamAV y ClamTK
- 42. El nuevo Kali Linux 2020.4 se pasa a ZSH



REDES

- 44. HTTPie – El cliente http en linea de comandos

DISTROS LINUX

- 53. Los mejores derivados de Arch Linux del 2021



ENTREVISTAS

- 58. Entrevista a Hernán Administrador de su Blog personal hernanalbornoz.wordpress.com



**INSTITUTO
LINUX**



CURSO Y CERTIFICACIÓN

€115
CADA UNO

LPI LINUX ESSENTIALS

**PREPARA TU PRIMERA
CERTIFICACIÓN INTERNACIONAL LINUX**

- ACCESO INMEDIATO -

Tutorías de Fabián Ampalio



+54 9 11 6969 9993



@fabianampalio

Instalar php 8.0 en Ubuntu 20.04

Sergio G.B. | 27/11/2020

MANUALES



Instalar php 8.0 en Ubuntu 20.04, Ubuntu 18.04 y derivados. PHP 8.0 es la nueva versión principal del lenguaje de programación más extendido, si hablamos de sitios web. Nos sorprende con un montón de mejoras y nuevas características, que puedes ...

ARTICULO COMPLETO >

MÁS VISTOS HISTÓRICO

Las mejores alternativas a Rufus en Linux

05/09/2019

Las mejores distribuciones linux del 2020

21/01/2020

Formulario de contacto sencillo con HTML5, CSS y PHP

20/01/2015

Ver la TV online con listas IPTV m3u - SMPlayer

25/04/2020



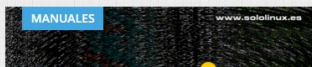
Buscar...



ÚLTIMOS ARTICULOS

- Instalar php 8.0 en Ubuntu 20.04
- Extraer archivos zip con unzip
- Configurar OPcache en Ubuntu 20.04
- Como usar zip en linux
- Como usar el comando Hexdump en linux
- 4 formas de buscar la ip de un dominio en terminal
- HTTPIe - El cliente http en linea de comandos
- El nuevo Kali Linux 2020.4 se pasa a ZSH

MANUALES



NOTICIAS

Top 5 - Las supercomputadoras más

SoloWordpress

#UsaMascarilla
#LávateLasManos

¡Bienvenido a WordPress!

Hemos recopilado algunos enlaces para que puedas comenzar

Inicio Manuales Noticias Temas Plugins SEO Seguridad Contacto

Novedades



Recomendado



Portal Mensajes nuevos Mensajes de hoy Búsqueda Lista de miembros

Iniciar sesión Registrarse

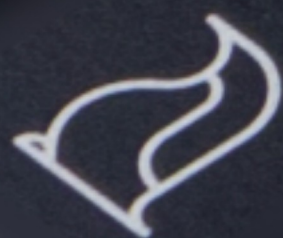
¡Hola, Invitado!

ADMINISTRACIÓN FOROLINUX				
Foro				Último mensaje
	Anuncios y Novedades de FOROLINUX	0 Temas 0 Mensajes		Nunca
	Anuncios, novedades y avisos sobre temas importantes que se estén dando en FOROLINUX			
	Ayúdanos a mejorar	0 Temas 0 Mensajes		Nunca
	Toda ayuda para mejorar el foro y a su comunidad será bien recibida.			
	Bugs y errores en el Foro	0 Temas 0 Mensajes		Nunca
	Has encontrado un error en el funcionamiento del FORO. Repórtalo aquí.			
GENERAL FOROLINUX				
Foro				Último mensaje
	Nuevo en FOROLINUX (Presentate)	0 Temas 0 Mensajes		Nunca
	Puedes presentarte aquí, preséntate para que todos podamos conocerte.			
	Nuevo en Linux	0 Temas 0 Mensajes		Nunca
	Eres nuevo en este Sistema Operativo, cuéntanos tus dudas y podremos orientarte, para que le vayas cogiendo el gusto a esto de GNU/LINUX			
	Noticias y Actualidad	0 Temas 0 Mensajes		Nunca
	Noticias y actualidad sobre GNU/LINUX y SOFTWARE LIBRE.			
	Documentación	0 Temas 0 Mensajes		Nunca
	Foro dedicado para documentación libre. Cualquier documento con derechos de autor será eliminado. Solo se permite			

VANT

S O M O S L I N U X E R O S

www.vantpc.es



la gama más completa de ordenadores linuxeros

 [@vantpc](https://twitter.com/vantpc)  [vant.pc](https://facebook.com/vant.pc)  [vantpc_es](https://instagram.com/vantpc_es)  t.me/vantpc

Linux Mint agrega Chromium a sus repositorios oficiales

Clement Lefebvre anuncio lo que era evidente, los usuarios de **Linux Mint y LMDE** (Linux Mint Debian Edition), al final podrán instalar el navegador web Chromium desde los repositorios de software oficiales, e instalar su nuevo reproductor IPTV.

Personalmente, estoy de acuerdo en la decisión de Linux Mint y otras **distribuciones linux**, en no admitir los **paquetes snap**. Pero lo inadmisibles es, no ofrecer una alternativa real y dificultar a los usuarios la instalación de uno de los navegadores web más apreciados por la **comunidad linux**, hablamos de **Chromium**.



Desde **Linux Mint 20 Ulyana** en adelante, los desarrolladores de Linux Mint empaquetarán Chromium para distribuirlo en sus repositorios oficiales. Lamentablemente... creo que esta decisión llega un poco tarde, pues sé a ciencia cierta que muchos usuarios optaron por continuar en **Linux Mint 19**, o simplemente por buscar otra alternativa.

Linux Mint agrega Chromium a sus repositorios oficiales

Chromium

Debian todavía mantiene una compilación nativa de Chromium en sus repositorios, pero como es habitual en ellos no siempre es la última. En este punto, Linux Mint no fue por la vía fácil de tomarlo de Debian, compila su propio paquete desde el código fuente. Esto es un punto a favor de Mint, bravo.

La instalación de Chromium es bastante simple, desde su gestor de paquetes, desde el administrador Synaptic, o simplemente ejecutando el siguiente comando.

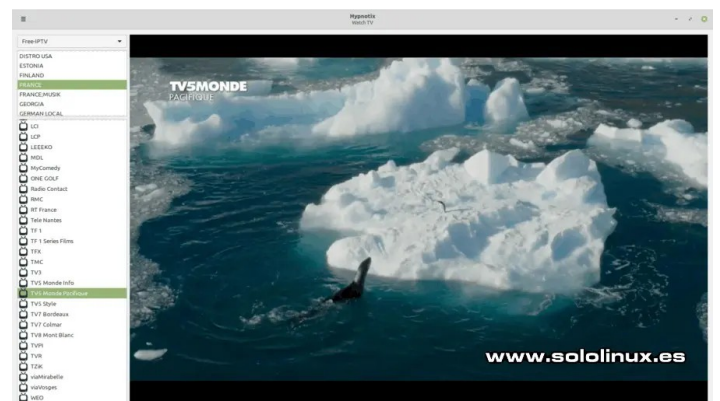
```
sudo apt install chromium
```

Así de fácil, se acabaron las complicaciones. Linux Mint agrega Chromium a sus repositorios oficiales.

Hypnotix IPTV

A petición de sus usuarios, Linux Mint desarrolla el **reproductor IPTV Hypnotix**. Pero debes tener presente que su última versión aún no conviene usar en producción, puede contener pequeños errores, pero se le augura un futuro prometedor.

El **reproductor de IPTV** viene preconfigurado con FreeIPTV. FreeIPTV es un proveedor de contenido gratuito que transmite una gran variedad de estaciones de TV, distribuidas por países. Como la utilidad aún está en desarrollo, de momento solo dispone del idioma inglés.



Puedes seguir su desarrollo en el **Github oficial**, o descargar la última versión lista para pruebas.

- [Descargar Hypnotix 1.0](#)

Top 5 – Las supercomputadoras más potentes del 2020

Las supercomputadoras son **sistemas** cuyo diseño de hardware y tecnológico ofrecen un alto rendimiento. Los usos de este tipo de máquinas son variados, desde administrar **bases de datos** gigantescas, a soportar grandes cantidades de computación inimaginables hace apenas unos años.

Desde 1993, el **proyecto Top500** analiza todas las supercomputadoras del mundo, y crea un ranking con las 500 más potentes. Esta lista es revisada cada seis meses. Es conocida como Top500. Es evidente que nosotros no vamos a listar las 500 más potentes, pero si las 5 supercomputadoras más potentes del mundo, en este año que está a punto de acabar.

Top 5 – Las supercomputadoras más potentes del 2020

Antes de comenzar, es interesante saber que la empresa **Hewlett Packard Enterprise (HPE)**, se encuentra en proceso de fabricación (en Finlandia) de una supercomputadora llamada **LUMI**. Se espera que esta supermaquina ofrezca un rendimiento máximo de 550 petaflops o más. **LUMI** será la próxima número uno, pero eso será en un futuro. De momento vemos las existentes en la actualidad.

Fugaku



La **supercomputadora Fugaku** funciona con el SoC A64FX de 48 núcleos de **Fujitsu**. Lo podemos encontrar en el Centro RIKEN de Ciencias Computacionales (R-CCS) en Kobe, Japón. Este supersistema alcanza los 415,5 petaflops. Destacamos que es la primera maquina en alcanzar el número uno, montado ARM.

IBM's Summit



La **Summit** fabricada por **IBM**, también se encuentra entre las **supercomputadoras más potentes del mundo**. Utiliza dos CPU Power9 de 22 núcleos y seis GPU NVIDIA Tesla V100. Esta supermaquina se encuentra en el Laboratorio Nacional Oak Ridge de Tennessee. Con un máximo de 148,8 petaflops, es la más poderosa de Estados Unidos.

The Sierra system



El **Laboratorio Lawrence Livermore** de California (Estados Unidos), también cuenta con una de las supercomputadoras más potentes del mundo. Monta dos **CPU Power9** y cuatro **GPU NVIDIA Tesla V100** en cada uno de sus 4320 nodos. Tiene un potencial de 94,6 petaflops.

Sunway TaihuLight



Con un índice de 93 petaflops, esta supercomputadora situada en el Centro Nacional de Supercomputación de China (Wuxi, Jiangsu), se dedica a la prospección de petróleo, ciencias de la vida, el tiempo, diseño industrial e investigación de nuevos fármacos. Tiene la particularidad de ser energéticamente eficiente.

Tianhe-2A



Esta supercomputadora tiene una arquitectura híbrida, monta CPU's Intel Xeon y coprocesadores Matrix-2000 personalizados. La **Tianhe-2A** también es conocida como Milky Way-2A y, alcanza un máximo de 61,4 petaflops. La podemos encontrar en el Centro Nacional de Supercomputadoras de Guangzhou.

¿Qué son los petaflops?

PetaFLOPS es el término usado en informática para medir la potencia de cálculo de las CPU y GPU. Composición de PetaFLOPS:

- **Peta** indica un factor de 1015 (mil billones).
- **FLOPS** (Floating Point Operations Per Second) las operaciones de coma flotante por segundo.

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es



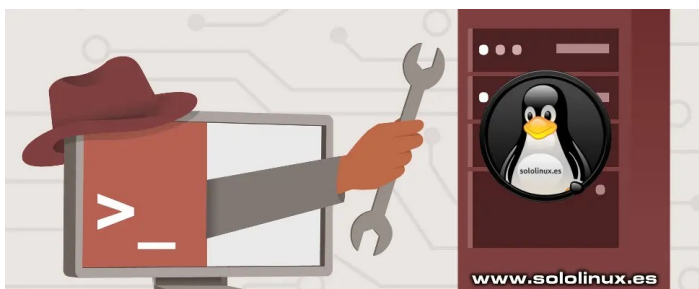
Eliminar módulos del kernel con el Comando Rmmod

```
snd_seq_device 16384 3 snd_seq,snd_seq_midi,snd_rawmidi
snd_timer 36864 2 snd_seq,snd_pcm
cfg80211 712704 3 8723de,mac80211,rtw88
lpc_ich 24576 0
snd 86016 17 snd_hda_codec_generic,snd_seq,snd_seq_device
snd_hda_codec_realtek,snd_timer,snd_pcm,snd_rawmidi
libarc4 135 4 1 mac80211
mei_tx 24576 0
mei 102400 3 mei_hdcp,mei_tx
intel_xhci_usb_role_switch 16384 0
roles 135 1 intel_xhci_usb_role_switch
soundcore 16384 1 snd
processor_thermal_device 24576 0
intel_rapl_common 24576 2 intel_rapl_common,processor_thermal_device
intel_sdt 24576 1 processor_thermal_device
kxcjk_1013 28672 0
industrialio_triggered_buffer 16384 1 kxcjk_1013
kfifo_buf 135 1 industrialio_triggered_buffer,kxcjk_1013
int3400_thermal 73728 3 industrialio_triggered_buffer,kfifo_buf,kxcjk_1013
industrialio 73728 1 industrialio_triggered_buffer,kfifo_buf,kxcjk_1013
mac_hid 16384 0
int3403_thermal 20480 0
acpi_thermal_rel 16384 1 int3400_thermal
int340x_thermal_zone 16384 2 int3403_thermal,processor_thermal_device
```

En los **sistemas operativos Linux**, el **kernel** es el encargado de administrar los recursos del sistema, y ejercer de enlace entre el hardware y el software de la máquina.

El kernel de Linux es una aplicación diseñada en sistema modular. Los módulos del kernel, también llamados controladores, son pequeños fragmentos de código que tienen la capacidad de ampliar las funciones del kernel. Los módulos pueden compilarse como módulos cargables o integrarse directamente en el **kernel**. Los módulos cargables se cargan y descargan dinámicamente en el kernel sin tener que reiniciar el sistema.

En este artículo, veremos como **eliminar módulos del Kernel Linux** que ya no utilizas, con el **comando Rmmod**.



Eliminar módulos del kernel con el comando Rmmod

La herramienta **rmmod** forma parte del paquete de **binarios kmod**, que se usa para administrar y controlar los módulos del **kernel linux**. Debes recordar que para utilizar estas herramientas, debes ser usuario **root** o contar con los suficientes privilegios administrativos.

La sintaxis de **rmmod** es sencilla.

```
rmmod [OPTIONS] MODULE_NAME
```

Existen algunas opciones, las vemos.

- **-v / -verbose** : Imprime mensajes sobre las operaciones.

- **-f -force** : Esta opción es extremadamente peligrosa, pues nos permite eliminar módulos en uso, y que no se permite su eliminación o están marcados como inseguros.
- **-w -wait** : Si el módulo está en uso, lo aísla y espera ejecutar la orden cuando este liberado.
- **-s -syslog** : Envía los errores a syslog.
- **-V -version** : Imprime la versión de la herramienta.

Puedes listar los módulos cargados en el kernel con el comando **lsmod**.

```
lsmod
```

```
[root@solo_linux ~]# lsmod
Module                Size  Used by
tcp_diag              12591  0
udp_diag              12801  0
inet_diag             18949  2 tcp_diag,udp_diag
xt_multiport          12798  5
binfmt_misc           17468  1
ip6t_rpfilter         12595  1
ip6t_REJECT           12625  3
nf_reject_ipv6        13717  1 ip6t_REJECT
ipt_REJECT            12541  303
nf_reject_ipv4        13373  1 ipt_REJECT
xt_conntrack          12760  52
ebtable_nat            12807  1
ebtable_broute        12731  1
bridge                151336 1 ebtable_broute
stp                   12976  1 bridge
llc                   14552  2 stp,bridge
ip6table_nat          12864  1
nf_conntrack_ipv6     18935  27
nf_defrag_ipv6        35104  1 nf_conntrack_ipv6
```

Ejemplos de uso del comando **Rmmod**.

```
rmmod tcp_diag
rmmod tcp_diag udp_diag
rmmod -w bridge
```

¡IMPORTANTE!!!

Los módulos que hemos descargado (eliminado) del kernel, volverán a cargar automáticamente al reiniciar el sistema. Si queremos que no se carguen con el sistema, pero tener la posibilidad de volver a usarlos más adelante... creamos una archivo de configuración en **/etc/modprobe.d**.

```
nano /etc/modprobe.d/blacklist.conf
```

Insertamos los modulo que no queremos que carguen con el sistema, uno por línea y con la sintaxis que vemos en el siguiente ejemplo.

```
blacklist tcp_diag
blacklist udp_diag
blacklist inet_diag
blacklist xt_multiport
blacklist binfmt_misc
blacklist ip6t_rpfilter
blacklist ip6t_REJECT
blacklist nf_reject_ipv6
blacklist ipt_REJECT
blacklist nf_reject_ipv4
blacklist xt_conntrack
blacklist ebtable_nat
blacklist ebtable_broute
```

Guardas el archivo, sales del editor y reinicias el sistema.

Extraer archivos en bash Con una función

Los **usuarios de linux**, disponemos de muchas aplicaciones con la capacidad de extraer **archivos comprimidos**, en diversos formatos. A veces tantos que nos puede resultar difícil recordar todos los comandos. Hoy solucionamos este problema.

Existe una función en bash, que tiene la capacidad de tomar los ejecutables de los comandos o herramientas de compresión que tengamos instaladas en nuestro linux, e iniciar y ejecutar las aplicaciones en línea de comandos sin ninguna intervención del usuario. Hablamos de extracciones.

A partir de hoy, extraer tus archivos en los formatos más habituales como: Tar.bz2, .tar.bz, .bz2, .rar, .zip y .7z, etc. No supondrá ningún problema, ni será necesario utilizar la aplicación específica de la extensión de archivo para extraer lo que necesites. Se acabaron los problemas. Extraer archivos en bash con una función.

```
# You may uncomment the following lines if you want 'ls' to be colorized:
# export LS_OPTIONS='--color=auto'
# eval "$(dircolors)"
# alias ls='ls $LS_OPTIONS'
# alias ll='ls $LS_OPTIONS -l'
# alias l='ls $LS_OPTIONS -lA'
#
# Some more alias to avoid making mistakes:
# alias rm='rm -i'
# alias cp='cp -i'
# alias mv='mv -i'
#
# Bash Function To Extract File Archives Of Various Types
extract () {
  if [ -f $1 ] ; then
    case $1 in
      *.tar.bz2) tar xjf $1 ;;
      *.tar.gz) tar xzf $1 ;;
      *.bz2) bunzip2 $1 ;;
      *.rar) rar x $1 ;;
      *.gz) gunzip $1 ;;
      *.tar) tar xf $1 ;;
      *.tbz2) tar xjf $1 ;;
      *.tgz) tar xzf $1 ;;
      *.zip) unzip $1 ;;
      *.Z) uncompress $1 ;;
      *.7z) 7z x $1 ;;
      *) echo "'$1' cannot be extracted"
    esac
  else
    echo "'$1' is not a valid file"
  fi
}
```

www.sololinux.es

Extraer archivos en bash con una función

Editamos el archivo lanzador de bash (si no tienes permisos lo ejecutas con sudo).

```
nano ~/.bashrc
```

Si tienes ubuntu o debian veremos algo similar a...

```
# ~/.bashrc: executed by bash(1) for non-login shells.
# Note: PS1 and umask are already set in /etc/profile. You
# should not
# need this unless you want different defaults for root.
# PS1='${debian_chroot:+($debian_chroot)}\h:\w$ '
# umask 022
# You may uncomment the following lines if you want 'ls'
# to be colorized:
# export LS_OPTIONS='--color=auto'
# eval "$(dircolors)"
# alias ls='ls $LS_OPTIONS'
# alias ll='ls $LS_OPTIONS -l'
# alias l='ls $LS_OPTIONS -lA'
#
# Some more alias to avoid making mistakes:
# alias rm='rm -i'
# alias cp='cp -i'
# alias mv='mv -i'
```

En CentOS...

```
# .bashrc
# User specific aliases and functions
alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'
# Source global definitions
if [ -f /etc/bashrc ]; then
  . /etc/bashrc
fi
```

Al final del archivo agrega las extensiones y llamadas que necesites, por ejemplo...

```
# Bash Function To Extract File Archives Of Various Types
extract () {
  if [ -f $1 ] ; then
    case $1 in
      *.tar.bz2) tar xjf $1 ;;
      *.tar.gz) tar xzf $1 ;;
      *.bz2) bunzip2 $1 ;;
      *.rar) rar x $1 ;;
      *.gz) gunzip $1 ;;
      *.tar) tar xf $1 ;;
      *.tbz2) tar xjf $1 ;;
      *.tgz) tar xzf $1 ;;
      *.zip) unzip $1 ;;
      *.Z) uncompress $1 ;;
      *.7z) 7z x $1 ;;
      *) echo "'$1' cannot be extracted"
    esac
  else
    echo "'$1' is not a valid file"
  fi
}
```

Guarda el archivo y cierra el editor. Recargamos el servicio.

```
source ~/.bashrc
```

Ahora mismo solo tienes que recordar el comando «**extract**». Tan solo debes ejecutar la siguiente sintaxis.

```
extract archivo.tar
```

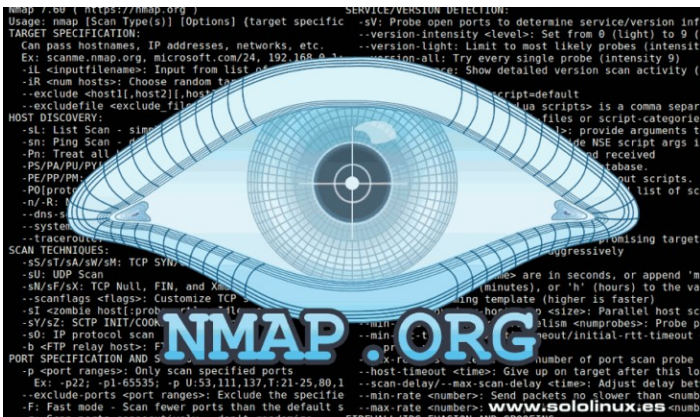
Por ejemplo...

```
extract demo.zip
```

```
root@sololinux-demo:~# extract demo.zip
Archive: demo.zip
  creating: demo/
  extracting: demo/demo1.txt
  extracting: demo/demo2.txt
  extracting: demo/demo3.odt
  extracting: demo/demo4.jpg
  extracting: demo/demo5.png
```

En nuestro ejemplo veremos los archivos descomprimidos en la carpeta demo, se acabaron los problemas. Extraer archivos en bash con una función.

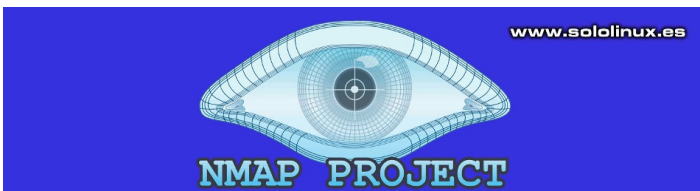
20 ejemplos de uso del Comando nmap en linux



Nmap (Network Mapper), es una herramienta que nos aporta importante información sobre redes. Es de código abierto y tiene la capacidad de revelar los hosts y, recopilar información detallada sobre ellos.

Escrito en C, C++ y Python, esta fabulosa herramienta fue lanzada por primera vez en septiembre de 1997. En la actualidad, **nmap**, es indispensable para cualquier **sysadmin**, forense digital o usuario aficionado. Útil para descubrir potenciales vulnerabilidades, revela información sobre los hosts activos de una red, sus puertos abiertos, servicios y sistemas operativos, además tiene la capacidad de lanzar escaneos silenciosos.

En este artículo, destacamos los 20 comandos más útiles de la herramienta que nos ayudaran a recopilar diversa, e importante información de sistemas host.



20 ejemplos de uso del comando nmap en linux

Instalar Nmap

Nmap no suele venir instalada por defecto, por tanto procedemos a ello.

Debian, Ununtu, Linux Mint y derivados:

```
sudo apt -y install nmap
```

CentOS 7, RHEL 7 y derivados:

```
sudo yum -y install nmap
```

Fedora, CentOS 8, RHEL 8 y derivados:

```
sudo dnf -y install nmap
```

Arch Linux, Manjaro y derivados:

```
sudo pacman -S nmap
```

20 ejemplos de uso del comando nmap

Escanear un host

Puedes insertar una ip o un dominio.

```
Nmap 192.168.254.231
nmap tudominio.com
```

Ejemplo...

```
sergio@sololinux:~$ nmap midominio.es
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 18:17 EET
Nmap scan report for midominio.es (161.112.197.97)
Host is up (0.054s latency).
rDNS record for 163.172.107.87: host.adminserver.es
Not shown: 930 filtered ports, 57 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
106/tcp   open  pop3pw
110/tcp   open  pop3
143/tcp   open  imap
443/tcp   open  https
465/tcp   open  smtps
993/tcp   open  imaps
995/tcp   open  pop3s
8443/tcp  open  https-alt
Nmap done: 1 IP address (1 host up) scanned in 5.05 seconds
```

Escanear varias ip o un rango

```
nmap 192.168.0.1 192.168.0.10
nmap 192.168.0.1-100
```

Escanear una subnet completa o definida

```
nmap 192.168.0.*
nmap 192.168.0.0/24
```

Información detallada de una ip o dominio

```
nmap -v 192.168.0.1
```

```
sergio@sololinux:~$ nmap -v 192.168.0.1
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 18:24 EET
Initiating Ping Scan at 18:24
Scanning 192.168.0.1 [2 ports]
Completed Ping Scan at 18:24, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 18:24
Completed Parallel DNS resolution of 1 host. at 18:24, 0.01s elapsed
Initiating Connect Scan at 18:24
Scanning _gateway (192.168.0.1) [1000 ports]
Discovered open port 80/tcp on 192.168.0.1
Discovered open port 1900/tcp on 192.168.0.1
Completed Connect Scan at 18:24, 0.26s elapsed (1000 total ports)
Nmap scan report for _gateway (192.168.0.1)
Host is up (0.0085s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
80/tcp    open  http
1900/tcp  open  upnp
Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds
```

Excluir ip del escaneo

```
nmap 192.168.0.0/24 --exclude 192.168.0.65
```

También puedes excluir las ip de un archivo de texto definido.

```
nmap 192.168.0.* --excludefile ip-excluidas.txt
```

20 ejemplos de uso del comando nmap en linux

Escaneo rápido

```
nmap -F 192.168.0.87
```

Detectar host activos de una red

```
nmap -sn 192.168.0.0/24
```

Analizar host desde un archivo

```
nmap iL list-hosts.txt
```

Detectar si el firewall está habilitado y funcionando

```
nmap -sA 192.168.0.79
```

Identificar el sistema operativo instalado en una ip

```
nmap -O 192.168.0.48
```

Escanear un puerto específico (en ejemplo el 80)

```
nmap -p 80 192.168.0.49
```

También puedes escanear varios puertos si los separas con una coma.

```
nmap -p 80,443 192.168.0.1.49
```

Incluso definir un rango de puertos.

```
nmap -p 80-443 192.168.0.49
```

Ejemplo...

```
sergio@sololinux:~$ nmap -p 80-443 192.168.0.73
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 18:41 EET
Nmap scan report for hostdemo.com (192.168.0.73)
Host is up (0.099s latency).
Not shown: 359 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
110/tcp   open  pop3
143/tcp   open  imap
232/tcp   open  unknown
443/tcp   open  https
Nmap done: 1 IP address (1 host up) scanned in 6.26 seconds
```

```
sergio@sololinux:~$ nmap -sT 62.210.131.18
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 18:46 EET
Nmap scan report for 62-210-131-18.rev.poneytelecom.eu (62.210.131.18)
Host is up (0.048s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
80/tcp    open  http
135/tcp    filtered msrpc
139/tcp    filtered netbios-ssn
445/tcp    filtered microsoft-ds
7999/tcp   open  irdmi2
Nmap done: 1 IP address (1 host up) scanned in 4.85 seconds
```

Escanear puertos TCP/UDP

Escanear puertos TCP con el comando nmap.

```
nmap -sT 192.168.0.14
```

Escanear un puerto TCP definido.

```
nmap -p T:80 192.168.0.87
```

Escanear puertos UDP con el comando nmap.

```
nmap -sU 192.168.0.28
```

Escanear un puerto UDP definido.

```
nmap -p U:79 192.168.0.31
```

Detectar los servicios y versiones que se ejecutan

```
nmap -sV 192.168.0.56
```

```
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 18:53 EET
Nmap scan report for host.adminserver.es (192.168.0.56)
Host is up (0.049s latency).
Not shown: 930 filtered ports, 57 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          ProFTPD
22/tcp    open  ssh          OpenSSH 7.4 (protocol 2.0)
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND none
80/tcp    open  http         nginx
106/tcp   open  tcpwrapped
110/tcp   open  pop3         Dovecot pop3d
143/tcp   open  imap         Dovecot imapd
443/tcp   open  ssl/http     nginx
465/tcp   open  ssl/smtp     Postfix smtpd
993/tcp   open  ssl/imap     Dovecot imapd
995/tcp   open  ssl/pop3     Dovecot pop3d
8443/tcp  open  ssl/https-alt sw-cp-server
```

Escaneo silencioso

Los escaneos de nmap, dejan un rastro visible en los registros del servidor. Puedes evitar esto con el siguiente comando.

```
nmap -sS 192.168.0.71
```

Averiguar los protocolos admitidos en el sistema remoto

```
sudo nmap -s0 192.168.0.48
```


Escaneo agresivo

Ofrece muchos detalles omitidos con otras opciones.

```
nmap -A 192.168.0.33
```

Guardar resultados en un archivo

```
nmap 192.168.0.11 > resultado.txt
```

Puedes imprimir el archivo con el siguiente comando:

```
cat resultado.txt
```

Localizar las interfaces y las rutas

```
nmap --iflist
```

Salida de ejemplo...

```
sergio@sololinux:~$ nmap --iflist
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 19:05 EET
*****INTERFACES*****
DEV (SHORT) IP/MASK TYPE UP MTU MAC
lo (lo) 127.0.0.1/8 loopback up 65536
lo (lo) ::1/128 loopback up 65536
wlo1 (wlo1) 192.168.0.104/24 ethernet up 1500 70:66:55:01:02:4D
wlo1 (wlo1) 192.168.100.25/24 ethernet up 1500 70:66:55:01:02:4D
wlo1 (wlo1) fe80::e8f4:fa6f:11f1:74f8/64 ethernet up 1500 70:66:55:01:02:4D
eno1 (eno1) (none)/0 ethernet up 1500 38:22:E2:BA:27:73
*****ROUTES*****
DST/MASK DEV METRIC GATEWAY
192.168.0.0/24 wlo1 600
192.168.100.0/24 wlo1 600
169.254.0.0/16 wlo1 1000
0.0.0.0/0 wlo1 600 192.168.0.1
::1/128 lo 0
fe80::e8f4:fa6f:11f1:74f8/128 wlo1 0
::1/128 lo 256
fe80::/64 wlo1 256
fe80::/64 wlo1 600
ff00::/8 wlo1 256
```

Ayuda del comando nmap

```
nmap -h
```

Versión instalada del comando nmap

```
nmap -v
```

Ejemplo de salida...

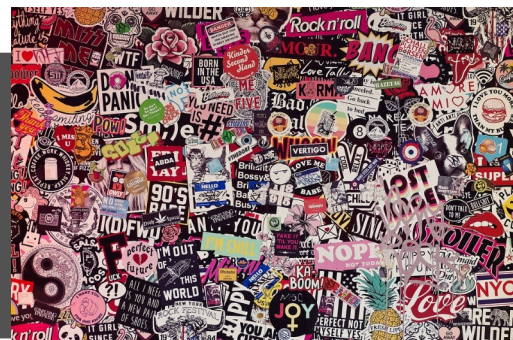
```
sergio@sololinux:~$ nmap --iflist
sergio@sololinux:~$ nmap -v
Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-05 19:07 EET
Read data files from: /usr/bin/./share/nmap
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.12 seconds
```

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL.

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es



Actualizar Fedora 32 a Fedora 33 sin problemas

Hace apenas una semana que **Fedora** lanzo su última versión, hablamos de **Fedora 33**. **Fedora Workstation** se centra en un escritorio orientado a los desarrolladores de software, que buscan un sistema operativo Linux que sea funcional y, que se adapte a las necesidades de su trabajo diario.

Si eres usuario de **Fedora**, seguro que ya sabes que la última versión ya está disponible. Como norma general siempre recomiendo hacer una instalación limpia, pero entiendo que no siempre es posible. Para aquellos usuarios que prefieren actualizar la versión, deben saber que también es posible.



Actualizar Fedora 32 a Fedora 33, no es difícil, pero puede tardar un tiempo, dependiendo de la velocidad de tu red y del hardware. La forma que nosotros proponemos es en **línea de comandos**, más segura, más eficaz.

Actualizar Fedora 32 a Fedora 33 sin problemas

Lo primero que debemos hacer es actualizar el sistema.

```
sudo dnf upgrade --refresh
```

Instalamos el complemento actualizador a Fedora 33.

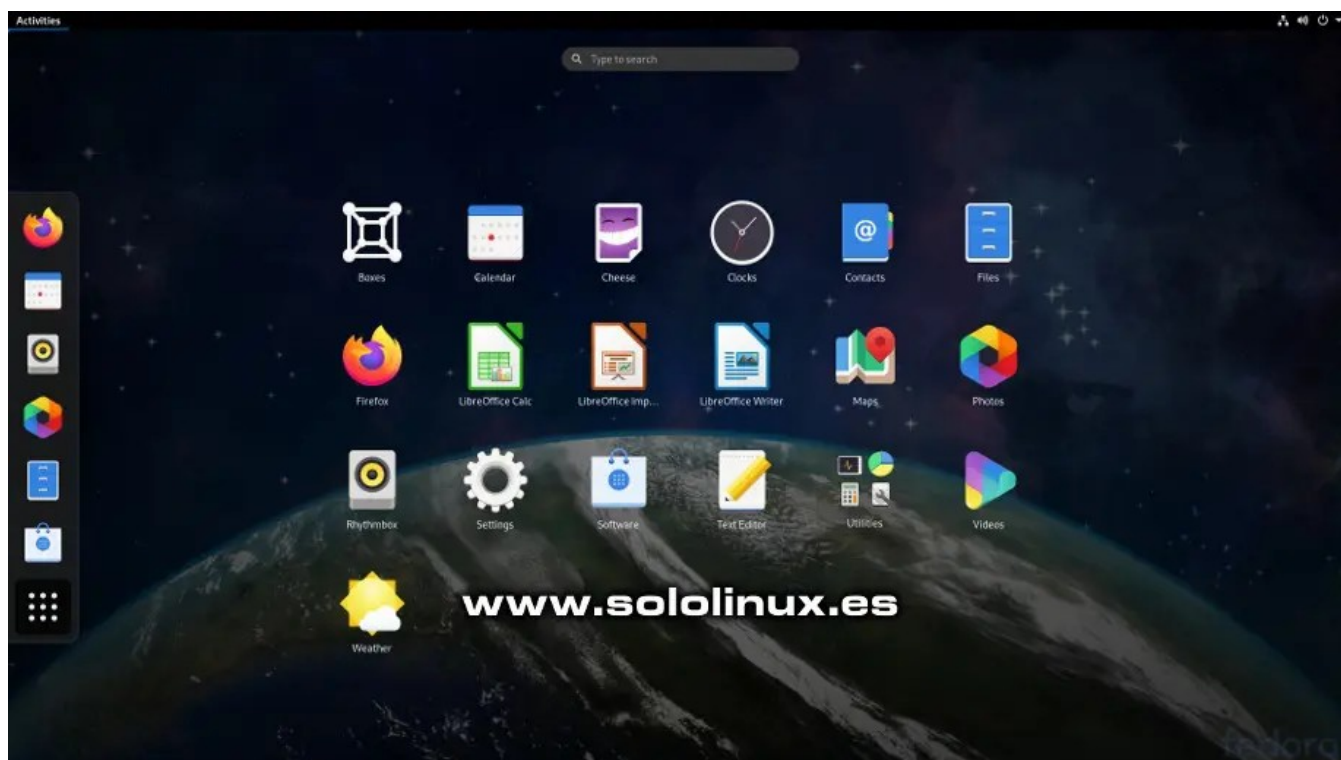
```
sudo dnf system-upgrade download --releasever=33
```

El comando anterior descarga las actualizaciones del nuevo sistema operativo. Al concluir debes ejecutar el siguiente comando.

```
sudo dnf system-upgrade reboot
```

Automáticamente, el sistema se reinicia y ejecuta todas las actualizaciones que requiere tu Fedora 32. Este punto de la actualización se puede tomar un tiempo, sé paciente el resultado vale la pena.

Al completar las actualizaciones, el sistema se reiniciará de nuevo y nos ofrecerá nuestro nuevo y flamante Fedora 33. Fácil, fácil.

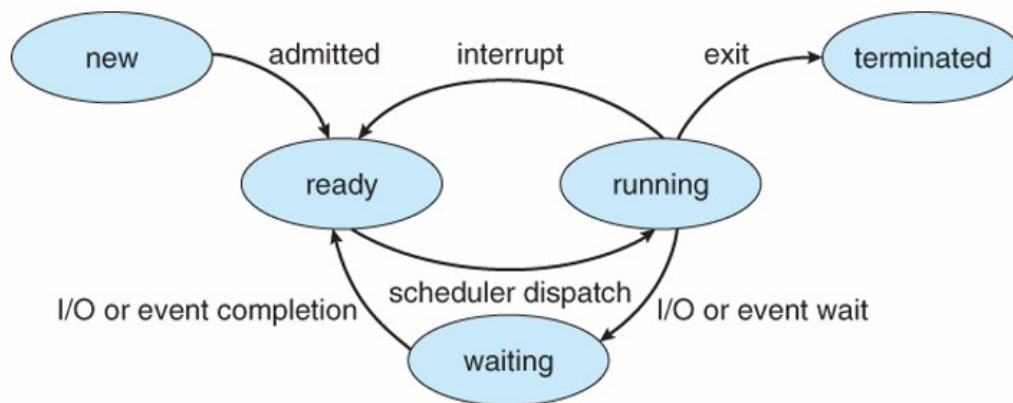


Listar los procesos que se ejecutan en linux

La instancia de una herramienta (o aplicación) en ejecución se conoce como **proceso**. Cada vez que lanzamos un comando de **shell**, inicia una herramienta definida y se crea un proceso exclusivo para él. En linux, cada proceso tiene su **ID** también conocido como **PID**, que a su vez está asociado a un usuario y grupo específico.

Linux es un **sistema** operativo **multitarea**, por tanto puede ejecutar varias aplicaciones y tareas. Cada proceso cree que es el único del sistema, para solucionar este problema tenemos aplicaciones en el kernel que se encargan de poner orden.

Todas las tareas del sistema comparten los mismos recursos, tanto CPU como memoria. Para ser un buen **sysadmin linux**, debes estar familiarizado con una gama de comandos que nos permiten enumerar, identificar o eliminar un proceso. Nos adentramos un poco en este mundo, aprendiendo a listar o enumerar los procesos de nuestro sistema.



Listar los procesos que se ejecutan en linux

Listar los procesos en ejecución

Vemos diversas opciones de uso para listar todos los procesos, con el comando ps.

ps aux

Este comando es el que ofrece más detalles.

```
ps aux
```

Ejemplo...

```

sergio@sololinux:~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.2 160112  9468 ?        Ss   11:07   0:13 /sbin/init
root         2  0.0  0.0      0     0 ?        S    11:07   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        I<   11:07   0:00 [rcu_gp]
root         4  0.0  0.0      0     0 ?        I<   11:07   0:00 [rcu_par_gp]
root         6  0.0  0.0      0     0 ?        I<   11:07   0:01 [kworker/0:0H-k
root         9  0.0  0.0      0     0 ?        I<   11:07   0:00 [mm_percpu_wq]
root        10  0.0  0.0      0     0 ?        S    11:07   0:01 [ksoftirqd/0]
  
```

ps -ef

Enumera todos los procesos que se ejecutan actualmente, con una sintaxis estándar.

```
ps -ef
```

Ejemplo...

```

sergio@sololinux:~$ ps -ef
UID        PID  PPID  C  STIME TTY          TIME CMD
root         1     0  0  11:07 ?        00:00:13 /sbin/init
root         2     0  0  11:07 ?        00:00:00 [kthreadd]
root         3     2  0  11:07 ?        00:00:00 [rcu_gp]
root         4     2  0  11:07 ?        00:00:00 [rcu_par_gp]
root         6     2  0  11:07 ?        00:00:01 [kworker/0:0H-kb]
root         9     2  0  11:07 ?        00:00:00 [mm_percpu_wq]
root        10     2  0  11:07 ?        00:00:01 [ksoftirqd/0]
  
```

ps -ely

Otro formato con algún detalle más que el comando anterior.

```
ps -ely
```

Se imprime lo siguiente...

```
sergio@sololinux:~$ ps -ely
S  UID  PID  PPID  C  PRI  NI  RSS  SZ  WCHAN  TTY  TIME  CMD
S   0    1    0    0  80    0  9468 40028 -      ?    00:00:13 systemd
S   0    2    0    0  80    0    0    0 -      ?    00:00:00 kthreadd
I   0    3    2    0  60 -20    0    0 -      ?    00:00:00 rcu_gp
I   0    4    2    0  60 -20    0    0 -      ?    00:00:00 rcu_par_gp
I   0    6    2    0  60 -20    0    0 -      ?    00:00:01 kworker/0:0H
I   0    9    2    0  60 -20    0    0 -      ?    00:00:00 mm_percpu_wq
S   0   10    2    0  80    0    0    0 -      ?    00:00:01 ksoftirqd/0
```

Listar procesos por usuario

Para listar los procesos de cada usuario, usamos la siguiente sintaxis.

```
ps -U USER -u USER u
```

Vemos un ejemplo con el usuario Sergio.

```
ps -U sergio -u sergio u
```

Verás algo similar a...

```
sergio@sololinux:~$ ps -U sergio -u sergio u
USER      PID  %CPU  %MEM  VSZ  RSS  TTY  STAT  START  TIME  COMMAND
sergio    1015  0.0   0.2  76944  8156 ?    Ss    11:07  0:00  /lib/systemd/sy
sergio    1016  0.0   0.0  132572  2996 ?    S     11:07  0:00  (sd-pam)
sergio    1029  0.0   0.2  285632  7984 ?    Sl    11:07  0:00  /usr/bin/gnome-
sergio    1032  0.0   0.5  428172  21956 ?    Ssl   11:07  0:00  xfce4-session
sergio    1094  0.0   0.1  50824  5280 ?    Ss    11:07  0:14  /usr/bin/dbus-d
sergio    1204  0.0   0.0  11312  324 ?    Ss    11:07  0:00  /usr/bin/ssh-ag
sergio    1259  0.0   0.1  349376  6388 ?    Ssl   11:07  0:00  /usr/lib/at-spi
sergio    1264  0.0   0.1  50056  4544 ?    S     11:07  0:02  /usr/bin/dbus-d
sergio    1267  0.0   0.1  220660  7016 ?    Sl    11:07  0:05  /usr/lib/at-spi
```

Listar los procesos en formato árbol

ps axjf

Es la manera más cómoda de entender los procesos padre e hijo.

```
ps axjf
# 0
ps -ef --forest
```

Observa...

```
784 794 794 794 tty7 794 Ssl+ 0 11:44 \_ /usr/lib/xorg/Xorg
784 1003 784 784 ? -1 Sl 0 0:00 \_ lightdm --session-c
1003 1032 1032 1032 ? -1 Ssl 1000 0:00 \_ xfce4-session
1032 1204 1204 1204 ? -1 Ss 1000 0:00 \_ /usr/bin/ss
1032 1282 1032 1032 ? -1 Sl 1000 0:21 \_ xfwm4
1032 1300 1032 1032 ? -1 Sl 1000 1:04 \_ xfce4-panel
1300 1314 1032 1032 ? -1 Sl 1000 0:17 | \_ /usr/li
1314 4423 1032 1032 ? -1 Sl 1000 0:01 | | \_ lea
1314 6596 1032 1032 ? -1 Sl 1000 0:00 | | \_ /us
6596 6613 1032 1032 ? -1 Sl 1000 0:09 | | \_
1300 1319 1032 1032 ? -1 Sl 1000 0:00 | | \_ /usr/li
1300 1320 1032 1032 ? -1 Sl 1000 0:02 | | \_ /usr/li
1300 1321 1032 1032 ? -1 Sl 1000 0:06 | | \_ /usr/li
1300 1322 1032 1032 ? -1 Sl 1000 1:29 | | \_ /usr/li
1300 1526 1032 1032 ? -1 Sll 1000 20:06 | | \_ /usr/li
1526 1542 1032 1032 ? -1 S 1000 0:00 | | \_ /us
1542 1563 1032 1032 ? -1 Sl 1000 25:26 | | \_ \_
1526 1543 1032 1032 ? -1 S 1000 0:00 | | \_ /us
1543 1545 1032 1032 ? -1 S 1000 0:00 | | \_ \_
```


pstree -p

Quizás no es tan completo como el comando anterior, pero visualmente aún es más comprensible.

```
pstree -p
```

```
sergio@sololinux:~$ pstree -p
systemd(1)─ModemManager(616)─{ModemManager}(627)
               │               │{ModemManager}(634)
               └─NetworkManager(635)─dhclient(1073)
                                   │{NetworkManager}(659)
                                   │{NetworkManager}(664)
               └─accounts-daemon(596)─{accounts-daemon}(598)
                                   │{accounts-daemon}(633)
               └─acpid(662)
               └─agetty(795)
               └─avahi-daemon(601)─avahi-daemon(604)
               └─bluetoothd(874)
               └─cron(608)
               └─cups-browsed(9218)─{cups-browsed}(9227)
                                   │{cups-browsed}(9228)
               └─cupsd(9217)
               └─dbus-daemon(620)
```

El comando pstree, también admite listar los procesos de un usuario.

```
pstree -p sergio
```

```
sergio@sololinux:~$ pstree -p sergio
gimp-2.8(5896)─script-fu(5908)─{script-fu}(5912)
               │               │{script-fu}(5913)
               │               │{gimp-2.8}(5899)
               │               │{gimp-2.8}(5900)
               │               │{gimp-2.8}(5902)
               │               │{gimp-2.8}(5903)
               └─gnome-keyring-d(1029)─{gnome-keyring-d}(1030)
                                   │{gnome-keyring-d}(1031)
                                   │{gnome-keyring-d}(1584)
               └─xfce4-session(1032)─Thunar(1308)─{Thunar}(1309)
                                   │{Thunar}(1310)
                                   └─applet.py(1364)─{applet.py}(1433)
                                   └─light-locker(1349)─{light-locker}(1367)
                                   │{light-locker}(1371)
                                   │{light-locker}(1374)
```

Listar los procesos que se ejecutan en linux**Número de subprocesos de cada proceso**

```
ps -eLf
```

```
ps -eLf
```

Los podemos identificar en la columna marcada como «LWP».

sergio	5373	1015	5390	0	5	12:53	?	00:00:00	/usr/libexec/xdg-desk
sergio	5373	1015	11793	0	5	16:50	?	00:00:00	/usr/libexec/xdg-desk
sergio	5377	1015	5377	0	6	12:53	?	00:00:00	/usr/libexec/xdg-docu
sergio	5377	1015	5378	0	6	12:53	?	00:00:00	/usr/libexec/xdg-docu
sergio	5377	1015	5379	0	6	12:53	?	00:00:00	/usr/libexec/xdg-docu
sergio	5377	1015	5387	0	6	12:53	?	00:00:00	/usr/libexec/xdg-docu
sergio	5377	1015	5388	0	6	12:53	?	00:00:00	/usr/libexec/xdg-docu
sergio	5377	1015	5389	0	6	12:53	?	00:00:00	/usr/libexec/xdg-docu
sergio	5380	1015	5380	0	3	12:53	?	00:00:00	/usr/libexec/xdg-perm
sergio	5380	1015	5381	0	3	12:53	?	00:00:00	/usr/libexec/xdg-perm

Mostrar el PID y el comando

ps -eo pid,comm

Si solo necesitas averiguar el PID y el comando ejecutado, esto es para ti.

```
ps -eo pid,comm
```

Ejemplo de salida...

```
sergio@sololinux:~$ ps -eo pid,comm
PID COMMAND
 1 systemd
 2 kthreadd
 3 rcu_gp
 4 rcu_par_gp
 6 kworker/0:0H-kb
 9 mm_percpu_wq
10 ksoftirqd/0
11 rcu_sched
```

Detalles de consumo de cada comando

ps -eo pid,%mem,%cpu,rss,rsz,vsz,comm

Interesante comando para identificar consumos por proceso, tanto de la cpu como de la memoria. Listar los procesos que se ejecutan en linux.

```
ps -eo pid,%mem,%cpu,rss,rsz,vsz,comm
```

Ejemplo...

```
sergio@sololinux:~$ ps -eo pid,%mem,%cpu,rss,rsz,vsz,comm
PID %MEM %CPU  RSS  RSZ  VSZ  COMMAND
 1  0.2  0.0  9468  9468  160112  systemd
 2  0.0  0.0    0    0    0  kthreadd
 3  0.0  0.0    0    0    0  rcu_gp
.....
5391  0.4  0.0 16524 16524 536936 xdg-desktop-por
5896  3.2  0.5 129304 129304 1113204 gimp-2.10
5908  0.4  0.0 16764 16764 306652 script-fu
6573  6.3 11.1 251404 251404 5666120 chromium-browser
6596  0.1  0.0  6364  6364 215004 oosplash
6613  5.6  0.0 222460 222460 1148452 soffice.bin
7793  0.0  0.1    0    0    0  kworker/0:4-eve
9217  0.2  0.0  8612  8612 105136 cupsd
9218  0.2  0.0 11024 11024 303532 cups-browsed
```

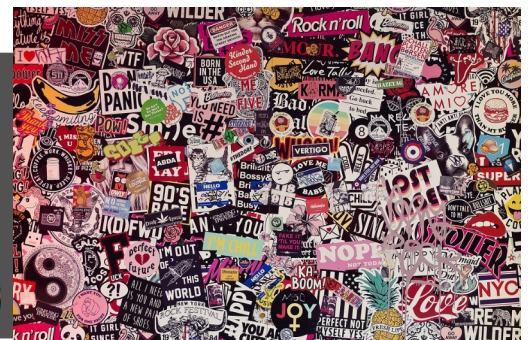
Hasta aquí el artículo sobre «Listar los procesos que se ejecutan en linux», en el próximo artículo veremos como identificar el PID de un proceso y, el proceso de un PID.

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es



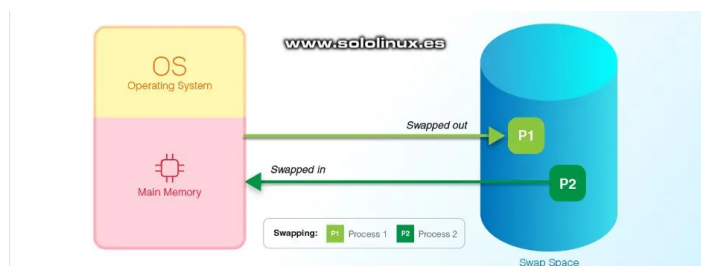
Equilibrar la carga entre la memoria ram y swap



En nuestro sistema operativo Linux, tenemos una cantidad dedicada de **memoria RAM** que hace posible que se ejecuten los procesos de una aplicación. Por desgracia, la **memoria ram** está limitada por nuestro hardware, por tanto no puede guardar excesivos datos.

Cuando en **linux** se agota la memoria RAM (se llena), toma prestado lo necesario de una memoria física que creamos en la instalación del sistema. Hablamos de la memoria de almacenamiento secundario, que acumula el contenido inactivo. Más conocida como **swap**, realmente es una pequeña porción de nuestro disco **hdd o ssd** aislado para tal uso.

De esta manera, la RAM encuentra suficiente espacio para guardar sus nuevos procesos dentro de ella. El equilibrio entre las dos memorias no siempre es el correcto, en el artículo de hoy veremos como modificarlo.



Equilibrar la carga entre la memoria ram y swap

El equilibrio de la carga se define por una numeración en **swappiness** (una herramienta del **kernel linux**). Nos ayudamos del **comando cat** para visualizarlo.

```
cat /proc/sys/vm/swappiness
```

La mayoría de las distribuciones linux, establecen por defecto este valor en 60. Vemos un ejemplo.

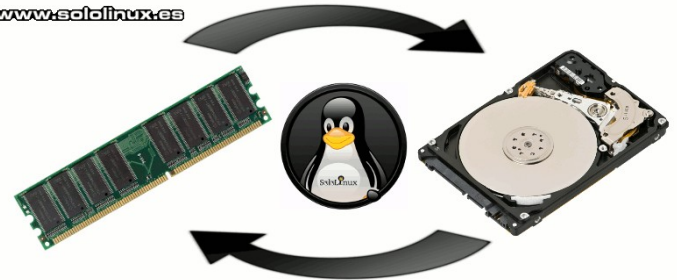
```
sergio@sololinux:~$ cat /proc/sys/vm/swappiness
60
sergio@sololinux:~$
```

Identificar el valor correcto para un sistema no es tarea sencilla, lo más fácil es que modifiques el 60 (que suele venir por defecto) por otros valores, como 10, 20, 30, 40 y tu mismo verifiques cuál te ofrece un mejor rendimiento. Puedes cambiar el valor con el siguiente comando.

```
nano /proc/sys/vm/swappiness
```

Normalmente verás el 60, modifícalo por ejemplo por 10. Guarda el archivo y cierra el editor. No reinicies el sistema, dado que se perderá este valor (es temporal). Realiza la misma operación con otros valores hasta que localices es que más se adapta a tu sistema. Cuanto más pequeño sea el valor, más tardara el sistema por recurrir a la **swap**, algo que viene muy bien en equipos antiguos que usan discos mecánicos de bajas revoluciones (5400 rpm).

www.sololinux.es



Hacer permanente la modificación

Una vez tengas claro el valor perfecto, de equilibrar la carga entre la memoria ram y swap de tu sistema linux, lo puedes hacer permanente. Ejecutamos el siguiente comando.

```
sudo nano /etc/sysctl.conf
```

Con las flechas del teclado bajas al final del archivo, ahora agrega lo siguiente (con el valor que te convenga).

```
vm.swappiness=20
```

Guarda el archivo y cierra el editor. Ya puedes reiniciar el sistema, hemos terminado de equilibrar la carga de las memorias.

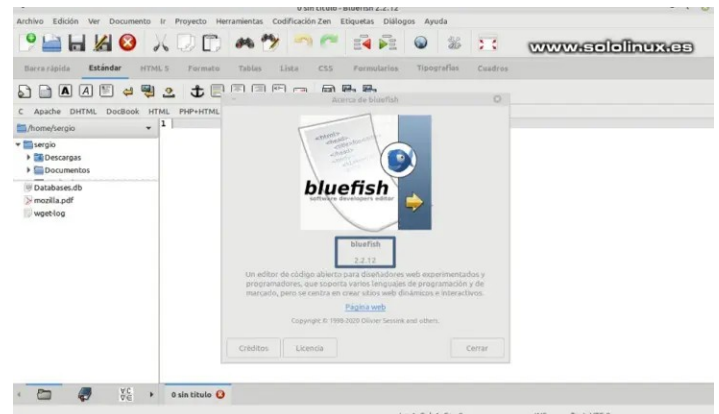
```
sudo reboot
```

Novedad – Instalar el editor Bluefish 2.2.12 en Ubuntu

Bluefish es un potente editor de código en pantalla gráfica, creado para desarrolladores y programadores web profesionales o aficionados. Admite la mayoría de lenguajes de programación y de marcado, a los cuales aporta grandes ventajas en su desarrollo.

Hace pocos días, se lanzó la nueva versión 2.2.12 con algunas características nuevas y otras correcciones menores. Vemos sus principales características.

- Correcciones en la búsqueda simple.
- La compatibilidad con **Python 3**, ha mejorado de forma considerable.
- La detección de codificación en Python aumenta.
- Permite seleccionar líneas con triple clic.
- Nuevas funciones de permisos en sistemas basados en Unix.
- El resaltado de sintaxis se amplía a nuevos lenguajes de programación.
- Ahora, **Bluefish** es 100% compatible con el corrector ortográfico **Enchant2**.



Instalar el editor Bluefish 2.2.12 en Ubuntu

Desde ayuda/información identificamos la versión instalada. En nuestro caso la 2.2.10.

Para **instalar o actualizar Bluefish**, agregamos el siguiente **ppa**.

```
sudo add-apt-repository ppa:klaus-vormweg/bluefish
```

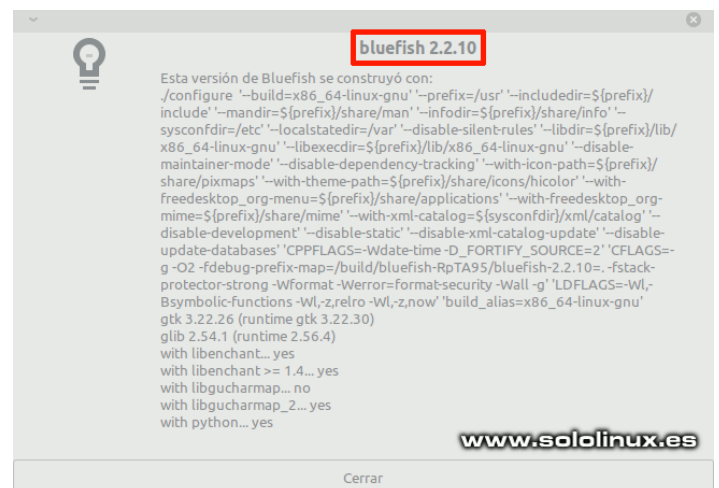
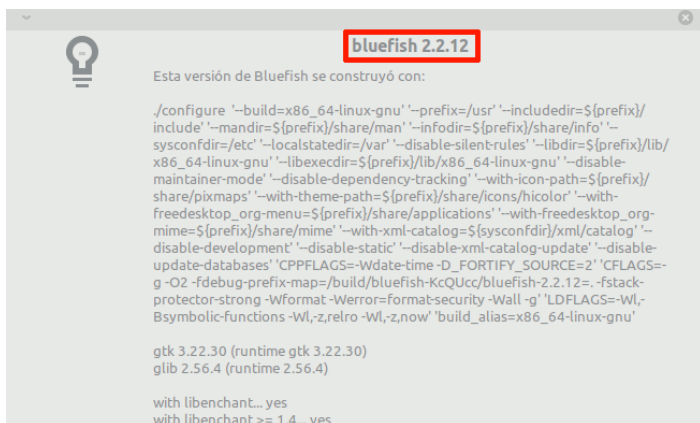
Instalamos la versión mejorada de Bluefish con...

```
sudo apt update
sudo apt install bluefish
```

Si ya eras usuario de este excelente editor, actualiza.

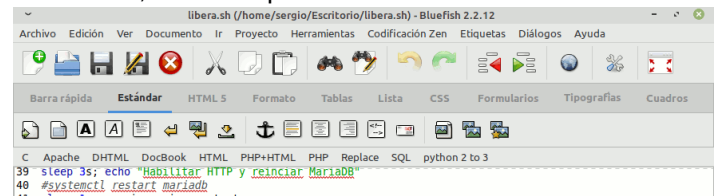
```
sudo apt update
sudo apt full-upgrade
```

Al concluir la instalación o actualización verifica la versión actual.



Estamos ante un editor altamente recomendable para cualquier tipo de usuario, sobre todo para los que comienzan en este mundillo. No tiene nada que envidiar a ningún otro de su clase, tan solo la poca promoción que hacen de él sus desarrolladores.

Pruébalo, no te arrepentirás.



Si por algún caso quieres desinstalar la nueva versión, puedes hacerlo sin problema alguno (ejecuta todos los comandos para volver a la versión anterior).

```
sudo add-apt-repository --remove ppa:klaus-vormweg/bluefish
sudo apt remove --purge bluefish
sudo apt autoremove
# Instalamos la version anterior
sudo apt update
sudo apt install bluefish
```


Instalar GIMP 2.99.2 en Ubuntu y derivados



GIMP 2.99.2, es la primera versión en desarrollo lanzada al público en general (a modo de prueba), de lo que será el próximo lanzamiento importante de la herramienta, hablamos de **GIMP 3.0**.

Algunas de las características que podemos ver en la versión en desarrollo, son las siguientes:

- Interfaz de usuario GTK+ 3.
- Soporte nativo para Wayland y HiDPI.
- Refactorización y pulidos muy mejorados.
- Nueva y moderna API.
- Ahora se admiten complementos Python 3, JavaScript, Lua y Vala.
- Invasión espacial de colores muy mejorada.
- Se permite el renderizado en caché, para mejorar el rendimiento.



Instalar GIMP 2.99.2 en Ubuntu y derivados

GIMP 2.99.2, solo está disponible a través del **repositorio oficial de Flathub**, por tanto, si no lo tienes instalado en tu sistema...

```
sudo apt install flatpak
```

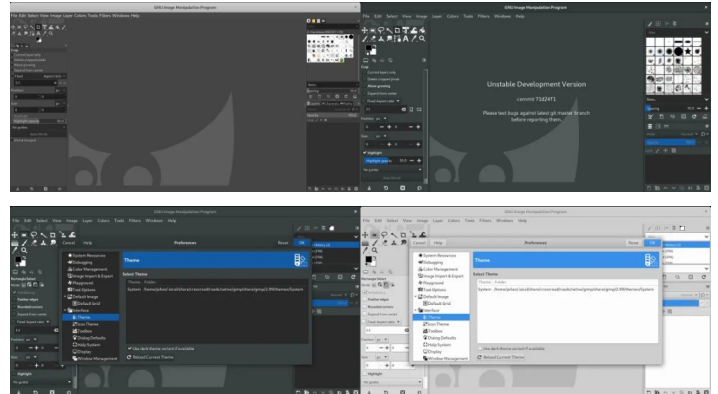
Por defecto, el repositorio no contiene las versiones en desarrollo (solo las estables), así que lo agregamos con el siguiente comando.

```
flatpak remote-add --user flathub-beta
https://flathub.org/beta-repo/flathub-beta.flatpakrepo
```

Instalamos la última versión.

```
flatpak install --user flathub-beta org.gimp.GIMP
```

Una vez concluya la instalación, puedes ejecutar **Gimp 2.99.2** desde el menú de aplicaciones de tu entorno de escritorio.



¡IMPORTANTE!!!

Si la versión de Gimp que tienes en tu sistema, la instalaste desde su paquete flatpak oficial, debes saber que solo puedes ejecutar una de las dos. Para ejecutar Gimp 2.99.2 en vez de la versión estable, ejecuta el siguiente comando.

```
flatpak make-current --user org.gimp.GIMP beta
```

Si quieres volver a la estable...

```
flatpak make-current --user org.gimp.GIMP stable
```

Para borrar el paquete...

```
flatpak uninstall org.gimp.GIMP
```

Cómo instalar Gravit Designer en Linux



En el campo del diseño, la herramienta «**Gravit Designer**» no es de las más conocidas. Aun así es de lo mejor que puedes encontrar si trabajas con ilustraciones y otros tipos de vectores. La flexibilidad es su punto fuerte, pues la tenemos disponible vía web, o desde escritorio (versión pro).

Si te digo la verdad, me suelo negar en rotundo a realizar artículos sobre software de pago (y menos de **Corel**); pero teniendo en cuenta una petición expresa de un usuario (en nuestro [chat de Telegram](#)), y que sí ofrece una versión online free, haremos una excepción sin que sirva de precedente.

Dadas las características de la herramienta, no quiero extenderme en sus bondades y aún menos en su uso. En este artículo nos limitamos a explicar como se instala en **linux** y punto, así que vamos a ello.

Cómo instalar Gravit Designer en Linux

Recuerda que la aplicación que vamos a instalar en nuestro linux, es una demo de 15 días. Si quieres utilizar la versión free online, visita esta [url](#) para acceder.

Comenzamos descargando la herramienta.

- [Descargar Gravit Designer Pro](#)

Una vez descargada la herramienta (normalmente en la carpeta Descargas o Downloads), la descomprimes como normalmente lo haces desde tu ratón, botón derecho, y en el menú haces click en «Extraer aquí».

Bien... ahora abres la terminal en la carpeta descomprimida de la herramienta, por ejemplo...

```
cd Descargas/GravitDesigner
```

Copiamos el binario a /opt.

```
sudo cp GravitDesigner.AppImage /opt
```

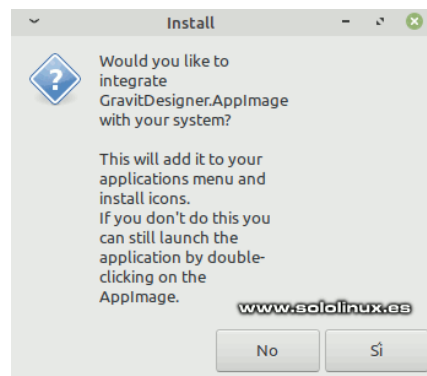
Nos movemos a /opt, y le concedemos los permisos de ejecución necesarios al archivo.

```
cd /opt
sudo chmod +x GravitDesigner.AppImage
```

Bien, ya lo tenemos. Lanzamos **Gravit Designer**.

```
sudo ./GravitDesigner.AppImage
```

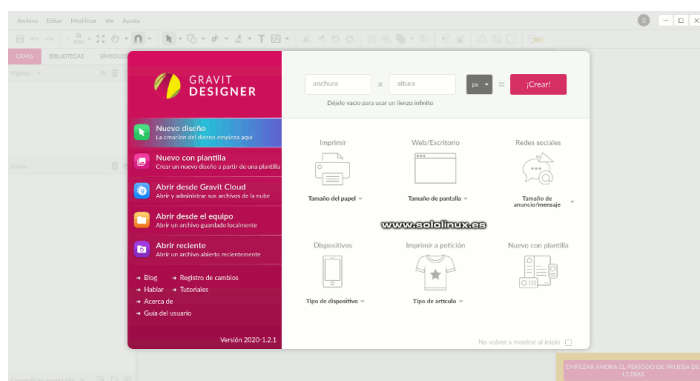
Nos aparece una ventana de diálogo que nos solicita crear un lanzador en tu menú de aplicaciones. Pulsa Aceptar, si no lo haces deberás iniciar desde la terminal, o directamente sobre el binario. Instalar Gravit Designer.



Al iniciar desde el menú de aplicaciones de tu escritorio, nos solicita el registro o inicio de sesión.



Seleccionas la plantilla o archivo, y comienzas a trabajar.



Probar el rendimiento del disco con KDiskMark



KDiskMark es una alternativa de **código abierto** a la conocida herramienta para Windows, **CrystalDiskMark**. **KDiskMark** nos presenta una interfaz de usuario simple, sencilla y fácil de usar. En realidad, es como si fuera un **clon de CrystalDiskMark**.

Esta herramienta nos ayuda a probar el rendimiento del disco (**HDD o SSD**), también debemos destacar su alta configuración, como el tamaño del bloque, las colas de trabajo y el recuento de subprocesos en cada prueba. La aplicación genera informes de referencia que podrás analizar más tarde, incluso compartirllos fácilmente para realizar comparativas. Probar el rendimiento del disco.

Principales funciones y características de **KDiskMark**:

- Interfaz agradable y sencilla de utilizar.
- Muestra los resultados de la velocidad de lectura y escritura.
- Configuración de la prueba:
 - Tamaño del bloque
 - Longitud de la cola
 - Cantidad de subprocesos para cada prueba
 - Permite pausar entre pruebas
- Perfiles para pruebas preestablecidos por defecto.
- Generador de completos informes.

Como puedes ver, puede resultar una herramienta de gran utilidad.

	Read [MB/s]	Write [MB/s]
SEQ1M Q8T1	465.48	172.13
SEQ1M Q1T1	431.19	282.40
RND4K Q32T16	199.84	123.10
RND4K Q1T1	35.64	116.44

Samsung SSD 850 www.sololinux.es

Probar el rendimiento del disco con KDiskMark
Antes de proceder a instalar KDiskMark, debes asegurarte que tu sistema tiene instaladas las dependencias necesarias.

- **GCC/Clang C++17** (o superior)
- **CMake** >= 3.5
- **Extra CMake Modules**
- **Qt with Widgets** >= 5.9
- **Flexible I/O Tester** with libaio >= 3.1
 - If you build FIO from source, install libaio-dev package.

Ahora vamos a instalar **KDiskMark**.

Ubuntu, Linux Mint y derivados:

```
sudo add-apt-repository ppa:jonmagon/kdiskmark
sudo apt update
sudo apt install kdiskmark
```

Arch Linux, Manjaro y derivados:

```
sudo pacman -S kdiskmark
```

Fedora y derivados:

```
sudo dnf install kdiskmark
```

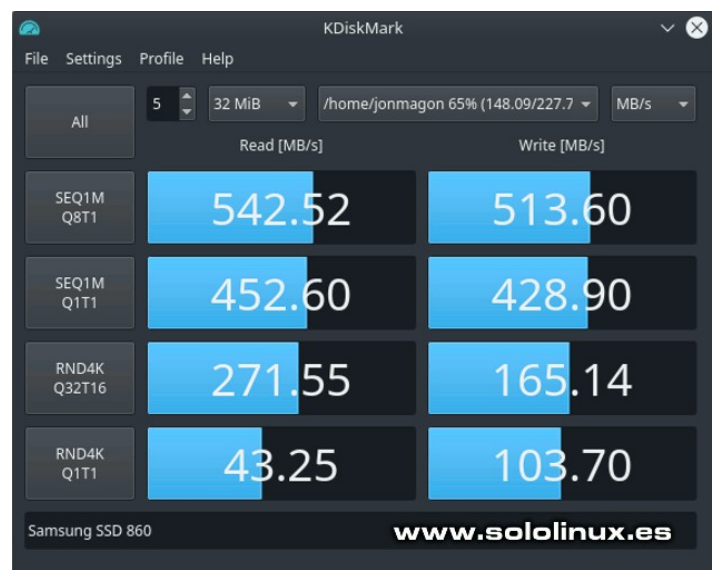
Suse, OpenSuse y derivados:

```
sudo zypper install kdiskmark
```

Paquete Snap:

```
sudo snap install kdiskmark
```

Una vez instalada la herramienta, la puedes ejecutar desde el icono de tu menú de aplicaciones. Su uso es bastante sencillo, imposible de cometer un error.



Actualizar Transmission en Ubuntu 20.04 y derivados

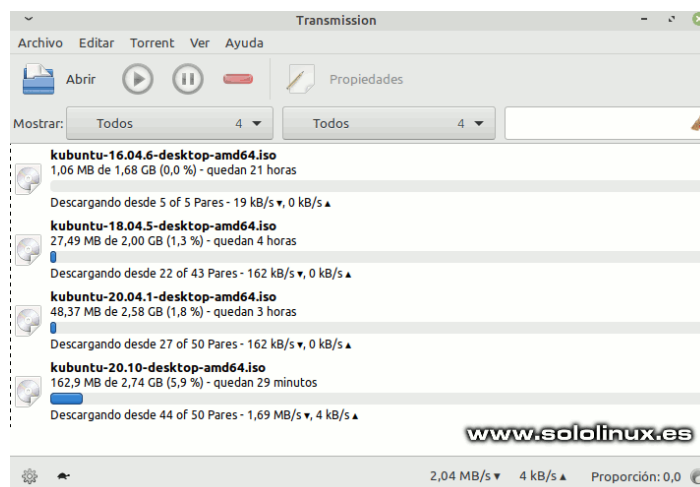


Hace unos meses que se lanzó la nueva versión del **cliente de torrents Transmission**. Por motivos diversos, me veo obligado a utilizarla casi a diario. Ayer quería limitar el tiempo de permanencia de las semillas inactivas, vaya por dios, es imposible.

Resulta, que como el sistema donde tengo instalado **Transmission** está basado en **Ubuntu 18.04**, viene con la versión 2.92 y, no incluye la característica que necesito. Tengo que actualizar a **Transmission 3.0**.

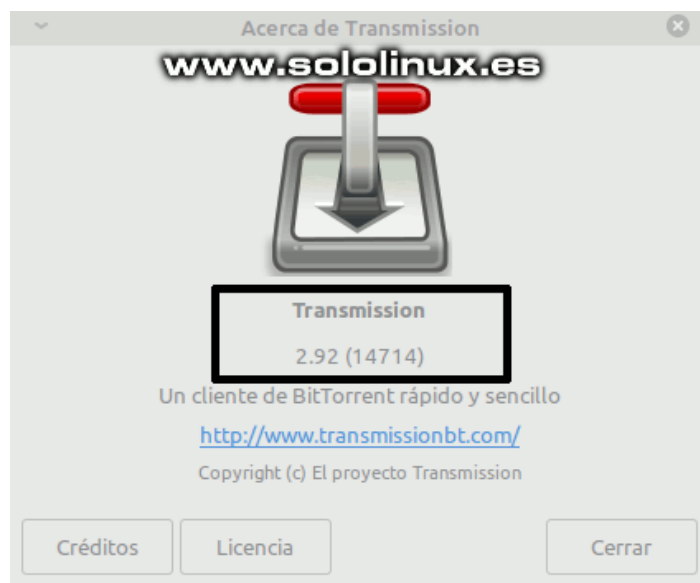
Transmission 3.0 representa una actualización importante respecto a sus antecesores, que cuentan con más de dos años a sus espaldas. Esta nueva versión no solo se limita a correcciones de errores, también aporta nuevas funciones y otras mejoradas (actualizar Transmission). Vemos las más importantes.

- Manejo del argumento «campos» en la solicitud RPC «obtener sesión».
- Se mejora el soporte para IPv6.
- Permite limitar los intentos de autenticación no válidos en su servidor web.
- Se establece un rango límite de semillas inactivas a 4 semanas.
- Agregar peer ID para torrent y clientes en descarga.
- El manejo de los rastreadores de Amazon S3, ha mejorado considerablemente.
- Se agrega soporte para **TCP_FASTOPEN**.
- Ahora se anula el protocolo de enlace, si falla el establecimiento codificado compartido DH.
- Detecta mucho mejor si una sesión es local o remota.
- Se permite mover la ubicación del **torrent** (incluso si no debes mover los datos).
- Admite listas de bloqueo con CIDR. Actualizar Transmission.
- En el encabezado de la **session-id**, no se distingue entre mayúsculas y minúsculas.
- Limpiar componentes de rutas sospechosas.
- Se agrega soporte para mbedtls, wolfssl y LibreSSL.
- Consumos de memoria inútiles reparados.
- Nuevo icono interactivo en su versión GTK.



Actualizar Transmission en Ubuntu 20.04 y derivados

Hasta el lanzamiento de **Ubuntu 20.10 Groovi**, **Transmission 3.0** no aparece en los repositorios oficiales. Esto quiere decir que todas las versiones anteriores de Ubuntu, todas las de **Linux Mint** y sus respectivos derivados... no cuentan con la **aplicación Transmission 3.0** de forma nativa.



Por suerte, esto tiene fácil solución. El desarrollador **Andreas Noteng**, mantiene un repositorio ppa del cliente **Transmission** siempre al día, por tanto, para **actualizar Transmission** en nuestro sistema debemos agregarlo. Sigue los pasos indicados.

```
sudo add-apt-repository ppa:transmissionbt/ppa
```

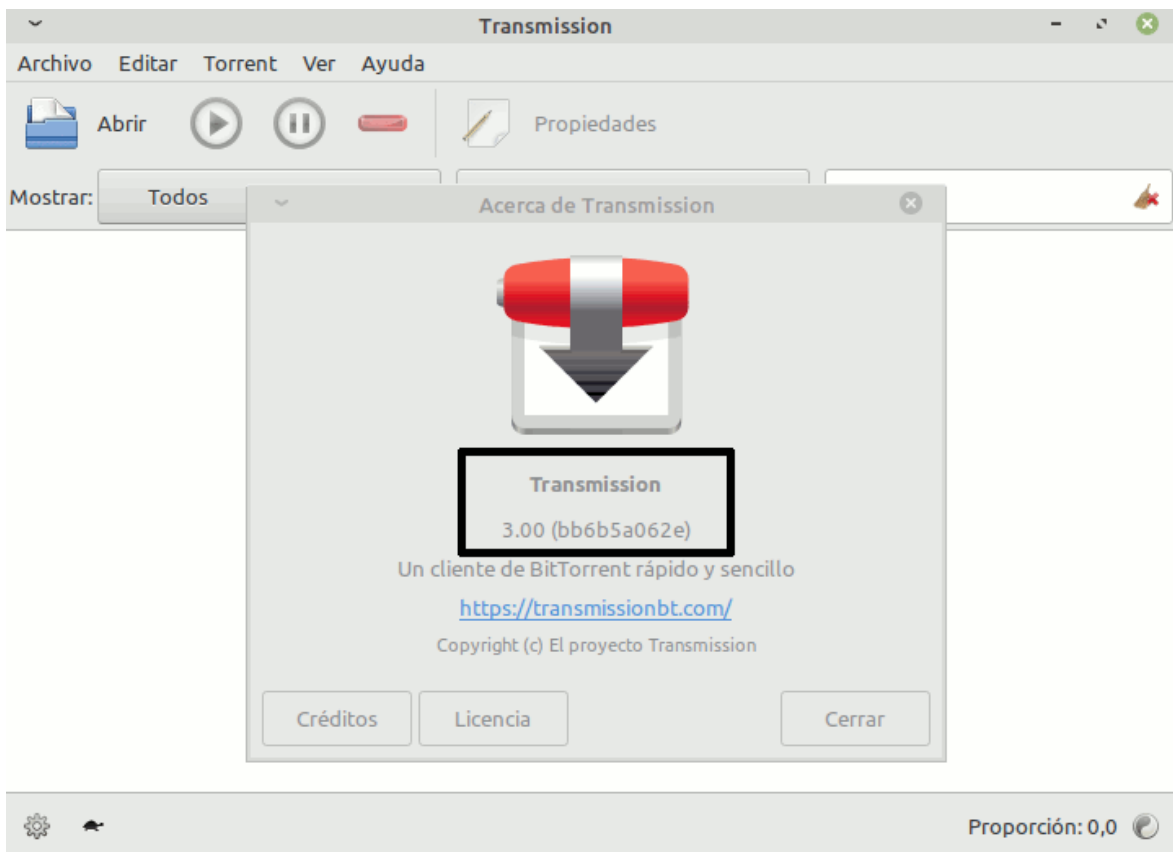
Actualiza e instala la nueva versión.

```
sudo apt update
sudo apt install transmission-gtk
```

Ejemplo de salida...

```
sergio@sololinux:~$ sudo apt install transmission-gtk
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  transmission-common
Se actualizarán los siguientes paquetes:
  transmission-common transmission-gtk
2 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 984 kB de archivos.
Se utilizarán 2.750 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
Des:1 http://ppa.launchpad.net/transmissionbt/ppa/ubuntu bionic/main amd64 transmission-common all 3.00-1ubuntu1~18.04.3 [246 kB]
Des:2 http://ppa.launchpad.net/transmissionbt/ppa/ubuntu bionic/main amd64 transmission-gtk amd64 3.00-1ubuntu1~18.04.3 [738 kB]
Descargados 984 kB en 2s (477 kB/s)
(Leyendo la base de datos ... 341611 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar ../transmission-common_3.00-1ubuntu1~18.04.3_all.deb ...
Desempaquetando transmission-common (3.00-1ubuntu1~18.04.3) sobre (2.92-3ubuntu2) ...
Preparando para desempaquetar ../transmission-gtk_3.00-1ubuntu1~18.04.3_amd64.deb ...
Desempaquetando transmission-gtk (3.00-1ubuntu1~18.04.3) sobre (2.92-3ubuntu2) ...
Configurando transmission-common (3.00-1ubuntu1~18.04.3) ...
Configurando transmission-gtk (3.00-1ubuntu1~18.04.3) ...
Procesando disparadores para desktop-file-utils (0.23+linuxmint8) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
Procesando disparadores para gnome-menus (3.13.3-1ubuntu1.1) ...
Procesando disparadores para hicolor-icon-theme (0.17-2) ...
Procesando disparadores para mime-support (3.60ubuntu1) ...
```

Al concluir la actualización puedes abrir la herramienta, desde tu menú de aplicaciones. Verifica que se actualizó correctamente en «**Ayuda / Acerca de...**».



Como crear archivos torrent con Transmission



Los **archivos torrent**, son sin dudarlos una de las formas más populares de compartir archivos online. En muchos casos, las descargas por **torrent** son más eficientes y rápidas que subir archivos en algún sitio de almacenamiento, y ofrecer la descarga desde allí.

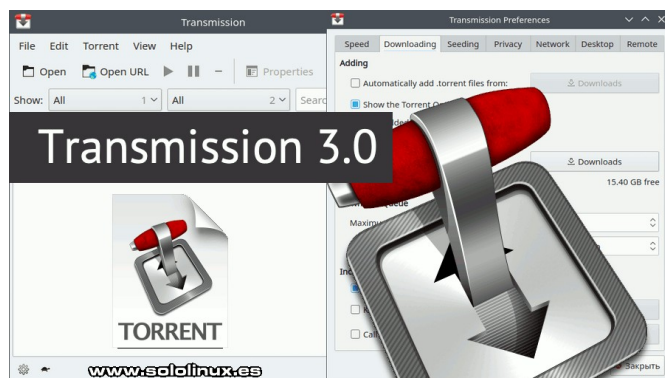
Lo dicho anteriormente se logra gracias al tipo de protocolo que usamos en **torrent**, dado que este, fue diseñado específicamente para el intercambio de archivos entre iguales (**peer-to-peer**) a través de Internet. Tiene la particularidad de dividir el archivo de destino, en pequeños fragmentos de información y datos para lograr una alta velocidad.

Seguro que más de una vez te has planteado crear archivos torrent (tus torrent), para poder compartir tus propios archivos.

Bueno... pues nada más fácil que realizar esa operación. Lo único que necesitas, es tener instalado un **cliente torrent** que admita la función de generar los archivos de **metadatos** necesarios para poder compartir el **torrent**. Este proceso es muy simple en la mayoría de clientes, en nuestro artículo de ejemplo realizamos la operación en el cliente **Transmission 3.0 para linux**, pues suele venir preinstalado en la mayoría de **distribuciones linux**, además de ser muy fácil. Este artículo también es válido para otros clientes como **qbittorrent**, el proceso es similar.

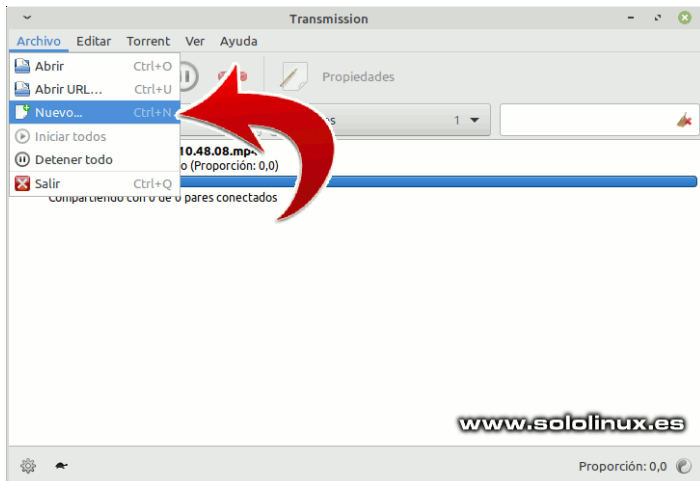
Como crear archivos torrent con Transmission 3.0

Al crear tu propio **archivo torrent**, permitimos que otras personas descarguen archivos directamente desde nuestro sistema. Por ello se le conoce como distribución descentralizada, no existe un servidor dedicado para tal fin. Lo descargan desde tu máquina, y los que lo descargaron se convierten automáticamente en semillas. También es posible cargar primero los archivos en un sitio de almacenamiento online y, posteriormente ofrecer la descarga remota desde allí.



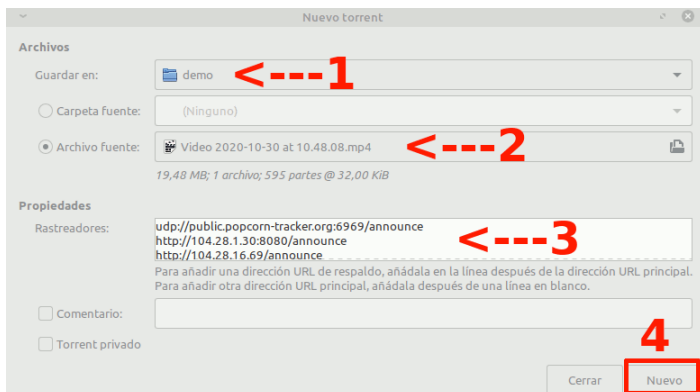
En nuestro caso operamos desde nuestro sistema local, por tanto comenzamos abriendo la **aplicación Transmission**. Sigue los pasos que te indico y no tendrás ningún problema para crear **archivos torrent**.

Una vez abierta la herramienta haces click en «**Archivo**» y en «**Nuevo**».



Vemos como configurar Transmission para crear archivos torrent.

- 1 – Donde queremos guardar el archivo de metadatos torrent.
- 2 – Selecciona el archivo a compartir.
- 3 – Insertamos los rastreadores (*).
- 4 – Pulsa en «**Nuevo**» para crear archivos torrent.



(*) – Los rastreadores, más conocidos como **torrent trackers** son los encargados de enlazar los archivos con los clientes (generar semillas). Los **torrent trackers** van cambiando cada cierto tiempo para evitar su bloqueo, es importante mantenerse al día. Dejo una lista de los activos en este mes de noviembre, del año en curso (2020).

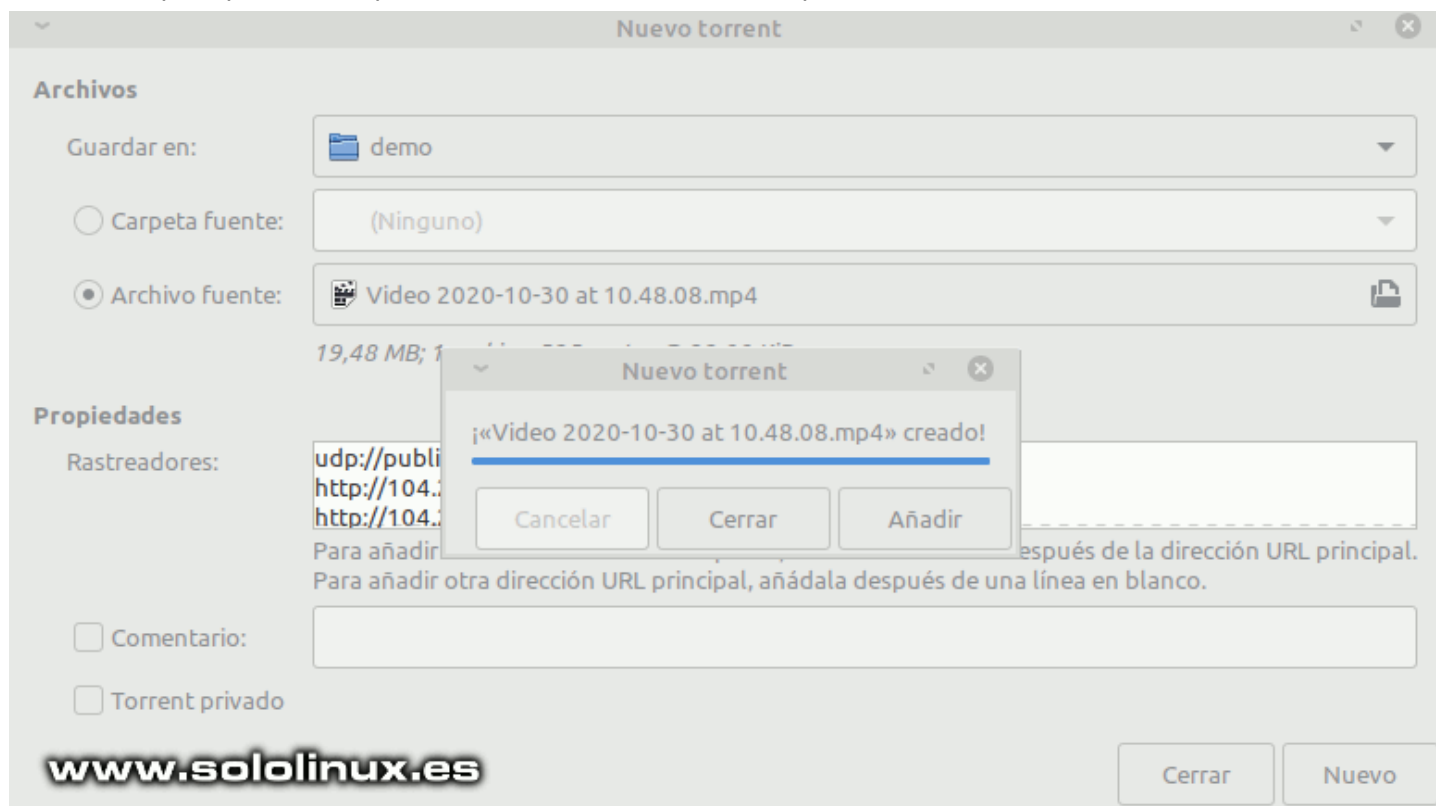
```

udp://public.popcorn-tracker.org:6969/announce
http://104.28.1.30:8080/announce
http://104.28.16.69/announce
http://107.150.14.110:6969/announce
http://109.121.134.121:1337/announce
http://114.55.113.60:6969/announce
http://125.227.35.196:6969/announce
http://128.199.70.66:5944/announce
http://157.7.202.64:8080/announce
http://158.69.146.212:7777/announce
http://173.254.204.71:1096/announce
http://178.175.143.27/announce
http://178.33.73.26:2710/announce
http://182.176.139.129:6969/announce
http://185.5.97.139:8089/announce
http://188.165.253.109:1337/announce
http://194.106.216.222/announce
http://195.123.209.37:1337/announce
http://210.244.71.25:6969/announce
http://210.244.71.26:6969/announce
http://213.159.215.198:6970/announce
http://213.163.67.56:1337/announce
http://37.19.5.139:6969/announce
http://37.19.5.155:6881/announce
http://46.4.109.148:6969/announce
http://5.79.249.77:6969/announce
http://5.79.83.193:2710/announce
http://51.254.244.161:6969/announce
http://59.36.96.77:6969/announce
http://74.82.52.209:6969/announce
http://80.246.243.18:6969/announce
http://81.200.2.231/announce
http://85.17.19.180/announce
http://87.248.186.252:8080/announce
http://87.253.152.137/announce
http://91.216.110.47/announce
http://91.217.91.21:3218/announce
http://91.218.230.81:6969/announce
http://93.92.64.5/announce
http://atrack.pow7.com/announce
http://bt.henbt.com:2710/announce
http://bt.pusacg.org:8080/announce
http://bt2.careland.com.cn:6969/announce
http://explodie.org:6969/announce
http://mgtracker.org:2710/announce
http://mgtracker.org:6969/announce
http://open.acgtracker.com:1096/announce
http://open.lolicon.eu:7777/announce
http://open.touki.ru/announce.php
http://p4p.arenabg.ch:1337/announce
http://p4p.arenabg.com:1337/announce
http://pow7.com:80/announce
http://retracker.gorcomnet.ru/announce
http://retracker.krs-ix.ru/announce
http://retracker.krs-ix.ru:80/announce
http://secure.pow7.com/announce
http://t1.pow7.com/announce
http://t2.pow7.com/announce
http://thetracker.org:80/announce
http://torrent.gresille.org/announce
http://torrentsmd.com:8080/announce
http://tracker.aleatorrenty.pl:2710/announce
http://tracker.baravik.org:6970/announce
http://tracker.bittor.pw:1337/announce
http://tracker.bittorrent.am/announce
http://tracker.calculate.ru:6969/announce
http://tracker.dler.org:6969/announce
http://tracker.dutchtracking.com/announce
http://tracker.dutchtracking.com:80/announce
http://tracker.dutchtracking.nl/announce
http://tracker.dutchtracking.nl:80/announce
http://tracker.edoardocolombo.eu:6969/announce
http://tracker.ex.ua/announce
http://tracker.ex.ua:80/announce
http://tracker.filetracker.pl:8089/announce

```

No es necesario que obligatoriamente insertes todos, todo depende de la difusión que quieras conceder al archivo, pero normalmente con 6 u 8 es suficiente. Si el listado anterior ya no es válido, visita [esta página](#) y usas otros más nuevos.

Continuamos... una vez pulses en «Nuevo» comienza la generación del **archivo torrent**. Normalmente es un proceso bastante rápido, pero todo depende del tamaño del archivo a compartir.



Podrás encontrar el archivo generado en la carpeta seleccionada anteriormente para **guardar el torrent**, compártelo. Debes recordar que compartir archivos protegidos por derechos de autor puede suponer fuertes multas, incluso prisión, no fomentes la piratería si no quieres tener problemas (estás avisado).

Fácil, fácil, verdad?. Vemos una imagen de ejemplo del resultado final.



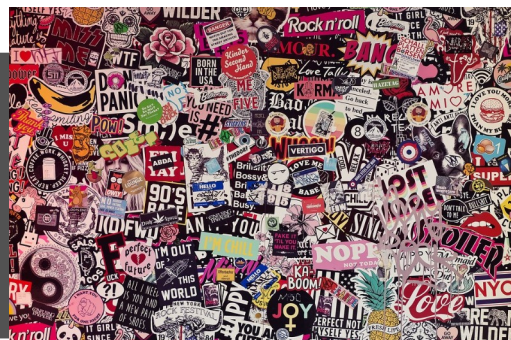
Video 2020-10-30 at
10.48.08.mp4.torrent

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

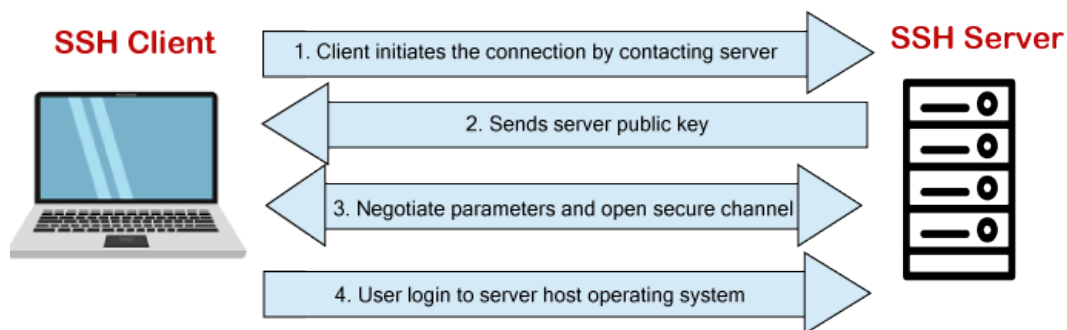
Para mayor información escribe un email a:
adrian@sololinux.es



8 formas de verificar las Conexiones ssh activas

SSH es un protocolo de red que nos permite conectar a sistemas remotos. Es un proceso común, que al administrar servidores Linux, necesites saber quién está conectado a través de SSH (normalmente por seguridad). En el artículo de hoy, veremos una serie de herramientas que nos ayudaran a listar y **verificar las conexiones SSH activas** (sesiones abiertas).

Existen múltiples comandos y herramientas en Linux, que podemos usar para lograr nuestro objetivo. Nosotros vemos las 8 formas de verificar las **conexiones ssh** activas, más habituales.



1 – Comando ss

Con «ss» podemos volcar las estadísticas de sockets.

```
ss | grep -i ssh
```

Ejemplo de salida...

```
[root@sololinux-demo ~]# ss | grep -i ssh
tcp    ESTAB 0      36          185.185.40.172:ssh        62.16.110.193:36018
```

2 – Comando last

Este comando rastrea el **archivo «/var/log/wtmp»**, para identificar los usuarios conectados.

```
last -a | grep -i still
```

En el ejemplo, vemos que la herramienta nos especifica el usuario actualmente conectado.

```
[root@sololinux-demo ~]# last -a | grep -i still
root    pts/1      Mon Nov 16 11:47  still logged in    178.123.69.108
root    pts/0      Mon Nov 16 10:38  still logged in    62.16.102.025
reboot  system boot Sat Nov 7 11:39   still running      4.18.0
```

3 – Comando w

El comando w, nos muestra información sobre los usuarios actuales en la máquina y sus procesos. Esto proporciona información sobre los comandos que utilizan, y también nos permite verificar las conexiones ssh activas.

```
w
```

Observa que además ofrece otros datos relevantes.

```
[root@sololinux-demo ~]# w
11:54:33 up 9 days, 15 min, 2 users, load average: 0,00, 0,01, 0,05
USER  TTY      FROM          LOGIN@   IDLE   JCPU   PCPU   WHAT
root  pts/0    62.16.70.157  10:38    15:37  0.01s  0.01s  -bash
root  pts/1    178.166.49.201 11:47    1.00s  0.00s  0.00s  w
```


4 – Comando who

Who imprime en la consola los usuarios conectados, además de la hora de inicio de sesión.

```
who
```

```
[root@sololinux-demo ~]# who
root    pts/0      2020-11-16 10:38 (62.16.19.212)
root    pts/1      2020-11-16 11:47 (178.103.78.97)
```

5 – Con echo

Con «echo» resolvemos la ip local y remota, así como los puertos. Usamos la siguiente variable...

```
echo $SSH_CONNECTION
```

Veremos algo similar a...

```
[root@sololinux-demo ~]# echo $SSH_CONNECTION
178.133.17.158 24916 185.185.71.68 22
```

6 – Herramienta netstat

Netstat es una herramienta excepcional para tratar con redes, es evidente que también tiene la capacidad de verificar las conexiones ssh activas.

```
netstat -tnpa | grep 'ESTABLISHED.*sshd'
```

Observa su salida impresa.

```
[root@sololinux-demo ~]# netstat -tnpa | grep 'ESTABLISHED.*sshd'
tcp        0      0 185.185.102.77:22 62.16.9.101:36018 ESTABLISHED 34567/sshd: root [p
tcp        0      0 185.185.102.77:22 178.133.55.23:24916 ESTABLISHED 34738/sshd: root [p
```

7 – Comando ps

Es posible que con el **comando ps** no obtengamos información muy precisa, pero a cambio nos ofrece otra información valiosa como el **PID** y más. El **comando ps** nos ofrece diversas variantes a través de sus opciones, nosotros veremos la que tal vez sea más básica, y la que posiblemente sea una de las más completas.

ps auxwww

```
ps auxwww | grep sshd: | grep -v grep
```

```
[root@sololinux-demo ~]# ps auxwww | grep sshd: | grep -v grep
root      35261  0.0  1.8 126172  9732 ?        Ss   14:33   0:00 sshd: root [priv]
root      35270  0.0  1.0 126172  5380 ?        S    14:34   0:00 sshd: root@pts/0
root      35311  0.0  1.3 78520  7068 ?        Ss   14:38   0:00 sshd: [accepted]
sshd      35312  0.0  0.5 78520  2892 ?        S    14:38   0:00 sshd: [net]
```

ps axfj

```
ps axfj | grep sshd
```

```
[root@sololinux-demo ~]# ps axfj | grep sshd
1      469      469 ?        -1 Ss      0      0:31 /usr/sbin/sshd -D -oCiphers=aes256-gcm@openssh.com,
chacha20-poly1305@openssh.com,aes256-ctr,aes256-cbc,aes128-gcm@openssh.com,aes128-ctr,aes128-cbc
-oMACs= hmac-sha2-256-etm@openssh.com,hmac-sha1-etm@openssh.com,umac-128-etm@openssh.com,hmac-sha2-512-etm@
openssh.com,hmac-sha2-256,hmac-sha1,umac-128@openssh.com,hmac-sha2-512 -oGSSAPIKexAlgorithms=gss-gex-sha1-,
gss-group14-sha1 -oKexAlgorithms=curve25519-sha256,curve25519-sha256@libssh.org,ecdh-sha2-nistp256,ecdh-sh
a2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256,diffie-hellman-group14-sha256,diffie-he
llman-group16-sha512,diffie-hellman-group18-sha512,diffie-hellman-group-exchange-sha1,diffie-hellman-group1
4-sha1 -oHostKeyAlgorithms=rsa-sha2-256,rsa-sha2-256-cert-v01@openssh.com,ecdsa-sha2-nistp256,ecdsa-sha2-ni
stp256-cert-v01@openssh.com,ecdsa-sha2-nistp256,rsa-sha2-256-cert-v01@openssh.com,rsa-sha2-256,rsa-sha2-256-cert-v01@openssh.com,ecdsa-sha2-nistp256,ecdsa-sha2-nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp
384,ecdsa-sha2-nistp384-cert-v01@openssh.com,rsa-sha2-512,rsa-sha2-512-cert-v01@openssh.com,ecdsa-sha2-nistp
521,ecdsa-sha2-nistp521-cert-v01@openssh.com,ssh-ed25519,ssh-ed25519-cert-v01@openssh.com,ssh-rsa,ssh-rsa-
cert-v01@openssh.com -oCASignatureAlgorithms=rsa-sha2-256,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,rsa-sha2-
512,ecdsa-sha2-nistp521,ssh-ed25519,ssh-rsa
469      35261  35261  35261 ?        -1 Ss      0      0:00 \_ sshd: root [priv]
35261    35270  35261  35261 ?        -1 S        0      0:00 \_ sshd: root@pts/0
35271    35336  35335  35271 pts/0    S+       0      0:00 \_ grep --color=auto sshd
```

8 – Revisar el registro

Revisar los registros de **actividad ssh** puede ser una forma poco ortodoxa de operar, pero seguro que es efectiva. El principal problema radica que cada **distribución linux** (incluso dependiendo de la versión), tiene su propia ruta en el directorio **/var/log**.

En nuestro caso, el servidor demo es un **CentOS 8**. En CentOS 8 y últimas versiones de CentOS 7, localizamos el registro de accesos en el **archivo secure**. Para verificar las **conexiones ssh activas**, ejecuta el siguiente comando.

```
cat /var/log/secure
```

Como puedes ver en el ejemplo, también aparecen los errores de autenticación producidos por intentos de acceso ajenos al sistema.

```
[root@sololinux-demo log]# cat secure
Nov 15 02:31:01 sololinux-demo sshd[30121]: Invalid user
spider from 118.97.16.127 port 40270
Nov 15 02:31:01 sololinux-demo sshd[30121]:
pam_unix(sshd:auth): check pass; user unknown
Nov 15 02:31:01 sololinux-demo sshd[30121]:
pam_unix(sshd:auth): authentication failure; logname=
uid=0 euid=0 tty=ssh ruser= rhost=118.97.16.127
Nov 15 02:31:03 sololinux-demo sshd[30121]: Failed
password for invalid user spider from 118.97.16.127 port
40270 ssh2
Nov 15 02:31:03 sololinux-demo sshd[30121]: Received
disconnect from 118.97.16.127 port 40270:11: Bye Bye
[preauth]
Nov 15 02:31:03 sololinux-demo sshd[30121]: Disconnected
from invalid user spider 118.97.16.127 port 40270
[preauth]
Nov 15 02:32:14 sololinux-demo sshd[30123]:
pam_unix(sshd:auth): authentication failure; logname=
uid=0 euid=0 tty=ssh ruser= rhost=118.97.16.127 user=root
Nov 15 02:32:16 sololinux-demo sshd[30123]: Failed
password for root from 118.97.16.127 port 53126 ssh2
Nov 15 02:32:18 sololinux-demo sshd[30123]: Received
disconnect from 118.97.16.127 port 53126:11: Bye Bye
[preauth]
Nov 15 02:32:18 sololinux-demo sshd[30123]: Disconnected
from authenticating user root 118.97.16.127 port 53126
[preauth]
Nov 15 02:33:32 sololinux-demo sshd[30125]: Invalid user
lsfadmin from 118.97.16.127 port 37738
Nov 15 02:33:32 sololinux-demo sshd[30125]:
pam_unix(sshd:auth): check pass; user unknown
Nov 15 02:33:32 sololinux-demo sshd[30125]:
pam_unix(sshd:auth): authentication failure; logname=
uid=0 euid=0 tty=ssh ruser= rhost=118.97.16.127
Nov 15 02:33:34 sololinux-demo sshd[30125]: Failed
password for invalid user lsfadmin from 118.97.16.127 port
37738 ssh2
Nov 15 02:33:34 sololinux-demo sshd[30125]: Received
disconnect from 118.97.16.127 port 37738:11: Bye Bye
[preauth]
Nov 15 02:33:34 sololinux-demo sshd[30125]: Disconnected
from invalid user lsfadmin 118.97.16.127 port 37738
[preauth]
Nov 15 02:34:44 sololinux-demo sshd[30128]: Invalid user
rh from 118.97.16.127 port 50598
Nov 15 02:34:44 sololinux-demo sshd[30128]:
pam_unix(sshd:auth): check pass; user unknown
Nov 15 02:34:44 sololinux-demo sshd[30128]:
pam_unix(sshd:auth): authentication failure; logname=
uid=0 euid=0 tty=ssh ruser= rhost=118.97.16.127
```

El registro en tiempo real

Puedes monitorear el registro en tiempo real con el **comando tail**.

```
tail -f -n 50 /var/log/secure | grep sshd
```

Ejemplo de salida...

```
[root@sololinux-demo log]# tail -f -n 50
/var/log/secure | grep sshd
Nov 16 15:15:08 sololinux-demo sshd[35427]: Failed
password for invalid user qazwsxedc!@# from
119.28.178.61 port 45004 ssh2
Nov 16 15:15:08 sololinux-demo sshd[35427]:
Received disconnect from 119.28.178.61 port
45004:11: Bye Bye [preauth]
Nov 16 15:15:08 sololinux-demo sshd[35427]:
Disconnected from invalid user qazwsxedc!@#
119.28.178.61 port 45004 [preauth]
Nov 16 15:15:25 sololinux-demo sshd[35429]: Invalid
user Q.1 from 206.189.43.92 port 58904
Nov 16 15:15:25 sololinux-demo sshd[35429]:
pam_unix(sshd:auth): check pass; user unknown
Nov 16 15:15:25 sololinux-demo sshd[35429]:
pam_unix(sshd:auth): authentication failure;
logname= uid=0 euid=0 tty=ssh ruser=
rhost=206.189.43.92
Nov 16 15:15:27 sololinux-demo sshd[35429]: Failed
password for invalid user Q.1 from 206.189.43.92
port 58904 ssh2
Nov 16 15:15:28 sololinux-demo sshd[35429]:
Received disconnect from 206.189.43.92 port
58904:11: Bye Bye [preauth]
Nov 16 15:15:28 sololinux-demo sshd[35429]:
Disconnected from invalid user Q.1 206.189.43.92
port 58904 [preauth]
Nov 16 15:15:45 sololinux-demo sshd[35431]: Invalid
user lqaz2wsx3edc1 from 105.163.182.191 port 44174
Nov 16 15:15:45 sololinux-demo sshd[35431]:
pam_unix(sshd:auth): check pass; user unknown
Nov 16 15:15:45 sololinux-demo sshd[35431]:
pam_unix(sshd:auth): authentication failure;
logname= uid=0 euid=0 tty=ssh ruser=
rhost=105.163.182.191
Nov 16 15:15:47 sololinux-demo sshd[35431]: Failed
password for invalid user lqaz2wsx3edc1 from
105.163.182.191 port 44174 ssh2
Nov 16 15:15:49 sololinux-demo sshd[35431]:
Received disconnect from 105.163.182.191 port
44174:11: Bye Bye [preauth]
Nov 16 15:15:49 sololinux-demo sshd[35431]:
Disconnected from invalid user lqaz2wsx3edc1
105.163.182.191 port 44174 [preauth]
Nov 16 15:17:02 sololinux-demo sshd[35434]: Invalid
user Changeme! from 200.170.248.190 port 55992
Nov 16 15:17:02 sololinux-demo sshd[35434]:
pam_unix(sshd:auth): check pass; user unknown
Nov 16 15:17:02 sololinux-demo sshd[35434]:
pam_unix(sshd:auth): authentication failure;
logname= uid=0 euid=0 tty=ssh ruser=
rhost=200.170.248.190
Nov 16 15:17:03 sololinux-demo sshd[35434]: Failed
password for invalid user Changeme! from
200.170.248.190 port 55992 ssh2
Nov 16 15:17:04 sololinux-demo sshd[35434]:
Received disconnect from 200.170.248.190 port
55992:11: Bye Bye [preauth]
Nov 16 15:17:04 sololinux-demo sshd[35434]:
Disconnected from invalid user Changeme!
200.170.248.190 port 55992 [preauth]
```

Configurar el Bluetooth en Linux



Cada día tenemos más **periféricos Bluetooth** a nuestra disposición, te puede gustar o no, pero es evidente que esta tecnología se impone en productos como ratones, teclados y auriculares. Realmente, a veces resulta difícil imaginar el uso de todos estos dispositivos mediante conexión por cable.

En muchas **distribuciones Linux**, Bluetooth ya viene configurado por defecto, otras facilitan su configuración. Muchas otras afamadas distros, requieren de una configuración básica antes de que puedas utilizar al cien por cien la **tecnología Bluetooth** en nuestra máquina.

Los pasos a seguir para configurar el **Bluetooth** en Linux, pueden variar un poco de una distribución a otra, pero como norma general suelen ser parecidos. En este artículo, vemos como configurar el Bluetooth en nuestro **sistema Linux**.



Configurar el Bluetooth en Linux

Si tu linux no tiene el Bluetooth configurado, lo primero que debes hacer es instalar el paquete «**bluez**». Este paquete contiene los controladores de adaptadores Bluetooth, así como varias utilidades en CLI que te ayudaran a configurarlo.

También es necesario un buen administrador de Bluetooth. Muchos entornos de escritorio vienen con sus propios administradores de Bluetooth integrados. Por ejemplo, KDE Plasma utiliza Bluetooth de KDE, **Gnome** también integra uno similar, etc. Si no tienes ninguno, debes añadir la herramienta **cliente bluetman**. Comenzamos.

Instalar Bluez

Debian y derivados

```
sudo apt-get install bluez*
```

Fedora y derivados

```
sudo dnf install bluez bluez-tools
```

Arch linux, Manjaro y derivados

```
sudo pacman -S bluez bluez-utils
```

Open Suse y derivados

```
# Modifica la version del repositorio si no  
# corresponde con la version instalada.  
zypper addrepo  
https://download.opensuse.org/repositories/openSUSE  
:Leap:15.2:Update/standard/  
openSUSE:Leap:15.2:Update.repo  
zypper refresh  
zypper install bluez
```

Instalar Blueman

Si tu entorno de escritorio no viene con un administrador de Bluetooth predeterminado, instala Blueman.

Debian y derivados

```
sudo apt install blueman
```

Fedora y derivados

```
sudo dnf install blueman
```

Arch linux, Manjaro y derivados

```
sudo pacman -S blueman
```

Open Suse y derivados

```
sudo zypper install blueman
```

Habilitar el bluetooth

Antes de **habilitar el bluetooth** en el sistema, debes tener presente que no es lo mismo un bluetooth integrado (actualmente la mayoría lo tienen), o utilizas un dispositivo externo (por ejemplo mediante **usb**). Si haces uso de un dispositivo ajeno al sistema, antes de continuar debes ejecutar el siguiente comando.

```
mdprobe btusb
```

Ahora, independientemente de tu dispositivo lo activamos o habilitamos. Puedes hacer esta operación desde la terminal, o desde el asistente de tu entorno de escritorio.

Habilitar el bluetooth desde consola

```
sudo systemctl start bluetooth.service  
sudo systemctl enable bluetooth.service
```

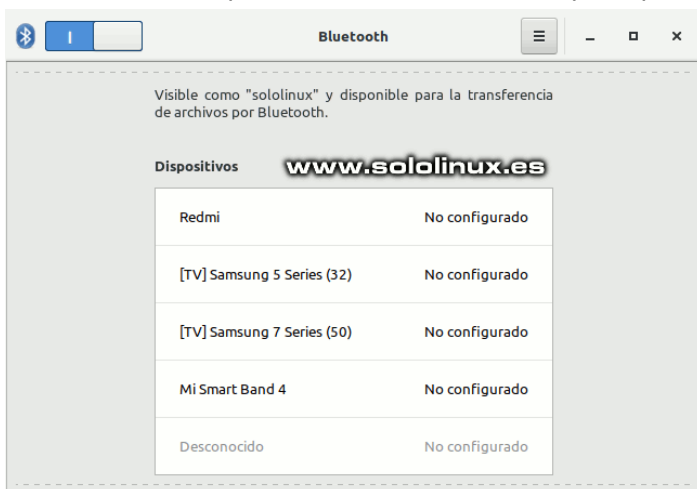

Habilitar el bluetooth desde menú de escritorio

Abres la herramienta de administración y habilitas el dispositivo.



Conectar dispositivos por Bluetooth

Actualizas los dispositivos en el administrador para que detecte las posibles conexiones.



Para conectar, pulsa sobre el que necesites y listo.



Antivirus para linux – ClamAV y ClamTK

Antivirus para linux – ClamAV y ClamTK, la combinación perfecta para los usuarios que no quieren complicarse la vida.

Linux es famoso por ser muy seguro y, tener pocos virus que realmente puedan romper el sistema. A medida que todo evoluciona y pasa el tiempo, es cierto, que cada día conocemos nuevos troyanos, malware y otras aplicaciones dañinas que aterrizan en linux.

Muchos usuarios, sobre todo empresas con servidores de alta producción en Linux, comienzan a preocuparse por este tema. Si eres prudente y tomas las debidas precauciones y medidas de seguridad, no deberías tener ningún problema en **linux**, aun siendo así, todos buscamos herramientas antivirus para **proteger nuestro/os sistemas de escritorio o servidores.**

ClamAV es un poderoso motor antivirus de código abierto, capaz de detectar virus, troyanos y otros **malwares**. También los puede eliminar o aislar. Este antivirus tiene una base escrita para trabajar en consola, esto puede complicar su acceso a los usuarios más novatos en linux; por suerte existe una interfaz gráfica que soluciona este problema (si es tu caso). Hablamos de **ClamTK**.

En sus orígenes, ClamAV fue diseñado como puerta de enlace para el escaneo de **correo electrónico** en **Unix**. Con el paso de los años, poco a poco, ha evolucionado hasta convertirse en una referencia como solución a los códigos malignos en Linux. En este artículo veremos como **instalar ClamAV y ClamTK** (también configurar) en nuestro linux. Antivirus para linux – ClamAV y ClamTK.

Antivirus para linux – ClamAV y ClamTK

Antes de **instalar ClamTK**, te recomiendo encarecidamente que instales ClamAV y lo actualices.



Instalar ClamAV

En Debian, Ubuntu, Linux Mint y derivados

```
sudo apt update
sudo apt install clamav
sudo freshclam
```

En CentOS 7, RHEL 7 y derivados

```
sudo yum install -y epel-release
sudo yum install -y clamav
sudo freshclam
```

Fedora y derivados

```
sudo dnf install -y clamav clamav-update
sudo freshclam
```

Arch linux, Manjaro y derivados

```
sudo pacman -S clamav
sudo freshclam
```

Instalar ClamTK

Ahora instalamos la interfaz gráfica de ClamAV, ClamTK.

En Debian, Ubuntu, Linux Mint y derivados

```
sudo apt install clamtk
```

En CentOS 7, RHEL 7 y derivados

```
wget
https://bitbucket.org/davem_/clamtk-gtk3/downloads/
clamtk-6.06-1.el7.noarch.rpm
yum localinstall clamtk-6.06-1.el7.noarch.rpm
```

Fedora y derivados

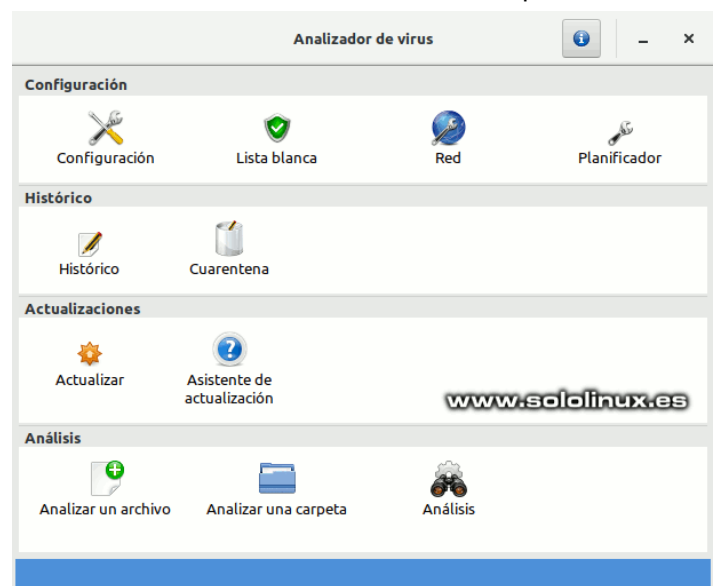
```
sudo dnf install -y clamtk
```

Arch linux, Manjaro y derivados

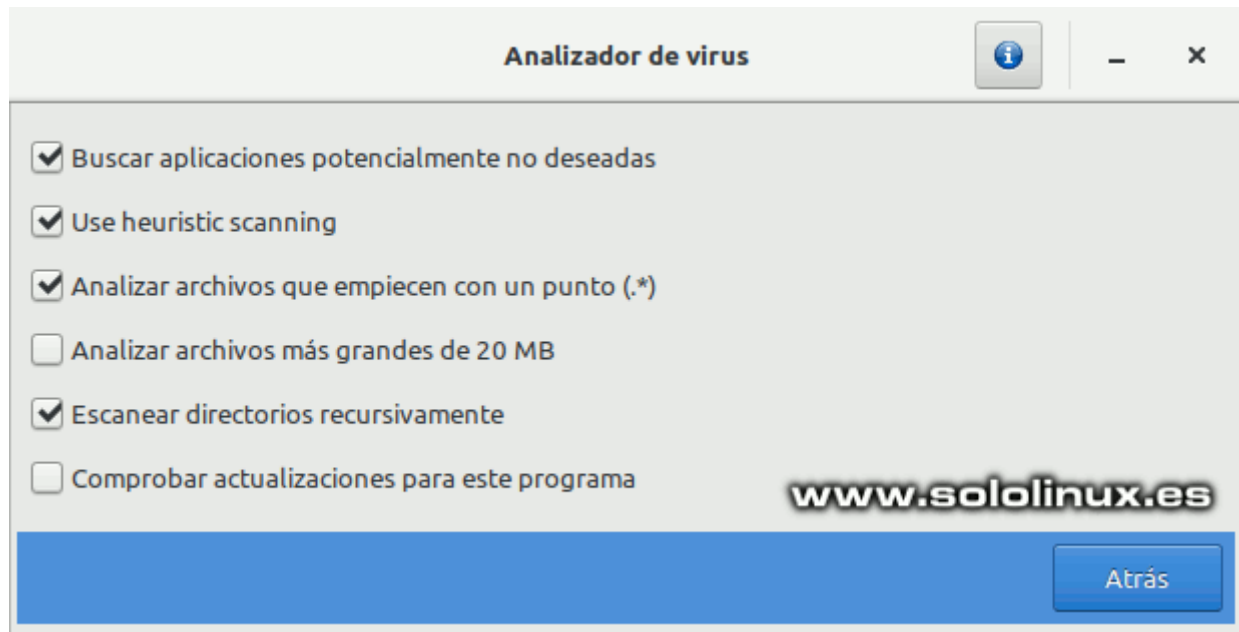
```
sudo pacman -S clamtk
```

Configurar ClamAV

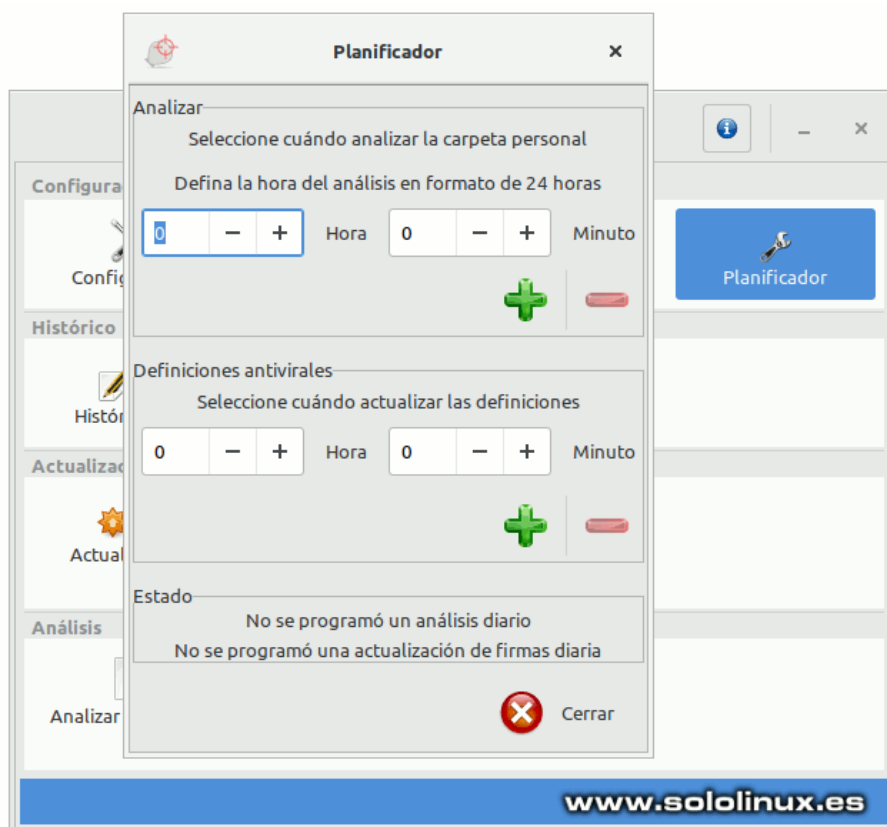
Una vez termine la instalación de ClamTK, puedes lanzar la herramienta desde tu menú de aplicaciones.



Antes de continuar, te recomiendo configurar como quieres que opere el antivirus.



La interfaz de ClamTK, es clara y concisa. Muy fácil de usar a la vez que completa, incluso permite programar las actualizaciones y, los análisis diarios (si tú lo requieres).



ClamAV + ClamTK forman una pareja excelente, pero si eres de los que prefieres operar en consola/terminal, revisa este anterior artículo.

- <https://www.sololinux.es/instalar-y-configurar-clamav-en-centos-7/>

Como instalar Nextcloud en Ubuntu 20.04



Nextcloud es una plataforma para compartir archivos de **código abierto**, que nos permite crear nuestro propio servicio de alojamiento de archivos. Con una forma de operar muy similar a **Dropbox** o **Google Drive**, realmente es una fantástica bifurcación de **ownCloud**.

Al instalar Nextcloud en un **servidor o vps**, podremos almacenar contenidos, como archivos privados, documentos, fotos, música, vídeos, etc. Es evidente que los podemos compartir con amigos y familiares. Además, destacamos que la herramienta viene con muchos complementos para ampliar sus funciones. En este artículo, veremos como montar tu propio **servidor Nextcloud** en un servidor o vps con **Ubuntu 20.04**.



Como instalar Nextcloud en Ubuntu 20.04

Comenzamos actualizando el sistema base con los últimos paquetes.

```
apt-get update -y
```

Instalar Lamp en Ubuntu 20.04

Una vez actualizado, procedemos a instalar Lamp en Ubuntu 20.04.

```
sudo apt install apache2 mariadb-server libapache2-mod-php php php-gmp php-bcmath php-gd php-json php-mysql php-curl php-mbstring php-intl php-imagick php-xml php-zip bzip2 -y
```

Editamos el archivo php.ini.

```
nano /etc/php/7.4/apache2/php.ini
```

Modifica el límite de memoria, y la situación geográfica de tu servidor (si es necesario).

```
*
memory_limit = 512M
*
date.timezone = Europe/Madrid
```

Guarda el archivo y cierra el editor. Es necesario reiniciar Apache.

```
systemctl restart apache2
```

Crear la base de datos Nextcloud

Conectamos a la consola de MariaDB con el siguiente comando.

```
mysql
```

```
root@sololinux-demo:~# mysql
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 49
Server version: 10.3.25-MariaDB-0ubuntu0.20.04.1
Ubuntu 20.04
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]>
```

Ahora creamos la base de datos «**nextclouddb**».

```
create database nextclouddb;
```

Agregamos el usuario «**nextcloud**» y la «**password**».

```
grant all on nextclouddb.* to
'nextcloud'@'localhost' identified by 'mipassword';
```

Eliminamos los privilegios y salimos de la consola de MariaDB.

```
flush privileges;
exit;
```

Descargar Nextcloud

Te recomiendo que revises la página de descargas **oficial**, para verificar la última versión instalada (en nuestro caso la 20).

```
wget
https://download.nextcloud.com/server/releases/latest-20.tar.bz2
```

Extraemos el archivo descargado.

```
tar -jxvf latest-20.tar.bz2
```

Movemos la aplicación a la raíz web de Apache y, concedemos los permisos requeridos.

```
mv nextcloud /var/www/html/
chown -R www-data:www-data /var/www/html/nextcloud
chmod -R 775 /var/www/html/nextcloud
```

Como instalar Nextcloud en Ubuntu 20.04

Configurar Apache para Nextcloud

Crearemos el virtual host.

```
nano /etc/apache2/sites-available/nextcloud.conf
```

Copia y pega lo siguiente (no olvides insertar tu dominio).

```
nano /etc<VirtualHost *:80>
DocumentRoot "/var/www/html/nextcloud"
ServerName midominio.com
ErrorLog ${APACHE_LOG_DIR}/nextcloud.error
CustomLog ${APACHE_LOG_DIR}/nextcloud.access
combined
<Directory /var/www/html/nextcloud/>
Require all granted
Options FollowSymLinks MultiViews
AllowOverride All
<IfModule mod_dav.c>
Dav off
</IfModule>
SetEnv HOME /var/www/html/nextcloud
SetEnv HTTP_HOME /var/www/html/nextcloud
Satisfy Any
</Directory>
</VirtualHost>apache2/sites-available/
nextcloud.conf
```

Guarda el archivo y cierra el editor.

Ahora necesitas habilitar el virtual host de Apache, junto con los módulos necesarios.

```
a2ensite nextcloud.conf
a2enmod rewrite headers env dir mime setenvif ssl
```

Para concluir la instalación de un servidor Nextcloud, solo falta reiniciar Apache. Instalar Nex

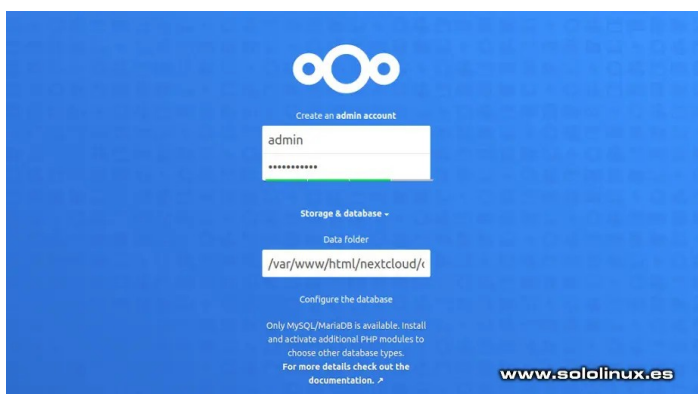
```
systemctl restart apache2
```

Acceder a Nextcloud

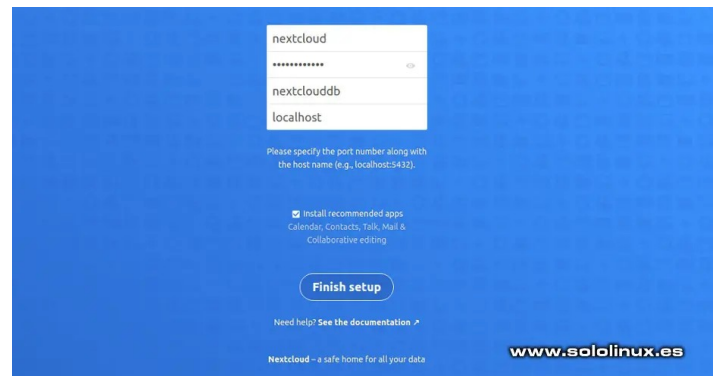
Para configurar la herramienta, introduces la url del servicio en tu **navegador web** favorito. Instalar **Nextcloud**.

http://midominio.com

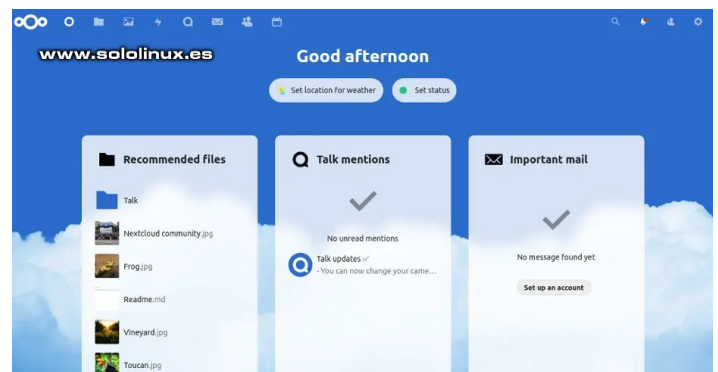
En la pantalla que nos aparece creamos nuestra cuenta admin.



En la siguiente pantalla insertamos el usuario y el password de la base de datos.



Se inicia nuestro **Nextcloud**. Veremos una pantalla similar a...

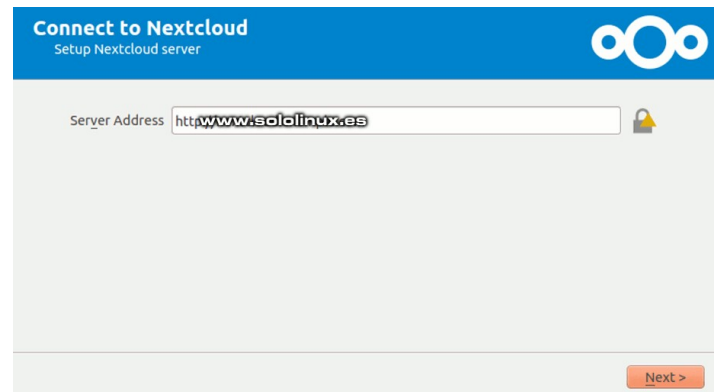
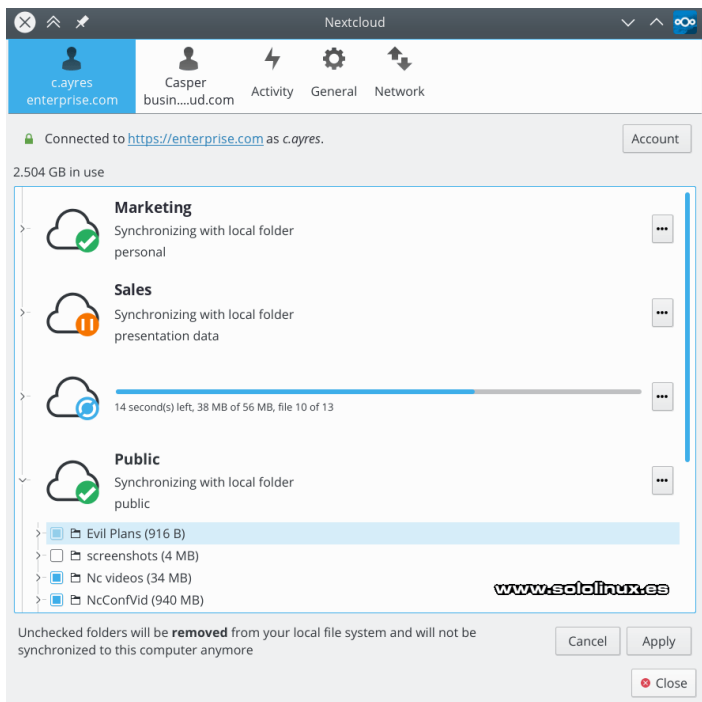


En el próximo artículo aprenderemos a instalar Nextcloud cliente, en nuestro sistema operativo linux.

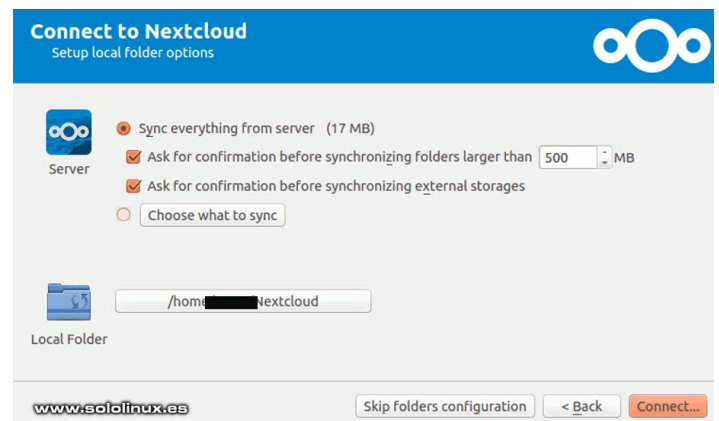
Instalar Nextcloud Client en Ubuntu 20.04

En el anterior artículo, aprendimos a **instalar Nextcloud en un servidor Ubuntu 20.04**. Hoy agregamos el cliente Nextcloud en nuestro sistema de escritorio y, nos conectaremos al servidor Nextcloud. Esto nos permitirá guardar y compartir de forma sencilla nuestro contenido con quienes queramos

Nextcloud ofrece aplicaciones cliente para Android, Windows, iOS y Linux. Nextcloud Client nos permite conectar con nuestro servidor remoto y copiar, pegar o sincronizar todos los archivos y datos que nos interesen.



Te solicita el usuario y el password de acceso. Por último veremos una pantalla donde debes seleccionar tu configuración preferida.



Pulsas en conectar para acceder al servidor, fácil, fácil.

Instalar Nextcloud Client en Ubuntu 20.04

Instalamos los requisitos previos.

```
sudo apt install software-properties-common gnupg2 -y
```

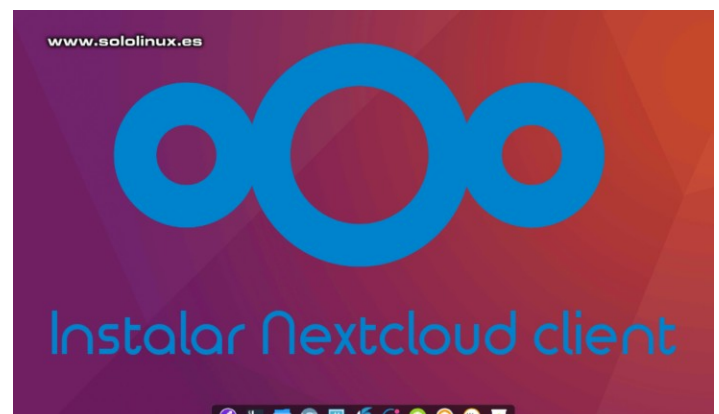
Ahora agregamos el repositorio correspondiente y actualizamos los paquetes.

```
sudo add-apt-repository ppa:nextcloud-devs/client
sudo apt update -y
```

Solo falta instalar Nextcloud Client.

```
apt-get install nextcloud-client -y
```

Inicias la herramienta desde tu menú de aplicaciones. Aparece una pantalla donde debes insertar la url o ip de tu **servidor Nexrcloud**.



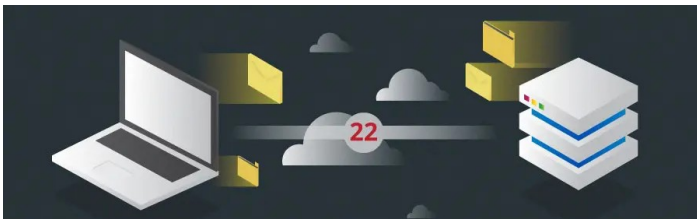
Conectar por SSH incluyendo el password



SSH es la forma preferida por los **administradores de sistemas**, para acceder a los servidores Linux u otros sistemas como firewalls, enrutadores y otros dispositivos. En este artículo de hoy, aprenderemos a **conectar por ssh** con una sola línea que contenga los datos necesarios para que la conexión sea válida.

Para lograr nuestro objetivo nos ayudamos de la herramienta **sshpass**. Esta utilidad fue diseñada para ejecutar **ssh** con lo que se conoce como autenticación de contraseña interactiva con teclado, pero en modo no interactivo. **Sshpass** ejecuta ssh en un tty dedicado, engañando al sistema de manera que este crea que el password lo introduce un humano mediante un teclado físico.

Si manejas muchas máquinas, el ahorro de tiempo puede ser enorme. También es válido para incluirlo en scripts y otros tipos de herramientas o aplicaciones.



Conectar por SSH incluyendo el password
La **utilidad sshpass** no viene instalada por defecto en casi ninguna **distribución linux**, así que procedemos a ello.

En Debian, Ubuntu, Linux Mint y derivados:

```
sudo apt install sshpass
```

```
root@demo:/home/sololinux-demo# apt install sshpass
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  sshpass
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 10,5 kB de archivos.
Se utilizarán 30,7 kB de espacio de disco adicional después de esta operación.
Des:1 http://ftp.byfly.by/ubuntu bionic/universe amd64 sshpass amd64 1.06-1 [10,5 kB]
Descargados 10,5 kB en 0s (33,7 kB/s)
Seleccionando el paquete sshpass previamente no seleccionado.
(Leyendo la base de datos ... 341651 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../sshpass_1.06-1_amd64.deb ...
Desempaquetando sshpass (1.06-1) ...
Configurando sshpass (1.06-1) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
```

En Fedora y derivados:

```
sudo dnf install sshpass
```

En CentOS y derivados:

```
sudo yum install epel-release
sudo yum update
sudo yum install sshpass
```

En OpenSuse, Suse y derivados:

```
sudo zypper install sshpass
```

sudo zypper install sshpass

```
sudo zypper install sshpass
```

Una vez instalada la herramienta, usamos la siguiente sintaxis para conectar con nuestro servidor. No te olvides de usar comillas simples en la contraseña.

```
# Con usuario
sshpass -p 'password' ssh usuario@dominio-o-ip
```

```
# Como root
sshpass -p 'password' ssh ip-del-server
```

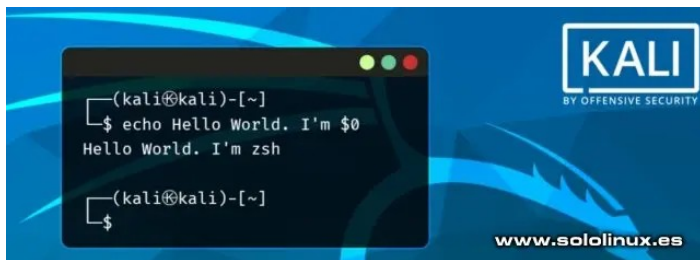
Es así de fácil. Vemos un ejemplo...

```
root@sololinux:/home/sergio# sshpass -p 'az%fg
%gHjm' ssh 185.195.27.172
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0
x86_64)
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
Last login: Thu Nov 19 11:01:17 2020 from
62.56.112.127
```

El nuevo Kali Linux 2020.4 se pasa a ZSH

Kali Linux 2020.4 fue lanzado hace apenas 48 horas por **Offensive Security**, y viene con una gran sorpresa. Kali Linux abandona la **shell bash** y se pasa a **ZSH**.

El nuevo Kali Linux 2020.4, viene con muchos paquetes de software y herramientas que permiten a los profesionales de la seguridad, realizar sus pruebas y auditorías. Está claro que los piratas informáticos también se aprovechan de esta excelente **distribución linux**.



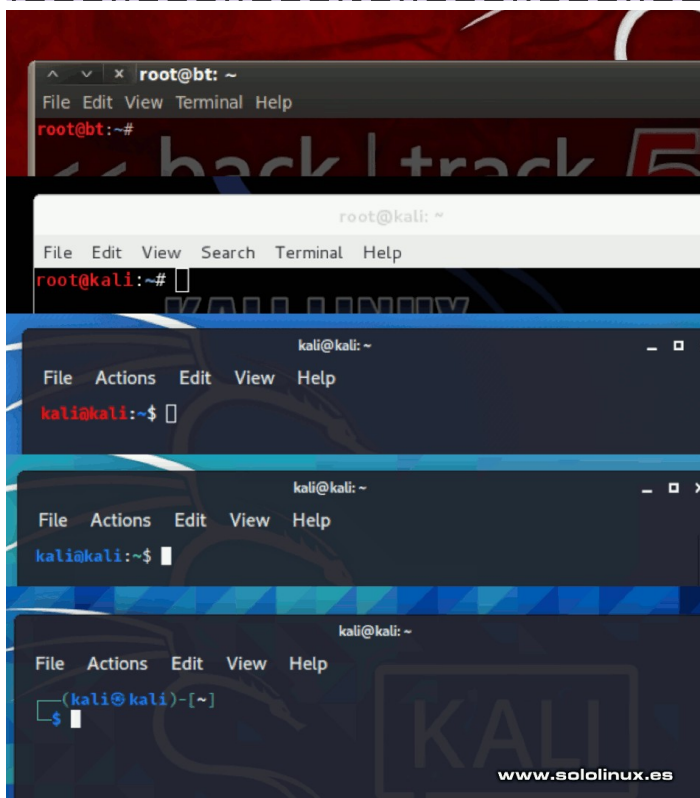
El nuevo Kali Linux 2020.4 se pasa a ZSH

En esta versión, Kali Linux Team nos presenta varias características nuevas que pasamos a describir.

Kali Linux se pasa a la shell ZSH

En Kali Linux 2020.4, la nueva shell predeterminada es **ZSH**, para los usuarios que instalan la distribución por primera vez. Si ya tienes Kali instalado y actualizas de versión, debes hacer el cambio de forma manual. Ejecuta los siguientes comandos:

```
cp -i /etc/skel/.zshrc ~/
chsh -s /bin/zsh
```



En instalaciones nuevas lo tendrás instalado por defecto. Si actualizas la versión debes copiar el nuevo archivo «.bashrc», con estos comandos.

```
cp -i .bashrc{,.bak}
cp -i /etc/skel/.bashrc ~/
source ~/.bashrc
```

Avisos al iniciar sesión

Esta nueva versión muestra un mensaje dinámico (MOTD) al iniciar sesión. Nos ofrece sugerencias que nos ayudaran a solucionar problemas en la instalación.

Vemos un ejemplo...

```
(Message from Kali developers)

This is a minimal installation of Kali Linux, you
likely
want to install supplementary tools. Learn how:
→
https://www.kali.org/docs/troubleshooting/common-
minimum-setup/

This is a cloud installation of Kali Linux. Learn
more about
the specificities of the various cloud images:
→
https://www.kali.org/docs/troubleshooting/common-
cloud-setup/

We have kept /usr/bin/python pointing to Python 2
for backwards
compatibility. Learn how to change this and avoid
this message:
→ https://www.kali.org/docs/general-use/python3-
transition/

(Run "touch ~/.hushlogin"

(kali@kali)-[~]
$
```

Si los mensajes te resultan molestos, puedes deshabilitarlos por usuario o a nivel global.

```
# por usuario
touch ~/.hushlogin
# para todos los usuarios
touch /etc/kali-motd/disable-all
```

Lista completa de modificaciones

- **ZSH** es el nuevo shell predeterminado.
- **Cambio del tema** de la shell Bash.
- **Kali linux** se asocia **conby3b133d3r**.
- **Mensajes de ayuda** al iniciar sesión (MOTD).
- **Se incluyen guías** para empaquetar nuevas herramientas.
- **Nuevo kernel** y varias herramientas antes no incluidas. Se actualizan las existentes, además de configurar Proxchains 4 como predeterminado.
- **Actualizaciones de NetHunter** con nuevo menú de configuración.
- **Win-KeX 2.5**, modo de sesión mejorado, acerca **Win-KeX** a los dispositivos ARM.
- **Vagrant y VMware** ahora tienen soporte total.

También se incluyen las siguientes herramientas:

- [Apple blee](#)
- [CertGraph](#)
- [dnscat2](#)
- [FinalRecon](#)
- [goDoH](#)
- [hostapd-mana](#)
- [Metasploit Framework v6](#)
- [Whatmask](#)

Descargar o actualizar Kali Linux

Puedes descargar la última versión desde su página oficial.

- [Descargar Kali linux 2020.4](#)



Con los siguientes comandos podrás actualizar tu versión de Kali Linux, y cambiar la shell a ZSH.

```
echo "deb http://http.kali.org/kali kali-rolling main non-free contrib" | sudo tee /etc/apt/sources.list
sudo apt update && sudo apt -y full-upgrade
cp -i /etc/skel/.bashrc ~/
cp -i /etc/skel/.zshrc ~/
chsh -s /bin/zsh
[ -f /var/run/reboot-required ] && sudo reboot -f
```

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es



HTTPIe – El Cliente http en línea de Comandos

HTTPIe es un moderno **cliente HTTP** en línea de comandos **open source**, además de multiplataforma. Da lo mismo si usas Linux, MacOS o Windows, con esta herramienta podrás verificar y depurar las API, servidores HTTP y sus servicios web.

Este **cliente HTTP** viene con **JSON**, **HTTPS**, proxies y soporte de autenticación. Otras características destacadas son el resaltado de sintaxis, sesiones persistentes y una gran compatibilidad con extensiones. También tiene la capacidad de crear formularios y descargar o cargar archivos. Está basado en Python.



HTTPIe – El cliente http en línea de comandos

Instalamos la herramienta en nuestra distribución linux preferida.

En Debian, Ubuntu, Linux Mint y derivados:

```
sudo apt install httpie
```

En CentOS, RHEL y derivados:

```
sudo yum install httpie
```

En Fedora y derivados:

```
sudo dnf install httpie
```

En Arch Linux, Manjaro y derivados:

```
sudo pacman -S httpie
```

En Gentoo y redicados:

```
emerge httpie
```

Ahora vemos un par de ejemplos de uso del cliente http, en el primero vemos los datos de un sitio web, en el segundo descargamos un archivo vía HTTP.

Datos de la url con el comando http:

```
sergio@sololinux:~$ http google.es
HTTP/1.1 301 Moved Permanently
Cache-Control: public, max-age=2592000
Content-Length: 218
Content-Type: text/html; charset=UTF-8
Date: Sat, 21 Nov 2020 15:49:35 GMT
Expires: Mon, 21 Dec 2020 15:49:35 GMT
Location: http://www.google.es/
Server: gws
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 0
<HTML><HEAD><meta http-equiv="content-type"
content="text/html; charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.es/">here</A>.
</BODY></HTML>
```

Descargar archivos el comando http y la opción download:

```
# En el ejemplo descargamos viber para Debian o
Ubuntu.
http --download
https://download.cdn.viber.com/cdn/desktop/Linux/vi
ber.deb
```

```
root@sololinux:~# http --download https://download.cdn.viber.com/cdn/desktop/Linux/viber.deb
HTTP/1.1 200 OK
Accept-Ranges: bytes
Connection: keep-alive
Content-Length: 99693958
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Date: Sat, 21 Nov 2020 15:55:56 GMT
ETag: "d85f509084d5b3d602eafe5dc22dead"
Last-Modified: Wed, 15 Jul 2020 11:56:07 GMT
Server: AmazonS3
Via: 1.1 d4ab4520827d99658a8d233539c37425.cloudfront.net (CloudFront)
X-Amz-Cf-Id: 8hZwqEPwt83qP7Vn-W0NkvLLN07xNF0sYFJl00w26EL_DPA3B_gpEQ==
X-Amz-Cf-Pop: ZRH50-C1
X-Cache: Hit from cloudfront

Downloading 95.08 MB to "viber.deb"
Done. 95.08 MB in 34.52167s (2.75 MB/s)
```

Comandos de HTTPIe, el cliente http en línea de comandos

Ejecutamos el comando con la opción h.

```
http -h
```

```
sergio@sololinux:~$ http -h
usage: http [--json] [--form] [--pretty]
{all,colors,format,none}]
        [--style STYLE] [--print WHAT] [--
headers] [--body] [--verbose]
        [--all] [--history-print WHAT] [--
stream] [--output FILE]
        [--download] [--continue]
        [--session SESSION_NAME_OR_PATH | --
session-read-only SESSION_NAME_OR_PATH]
        [--auth USER[:PASS]] [--auth-type
{basic,digest}]
        [--proxy PROTOCOL:PROXY_URL] [--follow]
        [--max-redirects MAX_REDIRECTS] [--
timeout SECONDS]
        [--check-status] [--verify VERIFY]
        [--ssl {ssl2.3,tls1,tls1.1,tls1.2}] [--
cert CERT]
        [--cert-key CERT_KEY] [--ignore-stdin]
[--help] [--version]
        [--traceback] [--default-scheme
DEFAULT_SCHEME] [--debug]
        [METHOD] URL [REQUEST_ITEM
[REQUEST_ITEM ...]]
http: error: the following arguments are required:
URL
```

4 formas de buscar la ip de un dominio en terminal



Muchas veces queremos **averiguar la ip de un dominio**. Existen muchas herramientas y sitios web que ofrecen tal servicio, para la verdad es que no vale la pena perder el tiempo en ellas. Nuestro sistema Linux, tiene la capacidad de manera predeterminada de buscar la **IP de un dominio** forma simple y rápida.

En este artículo, veremos 4 formas de buscar la ip de un dominio en la terminal o consola de nuestro **sistema linux**, sin tener que recurrir a ningún servicio de terceros. Todos los comandos expuestos en el post admiten más de un dominio a la vez (separados por un espacio), así que más no se puede pedir.



4 formas de buscar la ip de un dominio en terminal

Vemos los cuatro comandos (vienen por defecto), que nos ofrecerán el dato que necesitamos ágilmente. Mira que facil.

Buscar la IP con host

El comando host de los más sencillos de usar, observa la sintaxis.

```
host [dominio]
```

Ejemplo...

```
sergio@sololinux:~$ host marca.es
marca.es has address 193.110.128.199
marca.es mail is handled by 10 mx1.es.easynet.net.
marca.es mail is handled by 15 mx2.es.easynet.net.
marca.es mail is handled by 5 mail.marca.es.
sergio@sololinux:~$ host www.marca.es
www.marca.es has address 193.110.128.199
```

Buscar la IP con nslookup

El **comando nslookup**, pertenece a la herramienta de administración de redes **open source** del mismo nombre.

```
nslookup [dominio]
```

Salida...

```
sergio@sololinux:~$ nslookup marca.es
Server: 127.0.0.53
Address: 127.0.0.53#53
Non-authoritative answer:
Name: marca.es
Address: 193.110.128.199
```

Buscar la IP con ping

El **comando ping** es un indiscutible en este tipo de tareas linux, aquí tampoco podía faltar agregando las opciones «**-c 3**».

```
ping -c 3 [dominio]
```

Imagen de ejemplo.

```
sergio@sololinux:~$ ping -c 3 marca.es
PING marca.es (193.110.128.199) 56(84) bytes of data.
--- marca.es ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2026ms
```

Buscar la IP con dig

El **comando dig** es de mis preferidos, pues nos aporta además de la IP, otra valiosa información.

```
dig [dominio]
```

Observa la salida de ejemplo, es mucho más completa.

```
sergio@sololinux:~$ dig marca.es
; <<>> DiG 9.11.3-lubuntu1.13-Ubuntu <<>> marca.es
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 35834
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;marca.es. IN A
;; ANSWER SECTION:
marca.es. 175 IN A 193.110.128.199
;; Query time: 0 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Sun Nov 22 13:03:46 EET 2020
;; MSG SIZE rcvd: 53
```


Ver en decimal de dos bytes

```
hexdump -d [file]
```

Aparece al similar a...

```
sergio@sololinux:~$ hexdump -d demo.sh
00000000 08483 25135 28265 25135 29537 02664
25635 28005
00000010 11887 26739 25866 26723 08303 21538
28793 08293
00000020 28537 29301 28704 29537 30579 29295
11364 28704
00000030 25964 29537 14949 02594 25970 25697
20512 21313
00000040 02643 25445 28520 08736 28505 08309
30058 29811
00000050 29728 28793 25701 08250 20516 21313
08787 25866
00000060 26723 08303 17186 25970 29793 25701
25120 08313
00000070 25939 26482 28521 11847 11842 02594
25445 28520
00000080 08736 29800 28788 14963 12079 30583
11895 28531
00000090 28524 26988 30062 11896 29541 02594
25445 28520
000000a0 08736 11580 11565 11565 11565 11565
11565 11565
000000b0 11565 11565 11565 11565 11565 11565
11565 11565
*
000000d0 11565 11565 11565 11565 11565 11565
15917 00034
000000df
```

Hexadecimal de dos bytes

```
hexdump -x [file]
```

```
sergio@sololinux:~$ hexdump -x demo.sh
00000000 2123 622f 6e69 622f 7361 0a68
6423 6d65
00000010 2e6f 6873 650a 6863 206f 5422
7079 2065
00000020 6f79 7275 7020 7361 7773 726f
2c64 7020
00000030 656c 7361 3a65 0a22 6572 6461
5020 5341
00000040 0a53 6365 6f68 2220 6f59 2075
756a 7473
00000050 7420 7079 6465 203a 5024 5341
2253 650a
00000060 6863 206f 4322 6572 7461 6465
6220 2079
00000070 6553 6772 6f69 2e47 2e42 0a22
6365 6f68
00000080 2220 7468 7074 3a73 2f2f 7777
2e77 6f73
00000090 6f6c 696c 756e 2e78 7365 0a22
6365 6f68
000000a0 2220 2d3c 2d2d 2d2d 2d2d 2d2d
2d2d 2d2d
000000b0 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d
2d2d 2d2d
*
000000d0 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d
3e2d 0022
000000df
```

Visualizar en octal de dos bytes

```
hexdump -o [file]
```

Impresión en pantalla del comando hexdump con la opción -o.

```
sergio@sololinux:~$ hexdump -o demo.sh
00000000 020443 061057 067151 061057 071541 005150
062043 066545
00000010 027157 064163 062412 064143 020157 052042
070171 020145
00000020 067571 071165 070040 071541 073563 071157
026144 070040
00000030 062554 071541 035145 005042 062562 062141
050040 051501
00000040 005123 061545 067550 021040 067531 020165
072552 072163
00000050 072040 070171 062145 020072 050044 051501
021123 062412
00000060 064143 020157 041442 062562 072141 062145
061040 020171
00000070 062523 063562 067551 027107 027102 005042
061545 067550
00000080 021040 072150 070164 035163 027457 073567
027167 067563
00000090 067554 064554 072556 027170 071545 005042
061545 067550
000000a0 021040 026474 026455 026455 026455 026455
026455 026455
000000b0 026455 026455 026455 026455 026455 026455
026455 026455
*
000000d0 026455 026455 026455 026455 026455 026455
037055 000042
000000df
```

Limitar los bytes

El **comando hexdump** admite la opción de especificar los bytes de un archivo a hexdump. Para ello usaremos la opción «-s», seguida de la cantidad de bytes deseados.

```
hexdump -s 2 [file]
```

```
sergio@sololinux:~$ hexdump -s 2 demo.sh
00000002 622f 6e69 622f 7361 0a68 6423 6d65 2e6f
00000012 6873 650a 6863 206f 5422 7079 2065 6f79
00000022 7275 7020 7361 7773 726f 2c64 7020 656c
00000032 7361 3a65 0a22 6572 6461 5020 5341 0a53
00000042 6365 6f68 2220 6f59 2075 756a 7473 7420
00000052 7079 6465 203a 5024 5341 2253 650a 6863
00000062 206f 4322 6572 7461 6465 6220 2079 6553
00000072 6772 6f69 2e47 2e42 0a22 6365 6f68 2220
00000082 7468 7074 3a73 2f2f 7777 2e77 6f73 6f6c
00000092 696c 756e 2e78 7365 0a22 6365 6f68 2220
000000a2 2d3c 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d
000000b2 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d
*
000000d2 2d2d 2d2d 2d2d 2d2d 2d2d 3e2d 0022
000000df
```

Manual de hexdump

Los ejemplos vistos en este artículo son los más habituales, puedes ampliar información en su manual. Ejecuta el siguiente comando.

```
man hexdump
```

Como usar zip en linux

www.sololinux.es



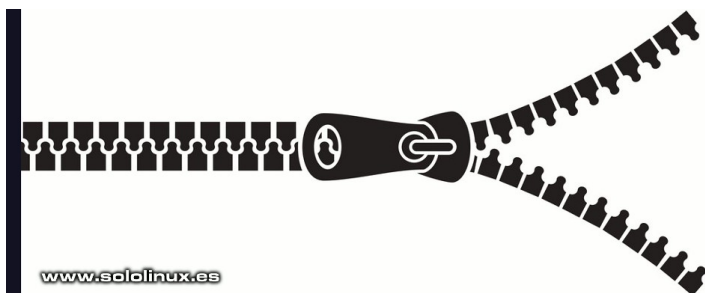
Cómo usar zip en linux

Zip es un comando multiplataforma, que se utiliza para comprimir o descomprimir archivos. Esta popular herramienta de compresión, nos ahorra espacio al reducir el tamaño de los archivos o datos.

La utilidad del **comando zip**, está fuera de toda duda. Por ejemplo:

- Ahorramos espacio de almacenamiento, ya que al comprimir reducimos el tamaño del archivo.
- Facilita la tarea de transferir archivos grandes por correo electrónico.
- Subir o descargar archivos a mucho más rápido.
- Economizamos nuestro ancho de banda.

En este artículo que hoy cubrimos, vemos la mayoría de situaciones que puedes encontrar en un entorno real, donde el **uso de zip** te ayudara en tu labor. Seguro que algunas no las conocías (como usar zip en linux).



Como usar zip en linux

Usar zip en linux de manera eficiente, no es tarea difícil. Como es lógico, el único requisito es tener la herramienta instalada, aunque la mayoría de distribuciones modernas ya la traen por defecto. Comenzamos el artículo (como usar zip en linux) que se basara en sintaxis y ejemplos de uso, verás que fácil.

Comprimir un archivo en zip

```
zip [destino] [origen]
```

Ejemplo...

```
sergio@sololinux:~$ zip demo.zip demo.sh
adding: demo.sh (deflated 39%)
```

Con el comando anterior, el archivo original se conserva.

Comprimir varios archivos en un zip

```
zip [destino] [origen] [origen1] [origen2]
```

Vemos un ejemplo real.

```
sergio@sololinux:~$ zip demo.zip demo.sh demo1.sh demo2.sh
adding: demo.sh (deflated 39%)
adding: demo1.sh (stored 0%)
adding: demo2.sh (deflated 72%)
```

También puedes aplicar un asterisco y comprimir todos los archivos con la misma extensión.

```
zip demo.zip *.sh
```

Comprimir en un directorio diferente

```
zip Documentos/archivos/demo.zip demo.sh
```

Agregar archivo a un zip existente

Para lograr este efecto, debes utilizar la opción -u.

```
zip -u demo.zip demo3.sh
```

Ejemplo de uso de zip, para agregar datos a un zip ya creado.

```
sergio@sololinux:~$ zip -u demo.zip demo3.sh
adding: demo3.sh (deflated 71%)
```

Eliminar archivo de un zip

Con la opción -d, podemos borrar un archivo de un zip ya construido

```
zip -d demo.zip demo3.sh
```

Observa...

```
sergio@sololinux:~$ zip -d demo.zip demo3.sh
deleting: demo3.sh
```

Comprimir en zip sin avisos

Si quieres evitar que se impriman informes de resultados, usa la opción -q.

```
sergio@sololinux:~$ zip -q demo.zip demo.sh
```

Eliminar archivos de origen al comprimir

Si no quieres conservar los archivos de origen una vez generada la compresión, puedes utilizar la opción **-m**. Como usar zip en linux.

```
sergio@sololinux:~$ zip -m demo.zip demo.sh
```

Comprimir un directorio y todo su contenido

```
zip -r [destino] [origen/]
```

Imagen de ejemplo...

```
sergio@sololinux:~$ zip -r demo.zip demo/
adding: demo/ (stored 0%)
adding: demo/demo2.sh (deflated 72%)
adding: demo/demo3.sh (deflated 71%)
adding: demo/demo1.sh (stored 0%)
adding: demo/demo.sh (deflated 39%)
sergio@sololinux:~$
```

Establecer el nivel de compresión

Los valores de compresión tienen una escala del 0 al 9. Por defecto es 6. Mira que fácil.

```
sergio@sololinux:~$ zip -8 demo.zip demo.sh
```

```
sergio@sololinux:~$ zip -8 demo.zip demo.sh
updating: demo.sh (deflated 39%)
```

Crear un zip protegido con password

Para proteger un archivo tan solo debes aplicar la opción **-e**, y escribir el password deseado. Como usar zip.

```
zip -e demo.zip demo.sh
```

Verás algo similar a...

```
sergio@sololinux:~$ zip -e demo.zip demo.sh
Enter password:
Verify password:
updating: demo.sh (deflated 39%)
```

Proteger un zip existente

Cuando hablamos de usar zip, es posible que nos asalte una duda. ¿Es posible agregar una contraseña a un zip ya creado?. La respuesta es **zipcloak**, si se puede.

```
zipcloak [archivo-zip]v
```

Así funciona...

```
sergio@sololinux:~$ zipcloak demo.zip
Enter password:
Verify password:
copying: demo.sh
encrypting: demo1.sh
encrypting: demo2.sh
encrypting: demo/
encrypting: demo/demo2.sh
encrypting: demo/demo3.sh
encrypting: demo/demo1.sh
encrypting: demo/demo.sh
```

Información detallada del archivo comprimido

Es posible que en casos puntuales necesites averiguar información detallada del archivo. Ejecuta el siguiente comando. Como usar zip en linux.

```
zipdetails demo.zip
```

Imagen de ejemplo de zipdetails.

```
sergio@sololinux:~$ zipdetails demo.zip
0000 LOCAL HEADER #1      04034B50
0004 Extract Zip Spec     14 '2.0'
0005 Extract OS           00 'MS-DOS'
0006 General Purpose Flag 0009
      [Bit 0]              1 'Encryption'
      [Bits 1-2]           1 'Maximum Compression'
      [Bit 3]              1 'Streamed'
0008 Compression Method   0008 'Deflated'
000A Last Mod Time        51779C63 'Mon Nov 23 19:35:06 2020'
000E CRC                  D5474540
0012 Compressed Length    00000094
0016 Uncompressed Length  000000DF
001A Filename Length      0007
001C Extra Length         001C
001E Filename             'demo.sh'
0025 Extra ID #0001       5455 'UT: Extended Timestamp'
0027 Length               0009
0029 Flags                 '03 mod access'
002A Mod Time             5FBBF2CA 'Mon Nov 23 19:35:06 2020'
002E Access Time          5FBBF2D9 'Mon Nov 23 19:35:21 2020'
0032 Extra ID #0002       7875 'ux: Unix Extra Type 3'
0034 Length               000B
0036 Version              01
0037 UID Size             04
0038 UID                  000003E8
003C GID Size             04
003D GID                  000003E8
0041 PAYLOAD              .s.....rqIxb}.....e....?'.....&.
.....&/.....k;.X.....p.e0/6.E...D.<&.e
...?[.].~Y..K0,..e...$....T;...q....or..
.....$.~.....0.....K.3

www.sololinux.es
```

Descomprimir archivos zip

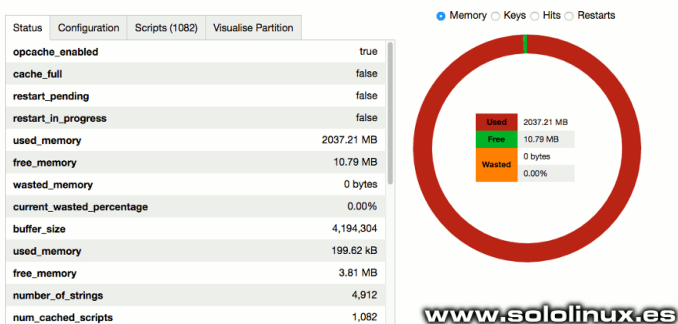
Para **descomprimir archivos zip**, necesitamos usar la herramienta **unzip**. Esto lo dejamos para un próximo artículo.

Configurar OPCache en Ubuntu 20.04

OPcache es un módulo de Apache para PHP, que se utiliza principalmente para aumentar el rendimiento de los sitios web. Su forma de operar es simple, almacena **scripts** precompilados en el espacio libre de la memoria compartida.

Al utilizar **OPcache**, el lenguaje PHP no necesita cargar e interpretar el mismo **script** cada vez que existe una nueva solicitud. Este efecto acelera el rendimiento de las aplicaciones basadas en PHP, de manera considerable.

Altamente recomendable para los que usan WordPress y otros cms, en sololinux y solowordpress lo utilizamos. Hoy veremos como configurarlo en un servidor basado en Ubuntu 20.04 o Ubuntu 18.04. Antes de habilitar y configurar opcache, debes tener instalado LAMP o LEMP, también es válido para cualquier panel de control que lo admita.



Configurar OPCache en Ubuntu 20.04

Comenzamos editando el archivo php.ini, en Apache o en Nginx.

En Apache:

```
sudo nano /etc/php/7.4/apache2/php.ini
```

En Nginx:

```
sudo nano /etc/php/7.4/fpm/php.ini
```

Para buscar con el **editor nano** «opcache», pulsa la tecla control + w, escribe «opcache» y pulsa enter. Verás algo similar a esto...

```
[opcache]
; Determines if Zend OPCache is enabled
;opcache.enable=1

; Determines if Zend OPCache is enabled for the CLI version of PHP
;opcache.enable_cli=0

; The OPcache shared memory storage size.
;opcache.memory_consumption=128

; The amount of memory for interned strings in Mbytes.
;opcache.interned_strings_buffer=8

; The maximum number of keys (scripts) in the OPcache hash table.
; Only numbers between 200 and 1000000 are allowed.
;opcache.max_accelerated_files=10000

; The maximum percentage of "wasted" memory until a restart is scheduled.
;opcache.max_wasted_percentage=5

; When this directive is enabled, the OPcache appends the current working
; directory to the script key, thus eliminating possible collisions between
; files with the same name (basename). Disabling the directive improves
; performance, but may break existing applications.
;opcache.use_cwd=1
```

www.sololinux.es

Bien... lo primero que debes hacer es **habilitar OPCache** en el php.ini. Al principio del archivo nos aparece lo siguiente...

```
[opcache]
; Determines if Zend OPCache is enabled
;opcache.enable=1
```

Lo modificamos como en el ejemplo.

```
[opcache]
; Determines if Zend OPCache is enabled
opcache.enable=1
```

Guarda el archivo con las teclas «Control+o», pero no cierras el editor. Los valores por defecto de **OPCache** son perfectos para pequeñas páginas web personales. Si tienes miles de visitas (diarias) reales, te recomiendo modificarlos. Por ejemplo...

```
.....
# Consumo de memoria
opcache.memory_consumption=356
.....
# Numero maximo de scripts
opcache.max_accelerated_files=8000
.....
# Frecuencia de actualizacion
opcache_revalidate_freq = 350
```

Ahora guardas el archivo y cierras el editor.

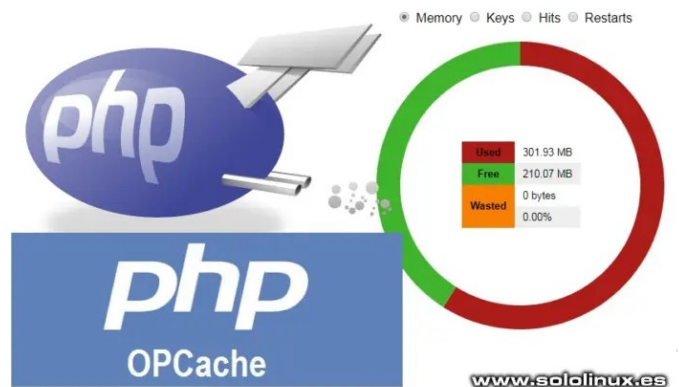
Como punto final necesitas reiniciar php.

En Apache:

```
sudo systemctl restart apache2
```

En Nginx:

```
sudo systemctl restart php7.4-fpm
sudo systemctl restart nginx
```



Extraer archivos zip Con unzip

En la mayoría de los **entornos de escritorio Linux**, descomprimir un archivo zip es tarea fácil. Simplemente, hacemos clic derecho en el archivo y aparecen las opciones incluidas. Normalmente «**Extraer aquí**» o «**Descomprimir aquí**».

Si tu **distribución linux**, no tiene una utilidad de archivos en GUI, o si prefieres utilizar otras funciones más potentes a la hora de **descomprimir archivos zip**, debes usar el **comando unzip** en la terminal del sistema. En un anterior artículo vimos el potencial del **comando zip con opciones**, hoy vemos como extraer archivos zip con unzip con funciones ampliadas.



Extraer archivos zip con unzip

La sintaxis de la herramienta unzip es comprensible y fácil de recordar.

```
unzip [opciones] archivo.zip
```

Descomprimir un zip

```
unzip demo.zip
```

```
sergio@sololinux:~$ unzip demo.zip
Archive: demo.zip
  inflating: demo.sh
  extracting: demo1.sh
  inflating: demo2.sh
  inflating: demo3.sh
```

Descomprimir un zip en un directorio definido

```
unzip demo.zip -d Documentos/demo/
```

Descomprimir todos los zip de una carpeta

```
unzip *.zip
```

```
sergio@sololinux:~$ unzip *.zip
Archive: demo.zip
  inflating: demo.sh
  extracting: demo1.sh
  inflating: demo2.sh
  inflating: demo3.sh
```

Extraer archivos sin informes en pantalla

```
unzip -q demo.zip
```

Descomprimir zip protegido por password

Para descomprimir archivos protegidos, usamos la siguiente sintaxis.

```
unzip -P [password] [archivo]
```

```
unzip -P 123456 demo.zip
```

Excluir archivo de la extracción

```
unzip demo.zip -x excluido.txt
```

Listar el contenido de un zip

```
unzip -l demo.zip
```

```
sergio@sololinux:~$ unzip -l demo.zip
Archive: demo.zip
  Length      Date    Time    Name
  -----
    223   2020-11-23  19:35   demo.sh
     42   2020-11-24  13:10   demo1.sh
    2072  2020-11-24  13:11   demo2.sh
    5729  2020-11-24  13:23   demo3.sh
  -----
    8066                                 4 files
```

Información detallada de un archivo zip

```
unzip -v demo.zip
```

```
sergio@sololinux:~$ unzip -v demo.zip
Archive: demo.zip
  Length  Method      Size  Cmpr  Date    Time    CRC-32   Name
  -----
    223   Defl:N      136   39%  2020-11-23  19:35  d5474540  demo.sh
     42   Stored       42    0%  2020-11-24  13:10  f15a14c7  demo1.sh
    2072   Defl:N      582   72%  2020-11-24  13:11  a24fd7b5  demo2.sh
    5729   Defl:N     1689   71%  2020-11-24  13:23  c030a180  demo3.sh
  -----
    8066                2449   70%                               4 files
```

Identificar errores en los archivos

```
unzip -t demo.zip
```

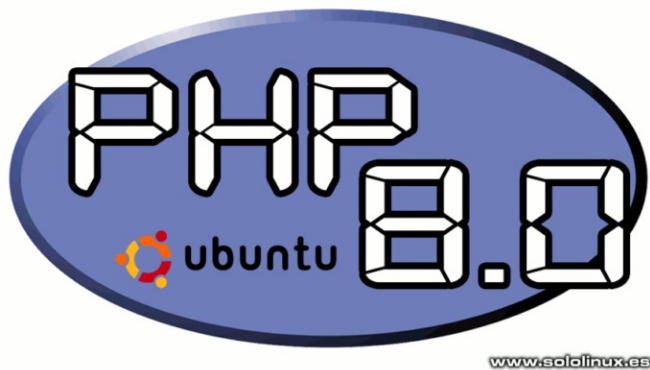
```
sergio@sololinux:~$ unzip -t demo.zip
Archive: demo.zip
  testing: demo.sh           OK
  testing: demo1.sh         OK
  testing: demo2.sh         OK
  testing: demo3.sh         OK
No errors detected in compressed data of demo.zip.
```

Manual de unzip

Para visualizar la ayuda en línea de comandos, tenemos dos opciones.

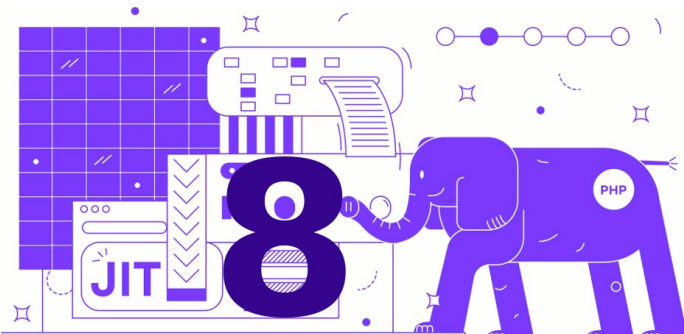
```
man unzip
# o
unzip --help
```

Instalar php 8.0 en Ubuntu 20.04



PHP 8.0 es la nueva versión principal del lenguaje de programación más extendido, si hablamos de sitios web. Nos sorprende con un montón de mejoras y nuevas características, que puedes revisar en su [anuncio oficial](#).

Este popular lenguaje es adecuado para el desarrollo web y, otras herramientas o aplicaciones. La mayoría de los sistemas de gestión de contenidos, como **WordPress, Joomla o Drupal**, están basados en **PHP**.



Instalar php 8.0 en Ubuntu 20.04

Php 8.0 es una novedad, por tanto es algo normal que no esté incluido en los **repositorios oficiales de Ubuntu**. Por suerte, el repositorio de **Ondrej** siempre lo tenemos actualizado, así que lo agregamos.

```
sudo apt update
sudo apt-get install software-properties-common
sudo add-apt-repository ppa:ondrej/php
```

Una vez agregado, actualizamos de nuevo.

```
sudo apt update
sudo apt full-upgrade
```

Ahora instalamos php 8.0 en Apache o Nginx.

En Apache:

```
sudo apt install php8.0 libapache2-mod-php8.0
```

```
root@sololinux-demo:~# sudo apt install php8.0 libapache2-mod-php8.0
Reading package lists... Done
Building dependency tree
Reading state information... Done
www.sololinux.es
The following additional packages will be installed:
  libsodium23 php-common php8.0-cli php8.0-common php8.0-opcache
  php8.0-readline
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libapache2-mod-php8.0 libsodium23 php-common php8.0 php8.0-cli php8.0-common
  php8.0-opcache php8.0-readline
0 upgraded, 8 newly installed, 0 to remove and 0 not upgraded.
Need to get 4463 kB of archives.
After this operation, 20.4 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Reinicia Apache.

```
sudo systemctl restart apache2
```

Verificamos la versión instalada con el siguiente comando...

```
php -v
```

```
root@sololinux-demo:~# php -v
PHP 8.0.0RC3 (cli) (built: Oct 31 2020 17:06:41)
( NTS )
Copyright (c) The PHP Group
Zend Engine v4.0.0-dev, Copyright (c) Zend
Technologies
    with Zend OPcache v8.0.0RC3, Copyright (c), by
    Zend Technologies
```

En Nginx:

Actualiza e instala php 8.0.

```
sudo apt update
sudo apt install nginx php8.0-fpm
```

Reinicia los servicios.

```
sudo systemctl restart nginx
sudo systemctl restart php8.0-fpm
```

También puedes verificar el servicio con...

```
sudo systemctl status php8.0-fpm
```

No te olvides que Nging no tiene soporte nativo para php, debes editar el archivo de configuración.

```
sudo nano /etc/nginx/sites-available/default
```

Más o menos debe quedar de la siguiente forma.

```
# Add index.php to the list if you are using PHP
index index.php index.html index.htm index.nginx-
debian.html;
# pass PHP scripts to FastCGI server
#
location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    #
    # With php-fpm (or other unix sockets):
    fastcgi_pass unix:/var/run/php/php8.0-fpm.sock;
    # With php-cgi (or other tcp sockets):
    # fastcgi_pass 127.0.0.1:9000;
}
# deny access to .htaccess files, if Apache's document root
# concurs with nginx's one
```

Guarda el archivo y cierra el editor. Solo nos falta reiniciar otra vez.

```
sudo systemctl reload nginx
sudo systemctl reload php8.0-fpm
```


Los mejores derivados de Arch Linux del 2021



Arch Linux es una de las distribuciones Linux más populares, su personalización y **repositorios** repletos de software de última generación, no dejan indiferente a nadie. **Arch** utiliza un modelo de lanzamiento continuo (**rolling**), lo que significa que puedes instalar el sistema y actualizarlo indefinidamente.

A pesar de todos sus beneficios, **Arch Linux** sigue siendo una distribución complicada de instalar y configurar para los profanos del **mundo linux**. Es cierto que para su instalación existen miles de tutoriales, oficiales o de clientes, para aun así... sigue asustando a muchos usuarios que quieren probarla.

La compleja instalación de **Arch**, dio lugar a una serie de distribuciones que mejoran en todo lo posible ese proceso. Ahora, los nuevos usuarios que quieren utilizar **Arch Linux**, no tienen que pasar por el calvario de su instalación y configuración.

Los mejores derivados de Arch Linux del 2021

En este artículo, vemos las que yo considero mejores derivados de Arch Linux para este año que estamos a punto de comenzar, el 2021. El orden establecido no corresponde a ninguna predilección ☐ .

EndeavourOS

Considerada por muchos como la sucesora del extinto proyecto Antergos, **EndeavourOS** es la distribución que irrumpió en el mundo linux como un elefante en una cacharrería, rompiendo moldes.

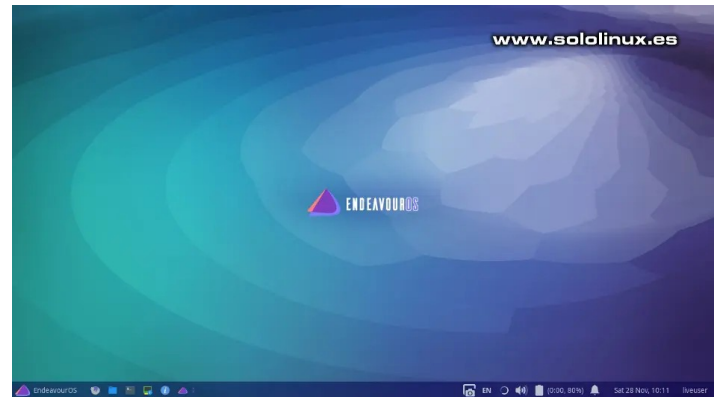
Este **derivado de Arch Linux**, intenta llegar al máximo posible de usuarios, por ello ofrece dos versiones de su instalador Calamares, online y offline. La version offline se ofrece con el escritorio XFCE y, la offline con Gnome, KDE, Deepin, Budgie, Cinnamon, Mate, i3 y LXQT.

Destacamos su trabajo por hacer el hardware Nvidia compatible al 100%. Cuando hablamos de aplicaciones, **EndeavourOS** incluye las herramientas esenciales de escritorio, para un trabajo correcto pero sin ser sobrecargada.

Una de sus mejores características es la aplicación de bienvenida, enlaces a la documentación y tareas importantes que debes tener en cuenta después de instalar uno de los **mejores derivados de Arch Linux**, como añadir aplicaciones como LibreOffice o, el **administrador de kernel akm** para cambiar de kernel si el que viene por defecto no te convence.

De momento carece de un administrador de paquetes gráfico, realmente útil. Pero me consta que los desarrolladores están en ello. Puedes **descargar EndeavourOS** en la siguiente url:

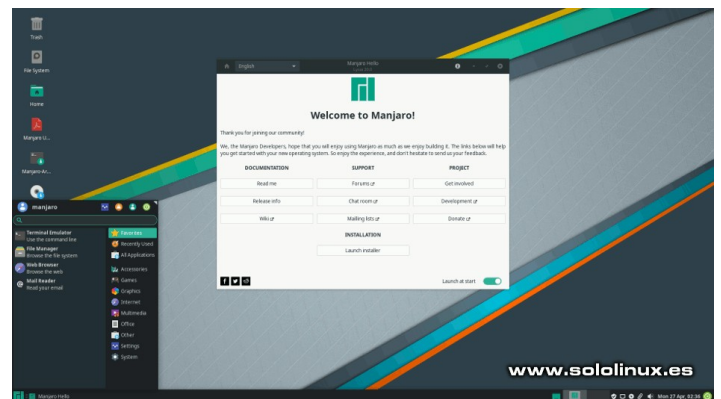
- [Descargar EndeavourOS](#)



Manjaro

Es evidente, que **Manjaro** no puede faltar en ninguna lista de mejores derivados de Arch Linux. A diferencia de este, **Manjaro** usa un elaborado **instalador Calamares** personalizado, lo que hace que sea bastante fácil instalar esta distribución. El instalador ofrece la opción de elegir entre **LibreOffice** y **FreeOffice** como suite ofimática predeterminada.

Se incluyen todas las aplicaciones habituales, como GIMP, VLC, Firefox, Thunderbird, el cliente Steam y muchos más. Puedes instalar cualquier aplicación extra, desde su administrador de paquetes Pamac, que carga los repositorios oficiales de Manjaro, así como del repositorio de usuarios AUR de Arch.



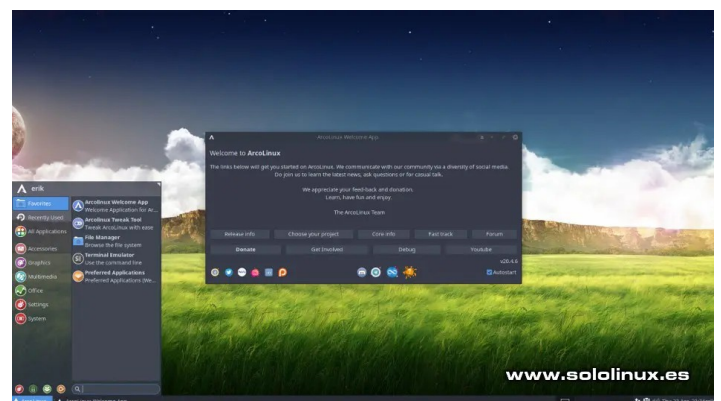
Otra ventaja que nos ofrece Manjaro son las herramientas personalizadas del proyecto, sobre todo la de detección de hardware Manjaro (MHWD), que nos instala los controladores correctos de todos los dispositivos conectados al sistema. Puedes **descargar Manjaro** desde su sitio oficial.

- [Descargar Manjaro](#)

ArcoLinux

A diferencia de otros **proyectos derivados de Arch Linux**, **ArcoLinux** se presenta como una plataforma de aprendizaje, que intenta guiar a los nuevos usuarios hasta convertirse en maestros de Arch. Por este motivo, el **proyecto ArcoLinux** crea varias distribuciones para los usuarios (dependiendo de su nivel).

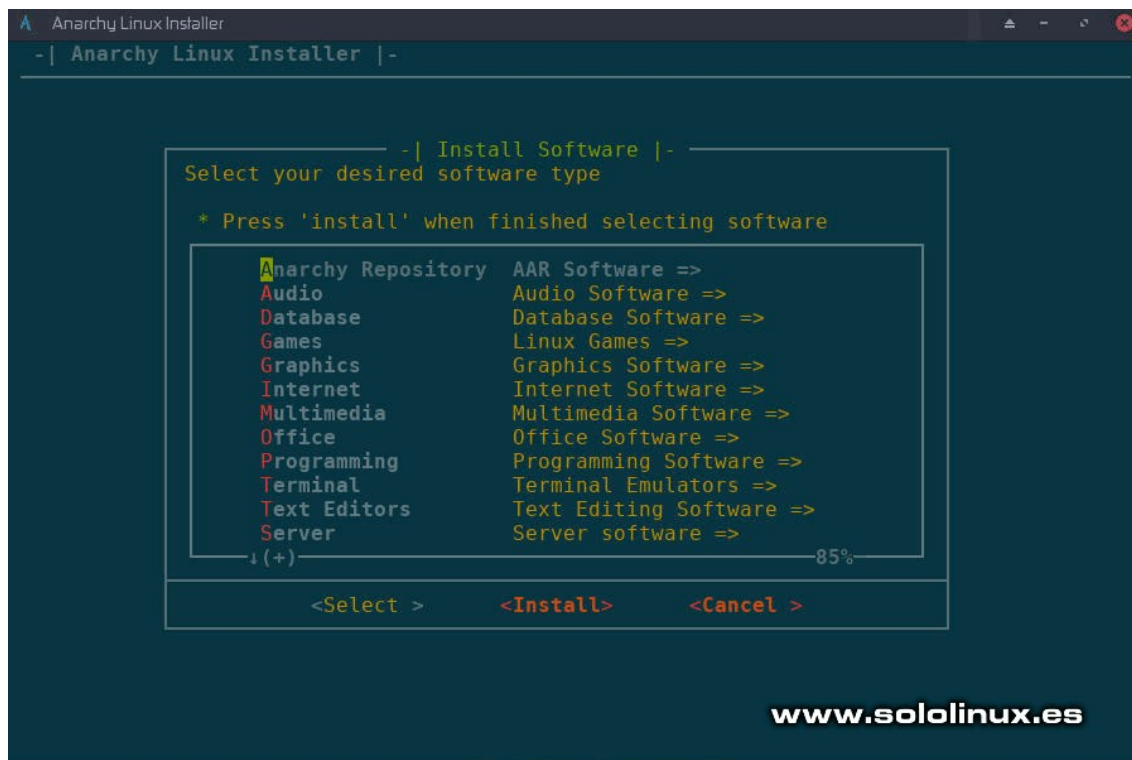
El proceso de aprendizaje de Arch con ArcoLinux se divide en seis fases, y las diferentes distribuciones ArcoLinux son parte de ellas. La versión minimalista se llama **ArcoLinuxD**, y viene con los componentes suficientes para ayudarte a construir tu propia instalación personalizada. También tenemos **ArcoLinuxB** que nos permite modificar el archivo ISO de ArcoLinux y, generar nuestra propia imagen personalizada. El proyecto aloja varias iso descargables, generadas por usuarios de la comunidad. Derivados de Arch Linux.



La versión principal de ArcoLinux viene con un instalador gráfico, muy fácil de usar. Viene con tres entornos de escritorio, y no echarás ninguna aplicación o herramienta en falta. Descarga la versión que más te interese desde su página oficial, pero recuerda que si buscas una distribución lista para usar, debes optar por la versión ArcoLinux normal.

- [Descargar ArcoLinux](#)

Anarchy Installer



Tal vez sería difícil de encajar **Anarchy** en el grupo de **derivados de Arch Linux**, pero considero que se merece una mención importante dado su carácter de intentar acercar Arch, al usuario menos experimentado.

Anarchy es un instalador en línea de comandos de Arch, fácil, intuitivo y sorprendentemente sencillo de usar. Su misión es centrarse en mejorar la experiencia del usuario, ofreciendo unas características con respecto a la personalización difíciles de igualar.

- Elige entre múltiples entornos de escritorio y administradores de ventanas.
- El particionado del disco puede ser automático o manual.
- Amplia selección de kernels Linux a elegir.
- Puedes elegir entre Bash, ZSH, Fish y más,
- Funciones increíbles, como instalar paquetes de aplicaciones personalizados.
- Mucho más...

Realmente... este proyecto se merece un artículo aparte, me lo apunto como tarea pendiente. Puedes descargar **Anarchy Linux Installer** desde su **Gitlab** oficial.

- [Descargar Anarchy Installer](#)

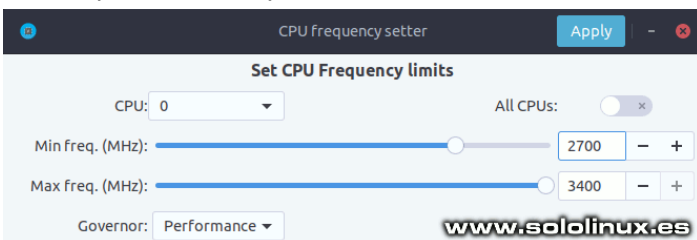
Existen muchos derivados de Arch Linux realmente interesantes, pero creo que estos son los que más darán que hablar en este año 2021.

Ajustar la frecuencia de la Cpu Con CpuPower-GUI



CPUPower-GUI es una sencilla herramienta **GUI**, que nos permite modificar los límites establecidos de frecuencia en la CPU y el gobernador. Su uso es bastante simple, puedes ajustar la frecuencia del núcleo de la CPU con unos sencillos controles deslizantes. También permite cambiar los perfiles del **gobernador de la CPU**, entre Performance y Enlaced.

Nos llama la atención, que se ofrece una opción para habilitar o deshabilitar una CPU específica desde una simple casilla de verificación. Debes recordar, que si la casilla de verificación tiene el color atenuado, quiere decir que la CPU no puede deshabilitarse, está en uso.



Ajustar la frecuencia de la cpu con CpuPower-GUI

La herramienta viene en los repositorios oficiales de muchas **distribuciones linux**, pero normalmente son versiones un tanto obsoletas. vemos como instalar CpuPower-GUI.

Instalar CpuPower-GUI en Ubuntu 20.04

```
echo 'deb
http://download.opensuse.org/repositories/home:erig
as:cpupower-gui/xUbuntu_20.04/ /' | sudo tee
/etc/apt/sources.list.d/home:erigas:cpupower-
gui.list

curl -fsSL
https://download.opensuse.org/repositories/home:erig
as:cpupower-gui/xUbuntu_20.04/Release.key | gpg --
dearmor | sudo tee
/etc/apt/trusted.gpg.d/home_erigas_cpupower-gui.gpg
> /dev/null

sudo apt update

sudo apt install cpupower-gui
```

Instalar CpuPower-GUI en Debian Testing y Unstable

```
# Debian testing
echo 'deb
http://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Debian_Testing/ /' | sudo tee
/etc/apt/sources.list.d/home:erigas:cpupower-
gui.list

curl -fsSL
https://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Debian_Testing/Release.key | gpg
--dearmor | sudo tee
/etc/apt/trusted.gpg.d/home_erigas_cpupower-gui.gpg
> /dev/null

sudo apt update

sudo apt install cpupower-gui

# Debian unstable
echo 'deb
http://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Debian_Unstable/ /' | sudo tee
/etc/apt/sources.list.d/home:erigas:cpupower-
gui.list

curl -fsSL
https://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Debian_Unstable/Release.key | gpg
--dearmor | sudo tee
/etc/apt/trusted.gpg.d/home_erigas_cpupower-gui.gpg
> /dev/null

sudo apt update

sudo apt install cpupower-gui
```

Instalar CpuPower-GUI en Fedora

```
# Fedora 32
dnf config-manager --add-repo
https://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Fedora_32/home:erigas:cpupower-
gui.repo

dnf install cpupower-gui

# Fedora 31
dnf config-manager --add-repo
https://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Fedora_31/home:erigas:cpupower-
gui.repo

dnf install cpupower-gui

# Fedora Rawhide
dnf config-manager --add-repo
https://download.opensuse.org/repositories/home:erig
as:cpupower-gui/Fedora_Rawhide/
home:erigas:cpupower-gui.repo

dnf install cpupower-gui
```

Instalar CpuPower-GUI en OpenSuse Tumbleweed

```
zypper addrepo https://download.opensuse.org/repositories/home:erigas:cpupower-gui/openSUSE_Tumbleweed/
home:erigas:cpupower-gui.repo

zypper refresh

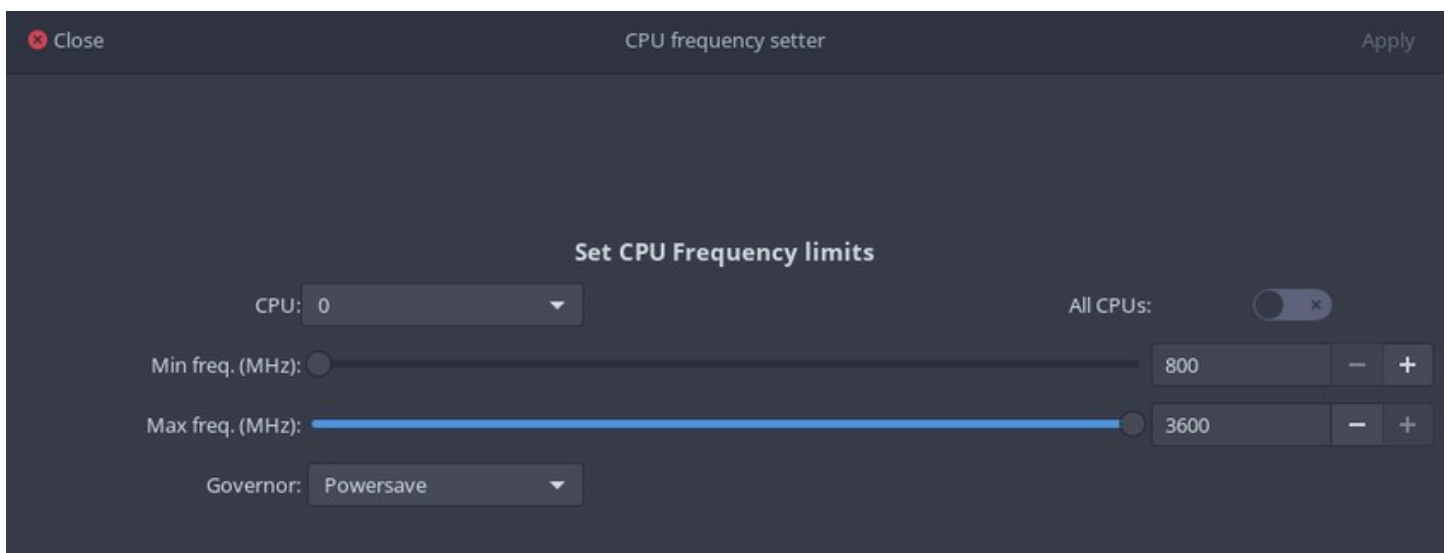
zypper install cpupower-gui
```

Instalar CpuPower-GUI en Arch y derivados

Puedes encontrar el paquete en AUR como **cpupower-gui**.

Con Pamac:

```
pamac build cpupower-gui
```



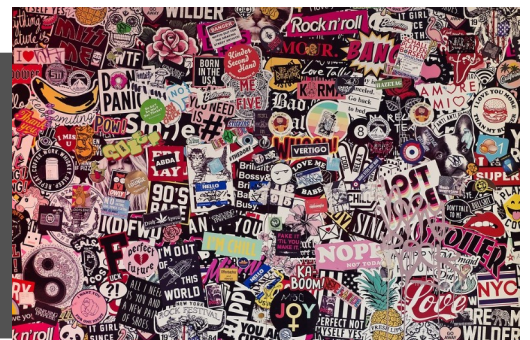
Una vez instalada la herramienta, la podrás encontrar en el menú de aplicaciones de tu entorno de escritorio. Te sorprenderá lo sencilla que es de utilizar.

Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es



Entrevista a Hernán Administrador de su blog personal hernanalbornoiz.wordpress.com

Desde **SOLOLINUX**, continuamos con las entrevistas a Blogs, Webs, desarrolladores de Distribuciones...

En el número de hoy os presentamos el Blog personal de Hernán. Donde encontramos bastante información sobre Software Libre y GNU/Linux.

Dejamos que el mismo se presente a toda nuestra comunidad.

HERNÁN: Me llamo **Hernán Albornoiz** y Vivo en **Victoria Entre Ríos (Argentina)**.

Egresé del Instituto Superior de Informática de Entre Ríos (ISIPER) con el título de **Analista Programador**, de la Facultad de Ciencia y

Tecnología de la Universidad Autónoma de Entre Ríos (UADER), con el título de **Analista Superior en Sistemas** y con el título de **Técnico Universitario en Software Libre** en la Facultad de Ingeniería y Ciencias Hídricas de la Universidad Nacional del Litoral de Santa Fe (UNL).

Actualmente estoy cursando la **carrera de Administrador y Experto en Seguridad en Redes Linux (Linux System Engineer y Ethical Hacking Expert)** en CLA Instituto Linux.

Desde que cursaba la escuela secundaria me interesaba la informática y usé por primera vez una computadora cuando en la escuela me enseñaron a programar en el lenguaje Basic. Luego comencé mis estudios en la ciudad de Paraná Entre Ríos y asistí durante ese tiempo a cuanto curso, seminario conferencia o talleres se hacían, relacionados con el área de la informática.

Me comencé a interesar por el Software Libre específicamente, gracias a las capacitaciones que brindaban los colegas de **Gugler** en la **Facultad de Ciencia y Tecnología de la UADER**. A partir de ese momento me uní a la Comunidad y fundamos junto a unos amigos y colegas, el **Grupo de Usuarios de Software Libre de Victoria Entre Ríos (LUG Victoria)** y me uní también a la **Comunidad de Usuarios de Huayra GNU/Linux** desde sus comienzos. Desde ese momento me hice **entusiasta, activista, militante y defensor del software libre** y nos empezamos a organizar para participar de diferentes eventos con el objetivo de difundir el uso de herramientas libres. Luego con el grupo de Software Libre logramos formar la **Cooperativa de Trabajo de Software Libre COOPEVIC Ltda**, de la cual soy socio y miembro fundador. A partir del año 2017 formo parte de la **organización y coordinación del FLISoL** en mi localidad (**Festival Latinoamericano de Instalación de Software Libre**), siendo este evento declarado de interés por diversos organismos gubernamentales.

Me matriculé en el **Colegio de Profesionales de Ciencias Informáticas de Entre Ríos (COPROCIER)** y ocupé los siguientes cargos: En el período 2012-2014 como 4° Vocal y como Responsable de la Comisión Prensa y Difusión, en el período 2014-2016 como 1° Vocal, en el período 2016-2018 como Titular de Organo de Fiscalización y en el período 2012-2017 como Responsable de la Subsección en Victoria Entre Ríos.

Desde hace unos años me desempeño además como **Capacitador en TIC – Tecnología de la Información y Comunicación** y en el **Taller de Asistente en Armado y Mantenimiento de Computadoras** en el **Centro Comunitario N° 30** dependiente de la Dirección de Educación de Jóvenes y Adultos del Consejo General de Educación de Entre Ríos (CGE).

En este año 2020 pensando que no teníamos oportunidad de armar un **nuevo FLISoL**, recibí una invitación de **Openlab Ecuador** para participar como panelista, contando mi experiencia con el Software Libre y con la organización del FLISoL, en la 2da. sesión sobre **“Experiencias de FLISoL en América Latina”** y fui invitado además para participar como **moderador** en la 3ra sesión de este evento.

Soy además editor en el blog tecnicxslibres.wordpress.com. Este blog se formó por iniciativa de los **profesores, alumnos y egresados de la Tecnicatura Universitaria en Software Libre de la Facultad de Ingeniería y Ciencias Hídricas de la Universidad Nacional del Litoral de Santa Fe (UNL)**.



SOLOLINUX: ¿Cuéntanos un poco sobre tu blog personal <https://hernanalbornoz.wordpress.com/>?

HERNÁN: Este es un blog informativo en donde se publican tanto noticias, novedades o eventos relacionados a la informática y preferentemente al Software Libre y la Cultura Libre, como así también mis experiencias con el Software Libre en los eventos en que he participado, como por ejemplo cuando realizamos algún evento como en el caso de FLISoL (Festival Latinoamericano de Instalación de Software Libre). También publico temas que considero interesantes de otros blogs u otras páginas web, pero en este caso cito la fuente. Mi idea es agregar una sección separada en donde se pueda publicar contenido técnico, es decir, una especie de tutorial sobre GNU/Linux por ejemplo, pero eso lo estoy analizando como hacerlo, si lo publico yo o con ayuda de otros editores.

SOLOLINUX: ¿Hoy en día aproximadamente cuantas personas colaboran en el BLOG?

HERNÁN: Este blog al ser personal lo administro y lo edito yo, pero mi intención es poder formar un blog comunitario en donde participen aquellas personas comprometidas con la comunidad del Software Libre. Hace poco tiempo realicé otro blog para que participen otros colaboradores de mi provincia, pero ha tenido muy poco movimiento y participación, así que mi intención es poder algún día realizar un blog un poco más amplio con referentes del Software Libre de mi país (Argentina) o también podría ser de América Latina.

SOLOLINUX: ¿A que perfil de usuarios estas destinados los artículos del BLOG?

HERNÁN: Los artículos del blog al ser solo informativo, está destinado a todo tipo de usuarios, con conocimientos o sin conocimiento sobre informática o Software Libre y como dije anteriormente, en caso de publicar en algún momento material más técnico, quiero que las publicaciones no sean tan complejas y sean entendibles y simples de interpretar por la mayoría de las personas que recién se inician en el mundo del Software Libre.

SOLOLINUX: ¿Donde y como podemos ponernos en contacto con vosotros?

HERNÁN: Para cualquier consulta pueden contactarse conmigo a la siguiente sección de contacto de mi blog <https://hernanalbornoz.wordpress.com/contacto/> o al siguiente correo electrónico hernanalbornoz70@gmail.com o en las redes sociales como por ejemplo facebook <https://www.facebook.com/hernan.albornoz.7/>, twitter [@CarlosHernanAlb](https://twitter.com/CarlosHernanAlb) y Telegram [@hertux](https://t.me/hertux)

SOLOLINUX: Y para terminar ¿Podrías darnos tu opinión sobre GNU/LINUX y software libre?

HERNÁN: Para comenzar voy a comentar como llegué a usar GNU/Linux y por último voy a dar mi opinión sobre este Sistema Operativo.

Conocí GNU/Linux muchos años después de que me recibí. Fue en una charla en la misma facultad en donde obtuve mi título. Hasta el momento solo conocía el Sistema Operativo Windows y no comprendía bien que diferencias existía con GNU/Linux.

Primero me llamó la atención la comunidad que se formaba detrás de el y el entusiasmo que ponían los estudiantes, como una especie de activismo social, cosa que no se veía con el otro sistema. Luego comprendí que era por su filosofía y por las cuatro libertades del Software Libre.

Comencé en el año 2005 como un usuario común más, la primera distribución que usé fue Mandriva, cuyo nombre derivaba de la fusión entre la empresa francesa Mandrake Soft y la brasileña Conectiva.



En el año 2009 buscando en Internet encontré una empresa denominada Canonical con sede en Londres y en donde ofrecían gratuitamente el Sistema Operativo Ubuntu GNU/Linux. Se podía pedir una distribución, cinco o hasta diez distribuciones. Como pensé que no iban a mandarme nada, probé pidiendo una distribución y por sorpresa a los pocos días llega el correo a mi casa con el sistema operativo Ubuntu GNU/Linux.

Fue recién a finales del 2012 que nos involucramos más activamente en la difusión del Software Libre cuando sale el Sistema Operativo argentino Huayra GNU/Linux para las netbooks del Programa lanzado por el Gobierno Nacional denominado Conectar Igualdad y que era destinado a la educación. Ahí comprendí mucho más la importancia de promover y difundir el Sistema Operativo GNU/Linux, cualquiera sea la distribución, en este caso era Huayra, ya que empecé a comprender el alcance social que esto originaba como así también la soberanía tecnológica.

También noté que con la salida de este sistema nacional, se involucraron muchos docentes y público en general con pocos conocimientos en informática y con intención de formar nuevas comunidades de Software Libre con un gran propósito como lo es la educación. Antes de la salida de este sistema, yo veía que los eventos de Software Libre como por ejemplo el FLISoL en donde mostraban diversas distribuciones de Software Libre, solo alcanzaba a una pequeña porción de la población con pocos conocimientos de informática ya que la mayoría de los asistentes eran informáticos y este tipo de evento se transformaba en una especie de reunión de gente geek o nerds que mostraba cuanto conocimiento tenía del tema.

Desde este momento me uní a la Comunidad de **Huayra GNU/Linux** y participé en cuanto evento podía con los compañeros de la comunidad para mostrar la importancia de usar un sistema operativo libre y propio para el crecimiento y desarrollo de la sociedad.

Ahora voy a dar mi opinión sobre el Software Libre. Como programador que soy, creo que el conocimiento no se debe restringir, es por ello que todos deberíamos usar Software Libre. Creo además que el uso de Software Libre logra de que la ciencia y la tecnología avancen más rápidamente y la sociedad en su conjunto se beneficie. Es por ello de que el código fuente de los programas tienen que ser libres para que otros programadores puedan reutilizarlo, ya sea para ampliarlo, mejorarlo o adaptarlo a sus necesidades. ¿Se imaginan ustedes si nuestros maestros en las escuelas primarias y secundarias nos ocultaran información?, la educación sería muy distinta.

Creo además que al ser libre el software, muchos programadores locales o regionales se benefician porque al tener acceso al código fuente no deben “reinventar la rueda” como acostumbramos a decir los programadores, sino que tienen la posibilidad de desarrollar software en menor tiempo ya que hay cosas que ya se inventaron. Muchas personas tienen el mito de que al ser el código libre, necesariamente tiene que ser gratuito y no alcanzan a comprender que al liberar el código fuente no significa que no podamos cobrar por nuestro trabajo. Al contrario de lo que piensa estas personas, tal vez por desconocimiento, creo que es altamente beneficioso porque tanto los gobiernos como las empresas privadas pueden dejar de pagar costosas licencias de software privativo, que además, nunca van a ser dueñas del software ya que el código está enlatado y corren además el riesgo de intrusiones y hacer peligrar la seguridad informática, al no saber si el software adquirido no tiene alguna puerta trasera en donde puedan espiar nuestro sistema y en el peor de los casos, producir un daño en los datos guardados.

Y por último, creo además, de que el Software Libre debe ser enseñado desde la escuela primaria, desde los más chicos y que no solo aprendan sobre aplicaciones de informática sino que también deben aprender a programar y ver con el tiempo las ventajas de usar Software Libre y que no los conviertan en esclavos del Software Privativo. Es sumamente importante que desde chicos aprendan a compartir los programas y las aplicaciones con otros compañeros y sin estar incumpliendo con ninguna ley de propiedad intelectual. Además, algo tan importante como la inclusión digital, ya que al ser libre el software, la brecha digital se achica y todos puedan tener las mismas posibilidades de acceder a las nuevas tecnologías.

Hay una frase que vi en un blog que me gustó mucho y que resume un poco todo lo dicho anteriormente y que dice “Si tengo una manzana y la comparto, me quedo sin manzana. Pero si tengo una idea y la comparto, no me quedo sin ideas. Esa es la Filosofía del Software Libre”.

Y para aquellos que como todo principiante comenzó alguna vez a usar Software Libre y ya no se pudieron separar de él, tengo esta frase de una docente que tuve y que dice **“El Software Libre es un camino sin retorno”**

Para finalizar, quiero agradecer a **www.sololinux.es** por permitirme participar de esta edición N.º 22 de la revista con esta entrevista.

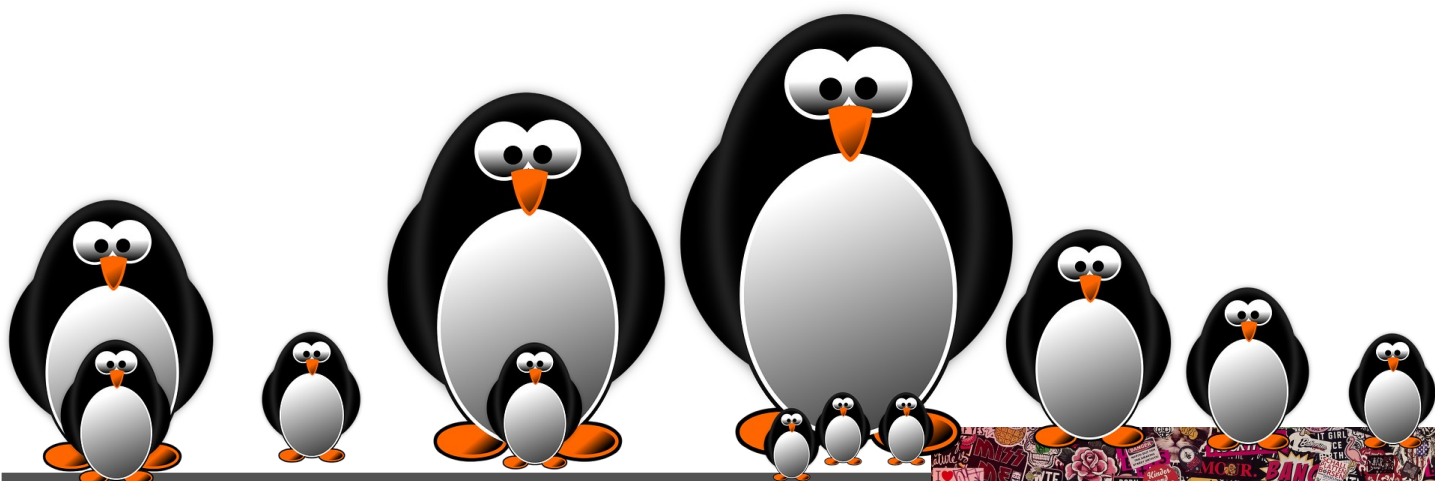
Quiero además contarles, de que descubrí esta revista a través de los buscadores de Internet y me encantó por el material y el diseño de la misma. Desde ese momento empecé a descargarlas y a leerlas a todas. Así que decidí también compartir desde ese momento esta revista todos los meses desde mi blog personal **hernanalbornoza.wordpress.com** para difundirla a toda la comunidad. Muchos de mis conocidos se han enterado de esta revista por estas publicaciones.

Siempre estoy al tanto de las novedades que surgen en relación al Software Libre y la Cultura Libre y como siempre, tratando de aportar mi granito de arena y de brindar mi apoyo y ayuda en la difusión del Software Libre en donde sea posible y necesario, para el beneficio de la sociedad. Es por ello que me gusta que me identifiquen como un militante más del Software Libre. Un saludo y sigan así con la revista que está muy buena.

¿Te interesa colaborar con SoloLinux?

Puedes enviarnos tus ideas,
propuestas, artículos, opiniones a
nuestra **dirección de correo:**
adrian@sololinux.es

También puedes contactar con nosotros mediante **TELEGRAM** en el canal **@sololinux**.

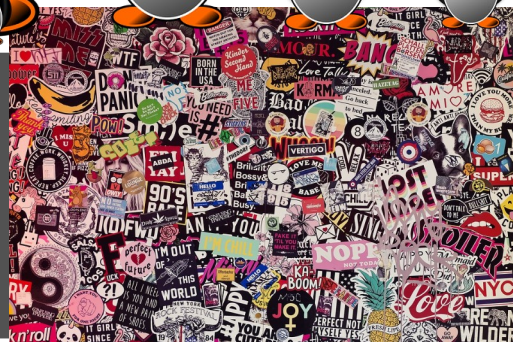


Publicidad:

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital gratuita de software libre y GNU/Linux en ESPAÑOL.

CON **SOLOLINUX** MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a:
adrian@sololinux.es





VANT

www.sololinux.es

