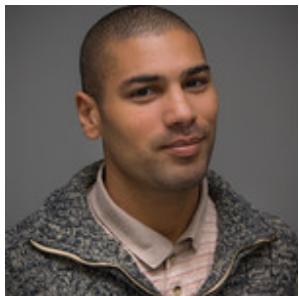


# Instalación, desinstalación y actualización de aplicaciones libres y propietarias.

## Caso práctico

**Juan** le ha comentado a **Vindio** que deben probar diferentes aplicaciones antes de ejecutarlas en la máquina real, para ello necesitará realizar su instalación en el ordenador "caja de herramientas" que ha creado con los diferentes plataformas de sistema operativo para su funcionamiento. Necesita instalar y desinstalar las aplicaciones y actualizaciones descargadas desde Internet o aportadas por proveedores de software.



[Alain Bachellier \(CC BY-NC-SA\)](#)

**Vindio** le contesta que se va a poner de inmediato a realizar dicha tarea y aprovechando que tienen a alumnas en prácticas del Ciclo de ASIR, les va a explicar como hacerlo y así les ayudaran en esta tarea.



[Jonny Goldstein \(CC BY\)](#)

Después de disponer de los sistemas operativos instalados aprenderemos a través de las siguientes unidades a operar y manejar dichos entornos de trabajo.

Esta unidad servirá para captar los suficientes conceptos como la aplicación de técnicas de actualización, conocer el proceso de inicio de los sistemas, gestión del registro de los sistemas que servirán pasar a realizar configuración base de los sistemas operativos que se verá en la siguiente Unidad.

También se estudia las diferentes formas de ampliación de los sistemas operativos mediante aplicaciones estudiando aspectos de la administración en instalación/desinstalación y actualización de las mismas.

Como ya disponemos de los sistemas instalados en nuestro ordenador de trabajo, se recomienda estudiar los contenidos con el ordenador como elemento de consulta y realización de los ejemplos realizados en la explicación de cada apartado, con el fin de facilitar la compresión y memorización de los conceptos teórico/prácticos.



[Ministerio de Educación y Formación Profesional](#) (Dominio público)

**Materiales formativos de FP Online propiedad del Ministerio de Educación y Formación Profesional.**

[Aviso Legal](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

# 1.- Instalación y desinstalación de aplicaciones. Requisitos, versiones y licencias.

## Caso práctico



[Alain Bachellier \(CC BY-NC-SA\)](#)

**Vindio** les explica a **Noiba, Naroba y Jana**, que van a utilizar máquinas virtuales para probar aplicaciones en diferentes sistemas operativos. En estas máquinas virtuales van a instalar/desinstalar aplicaciones y comprobar que funcionan sin ningún problema. Además van comprobar si necesitan actualizaciones, realizar dichas actualizaciones si son necesarias, tanto si son descargadas desde Internet o aportadas por proveedores de software.

**Noiba** le pregunta a **Vindio**, ¿Qué es una máquina virtual? Este le responde que es un software capaz de instalar un sistema operativo en su interior haciéndole creer que es un ordenador de verdad.

**Naroba** les comenta que ella ha utilizado máquinas virtuales y que son muy útiles pues permiten instalar cualquier aplicación o programa como si estuvieran utilizándolo en una maquina real.



[Alain Bachellier \(CC BY-NC-SA\)](#)

**El usuario Administrador del Sistema Informático (root en el caso de Linux) es el responsable de la instalación y desinstalación** de aplicaciones en el entorno operativo. Generalmente las aplicaciones se aportan en el formato denominado paquete que puede contener documentación, librerías, iconos o cualquier otra cosa que forma los programas necesarios para poner en funcionamiento la aplicación. Dichos paquetes se suelen almacenar para su distribución en soportes físicos como memorias Flash, CDs, DVDs o bien, se pueden descargar de Internet.

En muchas ocasiones por problemas de espacio y de facilitar la portabilidad el paquete (conjunto de programas que realizan una tarea y permiten el correcto funcionamiento de una aplicación informática) viene empaquetado y comprimido en un sólo fichero, y para su

instalación deberemos descomprimir y desempaquetar con alguna herramienta o aplicación (Zip, rar, etc.).

Las aplicaciones vienen adaptadas a las plataformas de hardware donde pueden ser instaladas mediante la identificación de la versión, por ejemplo, podemos tener aplicaciones con diferentes versiones adaptadas a plataformas con modelo de la CPU de x86(32 bits) o x64(64 bits). Otro dato a tener encuentra para seleccionar la aplicación son las limitaciones de uso por el número de licencias de uso, por ejemplo, Windows Server 2019.

Debemos considerar que los sistemas operativos actuales disponen de algún módulo encargado de facilitar al usuario administrador la instalación y desinstalación de aplicaciones encargándose de relacionar las dependencias de unas aplicaciones con otras y la compartición de librerías (librerías dinámicas). Los paquetes informáticos para su perfecto funcionamiento tiene que disponer de un fichero ejecutable que permita cargar y arrancar la aplicación, en ocasiones el usuario administrador adquiere los ficheros fuente teniendo que realizar la tarea de creación del ejecutable mediante el proceso de enlace o enlazado de librerías para pasar a la compilación del programa (creación del código binario que es el que entiende la máquina), para este proceso la propia aplicación dispone de las herramientas necesarias para realizar dicho proceso. La ventaja de los paquetes de tipo código fuente de los programas y ficheros es que permiten, hasta cierto punto, la modificación y configuración particular de la aplicación antes de su instalación para su personalización al entorno de trabajo.

Las aplicaciones que son directamente instalables vienen en paquetes binarios sin posibilidad de cambios directamente adaptadas a la máquina o hardware.

En la información de ayuda que nos aporta el fabricante o diseñador de la aplicación, "Manual operativo de instalación", podemos encontrar los datos informativos de la aplicación que pueden ser importantes para gestionar correctamente la posterior instalación como pueden ser: descripción de la aplicación (disponibilidad de versiones), dirección web del fabricante, nombre y datos del proveedor, licencias y derechos de uso, compatibilidades con los sistemas operativos y hardware, estructura de archivos, dependencias, logs, etc.

En el sistema operativo Linux las aplicaciones se identifican mediante una numeración que indica para qué distribución está dirigida, tipo de paquete, tipo de núcleo, etc. De la siguiente forma:

samba-0.7.3-1.0.src.i386.rpm

- Nombre de la aplicación
- Versión del paquete
- Número de entrega de la versión del paquete
- S.O. para que es compatible, `src` indica que es código fuente
- Instalable para micros de arquitectura Intel x86
- Tipo de paquete, rpm manejable por distribuciones red hat

Elaboración propia (Dominio público)

## **1.1.- Fases de implantación y desinstalación de aplicaciones.**

---

Cuando se instala una aplicación es recomendable realizar una serie de operaciones o fases de implantación:

**Estudio y análisis del software a instalar:** verificación de la compatibilidad con el hardware y otras aplicaciones (sobre todo con el sistema operativo), número de licencias, tipo de licencia... Analizar la necesidad de modificación del entorno operativo y sustitución de unas aplicaciones por otras.

Antes de comenzar la instalación debemos de realizar **copias de seguridad del sistema** de los datos y guardar las configuraciones para evitar posibles pérdidas de información.

Analizar la necesidad de creación de **particiones de disco**.

Estudiar el **mecanismo de instalación** consultando el manual de la aplicación: por copia al sistema, por gestor de paquetes, por instalador propio, por desempaquetado/descomprimir, instalación/compilación, etc.

Después de la instalación será necesario **realizar la configuración** para adaptar la aplicación al entorno operativo.

En muchas ocasiones será necesario **registrar la aplicación** para su uso legal y completo, bien a través de teléfono o aceptando la actualización de la misma y condiciones on line (web).

Esquema a seguir para la instalación de aplicaciones:

Elaboración propia (Dominio público)

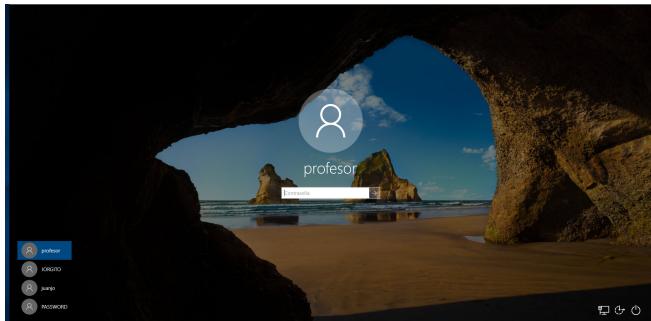
Cuando realizamos una **desinstalación de la aplicación hay que considerar los siguientes aspectos:**

Que no afecte a otras aplicaciones ya que se pueden producir problemas de dependencia y de integridad; de la misma manera que en la instalación se recomienda realizar copias de seguridad de la mayor parte del sistema y seguir el proceso mediante una herramienta o aplicación específica para este proceso aportada por el propio sistema operativo o por el propio programa/aplicación a desinstalar (el llamado programa **uninstall**).

Si el programa dispone de su propia herramienta de desinstalación, ejecutar la misma. Estudiar el mecanismo de desinstalación consultando el manual de la aplicación. Si el sistema nos pregunta de si deseamos eliminar ficheros que puede que dependan de otras aplicaciones es aconsejable contestar siempre que no, para evitar posibles errores a la hora de ejecutar otras aplicaciones. Si los datos generados por el uso de la aplicación, bases de datos, cuentas de correo, perfiles de usuario, deberemos realizar una copia de seguridad de dicha información o exportar dicha información a otra aplicación en uso, para dar la posibilidad de de poder usar el trabajo ya realizado.

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

# 1.2.- Instalación/desinstalación de aplicaciones en el sistema operativo Windows.



Microsoft - Captura de pantalla (Elaboración propia)

En el proceso de instalación el sistema solicita el nombre del usuario que actuará de administrador dentro del sistema. Si durante ese proceso creamos más usuarios o introducimos una contraseña para el usuario, el sistema de arranque solicitará el llamado **Login de entrada** donde deberemos identificarnos con nombre de usuario y clave para entrar en el sistema con un perfil determinado (con unos derechos y privilegios sobre los recursos del sistema).

El sistema administrador de paquetes (Package Management System) de Windows se encuentra en la herramienta de Agregar o Quitar software o en Programas en el Panel de Control. **El sistema administrador de paquetes gestiona la información de los paquetes instalados en el denominado repositorios de paquetes** (es una base de datos).

Los proveedores de aplicaciones para Windows aportan su propio sistema de instalación y desinstalación. En algunos procesos el paquete se aporta en formato ISO o comprimido con alguna aplicación del mercado informático.

**Las distribuciones se suelen distribuir en formatos comprimidos o ISO.** Comprimir una aplicación consiste en generar un único archivo que contendrá todos los ficheros y carpetas que conforman la aplicación para facilitar su transporte y distribución. Windows dispone de su propia **herramienta de compresión** mediante:

La **Compresión NTFS**: para comprimir archivos, carpetas o unidades de almacenamiento completas. Para su realización se pueden seguir los siguientes pasos:

Seleccionar la carpeta, mostrar menú con botón derecho del ratón o archivo desde el *Explorador de Windows*, seleccionar opción *Propiedades* y de la pestaña *General* pulsar en el botón de *Opciones avanzadas*.

En la ventana de *Atributos avanzados* marcar *Comprimir contenido para ahorrar espacio en disco*. En este punto debemos saber que no se puede tener una carpeta comprimida y cifrada a la vez, y que una carpeta comprimida puede tener archivos y subcarpetas no comprimidos, pero una carpeta no comprimida puede contener archivos comprimidos y evidentemente, no comprimidos.

Se mostrará una ventana de diálogo llamada *Confirmar cambios de atributos* donde podemos elegir entre *Aplicar cambios sólo a esta carpeta* o *Aplicar cambios a esta carpeta y todas las subcarpetas y archivos*.

*Carpetas comprimidas* permite crear carpetas comprimidas de manera que todos los archivos que se graban en dicha carpeta se comprimen. Para crear una carpeta de este tipo debemos pulsar desde el explorador en *Archivo-Nuevo-Carpeta Comprimida (en zip)*. Seguidamente todos los archivos que se vayan alojar estarán comprimidos, para descomprimir simplemente pasar los archivos a una carpeta normal o que no esté comprimida. Existen aplicaciones en el mercado que permiten comprimir/descomprimir aplicaciones como puede ser Winrar, **Winzip**, etc.

Existen aplicaciones de tamaño muy grande que se suelen descargar desde Internet en formato de un único ficheros o imagen ISO (es un archivo que contiene toda la información de un disco óptico y que comprende tanto la información real que necesitamos, como los datos sobre la estructura que esta información sigue en el dispositivo, su extensión es ISO). Para gestionar una imagen ISO y generar el disco CD o DVD original con la estructura de ficheros y directorios originales debemos de usar programas específicos de freeware como: **UltraISO**, **ISO Maker**, **K3b**. Entre los programas comerciales destacan: **CDBurnerXP**, **Easy CD Creator**, **ImgBurn**, **InfraRecorder**, **Nero Burning ROM**, **Roxio Creator**.

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 1.2.1.- Proceso de Instalación y desinstalación de aplicaciones Windows.

A la hora de instalar una aplicación es conveniente cerrar todas las aplicaciones y se deberá iniciar sesión como un usuario administrador.

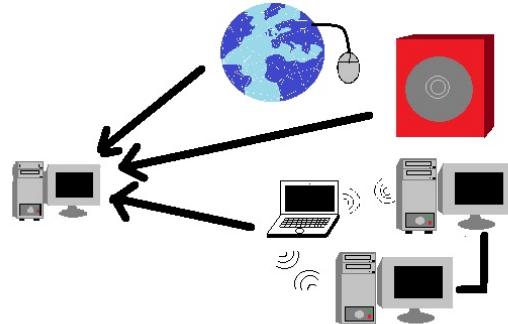
Podemos encontrarnos con tres casos diferentes o maneras de instalar una aplicación:

**Desde CD o DVD aportado por el fabricante:** si el sistema tiene configurado la Reproducción automática de soportes de CD/DVD se ejecuta automáticamente un asistente de instalación del programa aportado por la aplicación. Para activar en Windows la reproducción automática de un dispositivo seguir los siguientes pasos:

Botón *Inicio-Panel de control*. En el cuadro de búsqueda, escriba reproducción automática y, a continuación, en la lista de resultados, haga clic en *Iniciar o detener el uso de la reproducción automática para todos los medios y dispositivos (Windows 10)*.

En la lista que se encuentra junto al dispositivo o tipo de medio, haga clic en la nueva acción que desea utilizar.

Si un programa no inicia la instalación automáticamente podemos examinar el disco y abrir el archivo de instalación del programa, que generalmente se llama *<i>Setup.exe</i>* o *<i>Install.exe</i>*. Si el programa está creado para que funcione en una versión anterior de Windows podemos configurar el sistema para intentar solucionar el problema de que Windows 10 pueda ejecutar programas de versiones anteriores pulsando en: *Inicio-Panel de control*. En el cuadro de búsqueda, escriba *solucionador de problemas* y, después, haga clic en *Solución de problemas*. Desde *Programas-Ejecutar programas creados para versiones anteriores de Windows*. Seguir el asistente del solucionador de problemas.



Elaboración propia (Dominio público)

**Desde un distribuidor de Internet:** necesitamos conexión a Internet.

Entrar en la página Web donde se encuentra la aplicación a descargar, pulsar en enlace del programa.

De la ventana de diálogo que aparece podemos descargar el archivo para que desde una carpeta del disco duro pulsar en el programa para iniciar el asistente de instalación o pulsar en el botón de *Abrir* o *Ejecutar* con lo que aparecerá el asistente de instalación aportado por el proveedor de la aplicación.

**Por la red local donde se encuentra conectado el equipo:** significa que el ordenador se encuentra en un Grupo de trabajo o en un Dominio, la aplicación se encuentra alojada y compartida en otro equipo. Podemos instalarla desde el *Panel de control* siguiendo los pasos siguientes:

*Inicio-Panel de control-Programas-Programas y funciones-Instalar un programa desde la red*.

Selecciona un programa de la lista y, a continuación, haga clic en *Instalar*.

Muchas aplicaciones de Windows disponen de su propio programa desinstalador, para acceder a él deberemos de ir a *Inicio-Programas-* buscar la carpeta de la aplicación, entramos, seleccionar la opción *Desinstalar* o *Uninstall*.

Debemos avisar que durante el proceso de desinstalación de una aplicación Windows puede avisarnos mediante un mensaje de que algunos archivos (normalmente librerías) son compartidos con otra aplicación, en estos casos el sistema nos muestra una ventana de diálogo donde nos permite seleccionar si queremos eliminar el fichero o no, es aconsejable no eliminarlo a no ser que el administrador tenga el conocimiento de que no afectará al funcionamiento de otros programas.

Para desinstalar aplicaciones en Windows, podemos realizarlos desde dos entornos:

**Programa propio** que aporta la aplicación denominado normalmente **uninstall o desinstalar**:

Inicio - buscamos el programa, pulsamos botón derecho encima y seleccionamos la opción *Desinstalar*.

Otra opción es buscar el programa uninstall específico en el menú inicio de la aplicación a desinstalar.

Desde la **herramienta de desinstalación que aporta Windows**:

*Inicio-Panel de control-Programas*.

Aparece la lista de programas instalados. Seleccionar un programa y, a continuación, hacer clic en *Desinstalar*. Algunos programas incluyen la opción de cambiar o reparar el programa además de desinstalarlo, pero muchos solo ofrecen la opción de desinstalación. Para cambiar un programa, hacer clic en *Cambiar* o en *Reparar*.

## Autoevaluación

**¿En qué ventana de proceso de Windows se encuentra la herramienta que gestiona la instalación y desinstalación de aplicaciones?**

### Sugerencia

- Ejecutar programa creados para versiones anteriores de Windows.
- Entorno de instalación de aplicaciones.
- Panel de control.
- Accesorios.

**Incorrecto!** Debes volver a leer la parte del proceso de instalación y desinstalación de aplicaciones en Windows.

**Incorrecto!** Debes volver a leer la parte del proceso de instalación y desinstalación de aplicaciones en Windows.

**Correcto.** Muy bien, veo que vas progresando

**Incorrecto!** Debes volver a leer la parte del proceso de instalación y desinstalación de aplicaciones en Windows.

# Solución

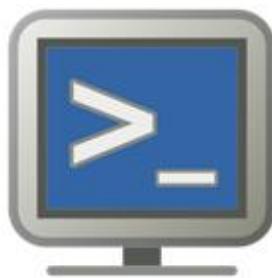
1. Incorrecto
2. Incorrecto
3. Opción correcta
4. Incorrecto

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 1.3.- Instalación/desinstalación de aplicaciones sistema operativo Linux.

Para instalar aplicaciones en Linux se puede **hacer en modo consola para recibir las órdenes mediante comandos de texto** o a través de un entorno gráfico, que es un entorno mucho más agradable y más parecido a otros sistemas como Windows. **En el modo consola, los comandos son ejecutados por el Shell** (intérprete de comandos). Los intérpretes de Linux son el *<i>sh</i>*, el *<i>csh</i>*, el *<i>ksh</i>* y el *<i>bash</i>* (el más usado, suele estar por defecto). Cada uno de ellos se diferencia del anterior en que mejora y complementa las órdenes existentes y añade nuevas posibilidades. Para entrar en un terminal de consola o línea de comandos, pulsamos en *mostrar aplicaciones*, escribimos terminal y hacemos clic en **Terminal**. Otra forma de hacerlo, es desde el escritorio, pulsamos la combinación de teclas **Ctrl+Alt+T** y se abre la consola.

Durante el proceso de instalación de Linux hay un paso en el que se pide el nombre de usuario para comenzar la sesión en el sistema (login de conexión) y una contraseña. Este será el usuario del sistema con el que se podrá conectar, abrir sesiones gráficas y entrar en modo consola. Además de este usuario, el sistema crea un usuario administrador o superusuario, llamado **root**, que tiene privilegios para realizar cualquier tarea en el sistema, por seguridad root no puede iniciar sesión en el sistema (se puede configurar que pueda en el login de conexión local y que pueda entrar de forma en conexión remota desde otro terminal que no sea el propio servidor).



[Openclipart](#) (Dominio público)

Cada vez que se intenta realizar alguna tarea de administrador como dar de alta nuevos usuarios o configurar todos los servicios que ofrece el sistema, el sistema en modo gráfico nos solicitará la contraseña de root que coincide con la contraseña del primer usuario genérico creado en el proceso de instalación, en modo consola el sistema nos avisará de que no tenemos privilegios cuando ejecutamos algún demonio (script, un proceso que normalmente está cargado en memoria esperando una señal para ser ejecutado) o comando de administración.

El sistema habilita un comando "sudo" por el que este usuario genérico puede realizar tareas de root sin serlo. A un usuario con *<i>sudo</i>*, le otorga privilegios de administrador al incluirlo en el grupo de usuarios 'admin' (el archivo /etc/sudoers, registra cada usuario en una línea).

Podemos comprobar si estamos como usuario root u otro distinto mediante el símbolo identificativo del prompt (path de entrada de línea de comando u órdenes). Para un usuario cualquiera el prompt es el símbolo *<i>\$</i>* mientras que para el root es *<i>#</i>*. Desde el usuario creado podemos para **realizar tareas de administrador con el comando**, siempre que escribamos la clave del usuario creado en el proceso de la instalación:

```
$ sudo comando_demonio  
password (introducir la contraseña del usuario)
```

Para que el "usuario root" se convierta de forma permanente, para que no solicite contraseña en cada comando (no es muy recomendable por seguridad del sistema), se puede ejecutar la siguiente orden:

```
$sudo su  
password (introducir contraseña de usuario)  
#
```

Para **salir como usuario root** ejecutamos la orden:

```
# exit
```

Para **cambiar la contraseña de root** para que no coincida con la del usuario genérico (por seguridad en el sistema) escribimos desde la línea de entrada:

```
# passwd
```

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 1.3.1.- Tipos de paquete de instalación y desinstalación en Linux.

Linux durante el proceso de instalación nos da la oportunidad de seleccionar paquetes de aplicaciones adicionales que sean necesarios para su correcto funcionamiento. Además, el sistema instala algún tipo de sistema de administración de paquetes que con sólo conocer el nombre del paquete se pondrá a instalarlo desde los repositorios o bases de datos de paquetes remotas (por conexión a Internet). Dichas herramientas de actualización on-line posibilitan la instalación y desinstalación de software, analizando el espacio en disco que necesita una aplicación, comprueba las dependencias con ficheros de otras aplicaciones, etc.

**Las herramientas o aplicaciones de instalación utilizadas por las distribuciones más populares son:**

Distribución	Herramienta Online	Administradores de Gestión de Paquetes
Debian-Ubuntu	APT	dpkg, Synaptic, Apt, Adept, Aptitude
Red Hat-Fedora	up2date, yum, APT	Rpm, yum
SUSE	Yast2	Rmp, Yast
Mandrake	urpmi	Rpmdrake

Cada distribución utiliza un sistema **Gestor de paquetes** o Administrador de paquetes, por ejemplo Red-Hat utiliza los paquetes .rpm, Slackware utiliza los paquetes .tar, Debian utiliza los paquetes .deb, etc. De manera que el nombre del paquete viene con una extensión (nombre de tres caracteres o más separado por un punto) que identifica el gestor de paquetes utilizado. La distribución Ubuntu utiliza el sistema de paquetes de Debian (.deb).

Para poder realizar copias de seguridad se utilizan herramientas de empaquetado (juntar todos los ficheros y carpetas de una aplicación en uno sólo) y de compresión (reducir el espacio ocupado por un fichero en bytes).

**Los paquetes de aplicaciones o programas puede estar en dos formatos diferentes:**

**Binarios:** contiene ejecutables, archivos de configuración, páginas man/info, información de copyright y documentación en general. Tienen la extensión .deb y se desempaquetan con la utilidad dpkg.

**Fuentes:** conjunto de archivos que definen el paquete. Se empaquetan /desempaquetan con dpkg-source.

Los **paquetes binarios .deb** contienen los siguientes archivos:

**debian-<span lang="en">binary</span>**: contiene la versión del archivo .deb  
**sección de control** del paquete ( control.tar.gz ) que contiene:

Archivo **control** (metadatos): dependencias del paquete, prioridad, mantenedor, arquitectura, conflictos, versión, md5sum,...

Scripts que se ejecutan antes o después de instalar el paquete para, por ejemplo, detener los servicios de los paquetes que se actualizan hasta que su instalación o actualización se completa o ejecutan órdenes necesarias para empezar o reanudar un servicio una vez que el paquete ha sido instalado o actualizado, etc.

**<b>data.tar.gz</b>**: contiene todos los archivos que se instalarán, con sus rutas de destino.

Los **paquetes fuente** contienen los siguientes archivos:

**<b>dsc</b>**: archivo de descripción del paquete. En general es información sobre el paquete.

**<b>orig.tar.gz</b>** : archivo fuente original.

**<b>diff.gz</b>** : archivo con los cambios de Debian sobre el código fuente original.

## **Los nombres de los paquetes binarios en Ubuntu siguen la siguiente convención de escritura:**

**nombre\_NúmeroDeVersión-NúmeroDeRevisiónUbuntu.deb**

Es el nombre del paquete

Es el número de versión establecido por el desarrollador. No sigue un estándar.

Es el número de revisión que normalmente implica cambios en los archivos de configuración usado en el paquete.

Es la extensión utilizada para identificar los paquetes de Ubuntu.

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 1.3.2.- Las herramientas de gestión de paquetes Linux.

---

Los gestores de paquetes se pueden utilizar desde la línea de órdenes como comandos o instrucciones para ellos deberemos entrar en una consola de comandos, pulsamos en *mostrar aplicaciones*, escribimos terminal y hacemos clic en **Terminal**. Otra forma de hacerlo, es desde el escritorio, pulsamos la combinación de teclas **Ctrl+Alt+T** y se abre la consola. Las herramientas más usuales dependiendo de la distribución de Linux son:

**La orden <i>dpkg</i>** : Herramienta de bajo nivel que se utiliza para instalar, eliminar y dar información sobre los paquetes .deb y constituye la base fundamental del sistema de gestión de paquetes de Debian.

**La orden apt** : Herramienta que permite instalar paquetes a través de la red y resolver los conflictos en las dependencias de paquetes que se puedan presentar.

**El comando <i>tar</i>** : Capaz de comprimir y empaquetar paquetes de ficheros y directorios que forman una aplicación. Generalmente las aplicaciones en código fuente vienen en este formato. Los paquetes fuente suelen estar comprimidos en ficheros con las siguientes extensiones: .tgz (o **tar.gz**) y .bz2.

**RPM**: Permite la instalación, modificación, actualización y borrado de los paquetes del sistema. En el proceso de instalación realiza comprobaciones de dependencias y versiones registrando los paquetes instalados en la base de datos de RPM. Tiene el problema de que no es capaz por sí sólo de obtener los ficheros de dependencias. El administrador será el encargado de realizar dicha tarea, de localizar las librerías e instalarlas previamente al paquete que se quiere instalar.

**La orden <abbr title="GNU Zip">Gzip</abbr>** : Comando que permite comprimir, nombre del archivo se identifica con la extensión .gz y debe de ser descomprimido antes de ser usado.

**La orden Bzip2** : Permite comprimir archivos. Comprime más que gzip (el algoritmo utilizado es más completo). La extensión que crea es .bz2

**La orden Zip**: Comprime y descomprime archivos permitiendo la transparencia y portabilidad con otros entornos de sistemas como Windows. Con el comando zip se comprime y con unzip se descomprime.

**La gestión de paquetes en Ubuntu se puede realizar de forma sencilla mediante la aplicación que se ejecuta en una ventana gráfica Synaptic. La aplicación Synaptic es una forma gráfica del sistema gestor de paquetes apt-get.** Es decir, desde Synaptic se pueden llevar a cabo las mismas acciones que en la línea de órdenes pero utilizando un entorno gráfico más sencillo y amigable. Las principales características de Synaptic son las siguientes:

Instala, elimina, configura, actualiza y descarga paquetes.

Actualiza el sistema completo.

Permite la gestión de los repositorios de paquetes.

Permite comparar la versión instalada de un paquete con la versión disponible en el repositorio correspondiente.

Resolución de dependencias.

Búsqueda de la lista de paquetes utilizando filtros, como pueden ser el nombre, descripción y otras propiedades de los paquetes.

Filtrado, de la lista de paquetes conocidos, por status, sección o propiedades.

Ordenar listados de paquetes por letra inicial, status...

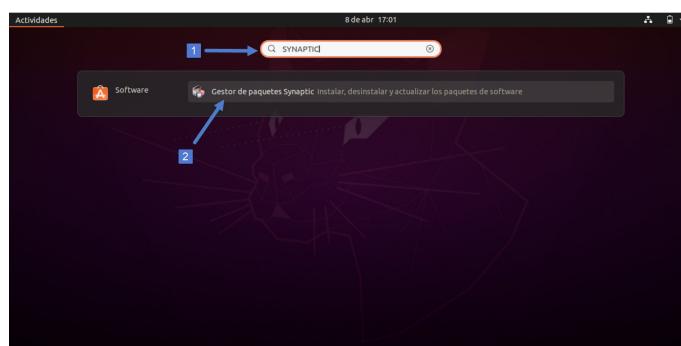
Visualizar toda la documentación en línea referida al paquete.

La utilización de Synaptic requiere tener privilegios de administrador (root) y se puede ejecutar de las siguientes formas:

Desde el escritorio pulsamos en mostrar aplicaciones y escribimos *Synaptic*.



Ubuntu - Captura de pantalla (Elaboración propia)



Ubuntu - Captura de pantalla (Elaboración propia)

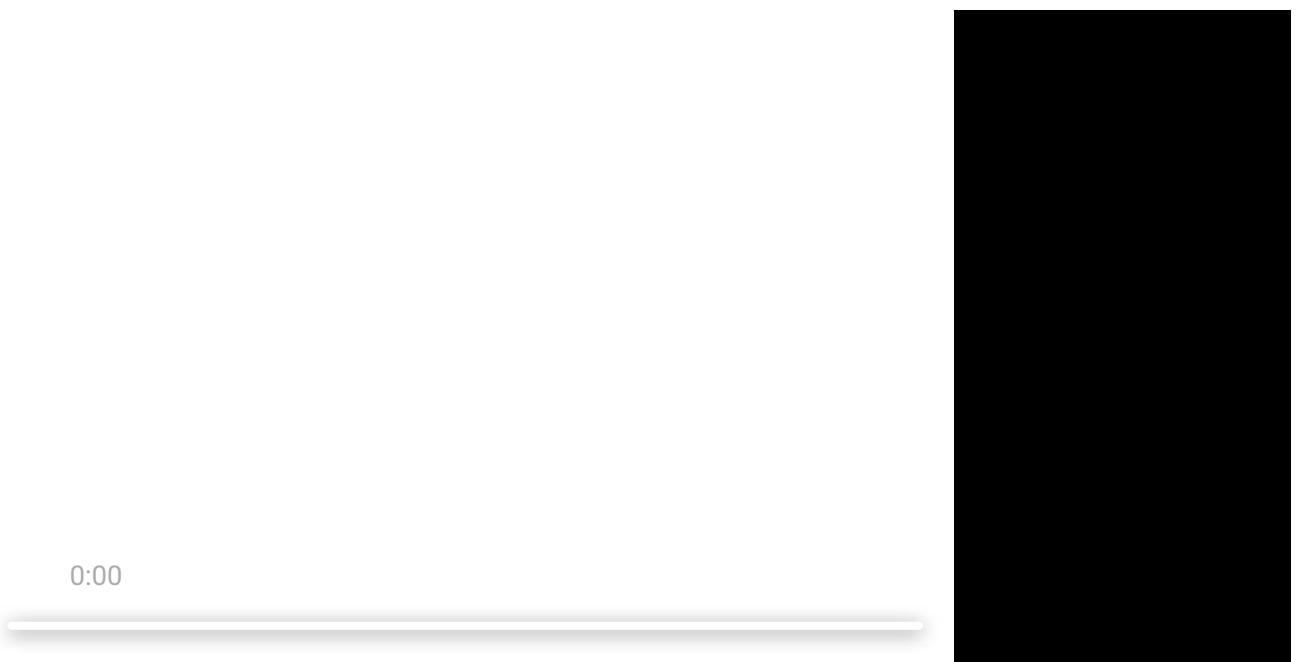
Desde la línea de orden: para arrancar Synaptic desde la línea de orden, en una terminal escribir:

```
prueba@prueba :/home/prueba# synaptic &
```

Ubuntu - Captura de pantalla (Elaboración propia)

El usuario administrador debe tener en cuenta que la utilización de Synaptic es crítica, ya que, en función de las actualizaciones que se hagan, se puede dejar el sistema inestable. Cuando se arranca Synaptic el sistema pide la contraseña del usuario administrador.

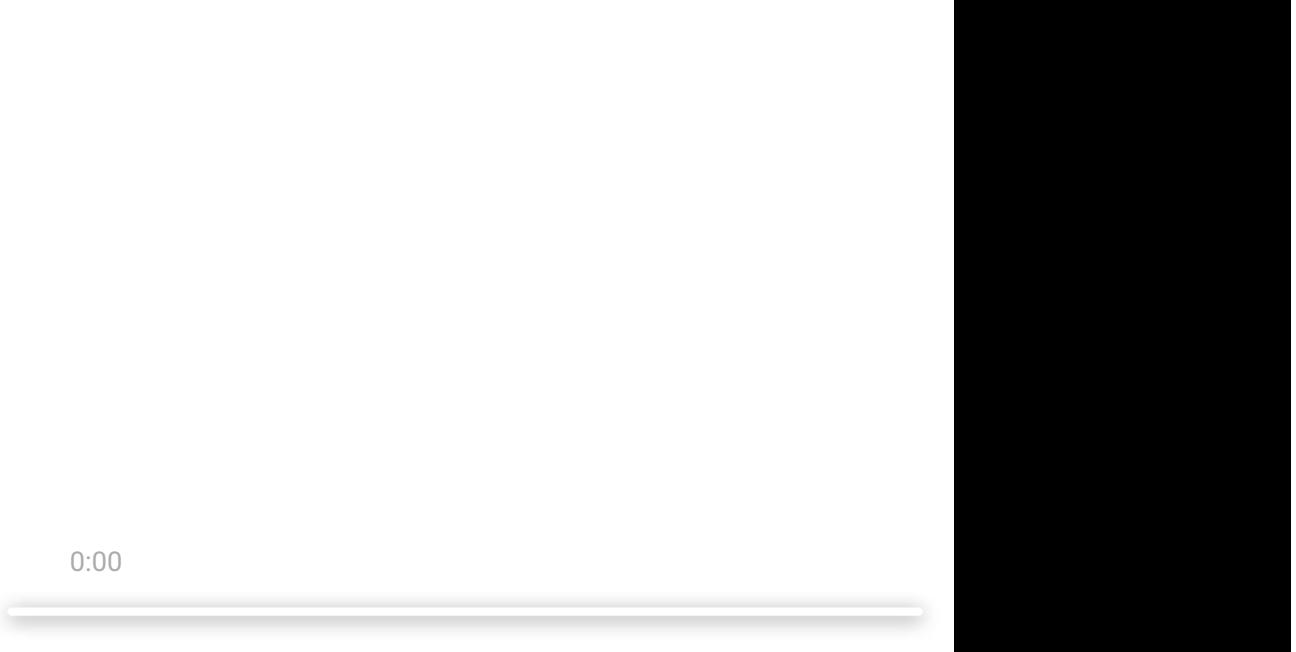
## Gestor de paquetes Synaptic - 1<sup>a</sup> Parte



0:00

Antonio José López Fernández.[Descripción textual alternativa del vídeo "Synaptic"](#) (Elaboración propia)

### Gestor de paquetes Synaptic-2<sup>a</sup> parte



0:00

Antonio José López Fernández.[Descripción textual alternativa del vídeo "Synaptic-2"](#) (Elaboración propia)

## La herramienta dpkg de gestión de paquetes Linux

La orden dpkg es una herramienta de bajo nivel que se utiliza para instalar, eliminar y dar información sobre los paquetes .deb y constituye la base fundamental del sistema de gestión de paquetes de Debian. Es decir, dpkg instala o elimina paquetes y maneja la base de datos (/var/lib/dpkg/) del sistema con la situación de paquetes indicando claramente los cambios. Su ejecución requiere privilegios de administrador (root) del sistema.

Consideraciones para el uso del la orden dpkg:

La instalación de un paquete .deb utilizando dpkg sólo se utiliza cuando se dispone del paquete en local, no a través de la red.

No resuelve las posibles dependencias que se puedan presentar en la instalación del paquete.

Las opciones más importantes en el formato de la orden dpkg son las siguientes:

Opción	Descripción
i <paquete.deb>	Instala un paquete .deb. El usuario debe ser root o ser un usuario sudo.
-L <paquete>	Lista los archivos de un paquete
-l	Lista paquetes instalados
-I <paquete.deb>	Información acerca de un paquete
-s <paquete>	Información del paquete instalado (contenido archivo control ). Admite comodines.
-S <archivo>	Buscar en qué paquete de los instalados está el archivo dado.
dpkg-reconfigure <paquete>	Configura el paquete indicado.
-r <paquete>	Borra el paquete instalado.
--purge <paquete>	Borra el paquete y sus archivos de configuración.
-c <paquete.deb>	Ver los archivos que contiene un paquete binario.

Existen muchas más opciones disponibles para dpkg que podemos ver en la página de su manual con la orden:

```
$man dpkg
```

Ejemplo: para instalar un paquete .deb hay que recordar que se requiere descargar el paquete desde Internet:

```
#dpkg -i nombre_de_paquete.deb
```

Ejemplo: para mostrar los archivos que copia en el sistema la instalación de la aplicación anterior:

```
#dpkg -L nombre
```

El funcionamiento de un paquete puede que dependa de otros. Es decir, su ejecución requiere la presencia en el sistema de otros paquetes. Cada paquete tiene asociado un archivo control donde están documentadas las dependencias de dicho paquete. Existen varios niveles de

dependencia. Como ejemplo utilizamos los paquetes A y B, pretendiéndose instalar el paquete A.

Por ejemplo, si se quiere conocer las dependencias del paquete Totem (aplicación utilizada para la reproducción de audio y vídeo) ejecutar en una terminal:

```
#dpkg -s totem
```

La línea de dependencias se ha subrayado y se comprueba que la instalación de Totem requiere la instalación de paquetes como: libc6, libcairo2, etc.

```
prueba@prueba:~$ dpkg -s totem
Package: totem
Status: install ok installed
Priority: optional
Section: video
Installed-Size: 209
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture: amd64
Version: 3.34.1-2ubuntu2
Depends: libc6 (>= 2.28), libcairo2 (>= 1.14.0), libgdk-pixbuf2.0-0 (>= 2.23.0),
          libglib2.0-0 (>= 2.35.0), libgstreamer-plugins-base1.0-0 (>= 1.6.0), libgstreamer1.0-0
          (>= 1.6.0), libgtk-3-0 (>= 3.19.4), libpango-1.0-0 (>= 1.14.0), libpango-cairo-1.0-0
          (>= 1.14.0), libtotem-plparser18 (>= 3.26.0), libtotem0 (>= 3.34.1-2ubuntu2),
          libtotem0 (<< 3.35), libx11-6, grilo-plugins-0.3-base, totem-common (>=
          3.34.1-2ubuntu2), gsettings-desktop-schemas, gstreamer1.0-clutter-3.0, gstreamer1.0-plugins-base
          (>= 1.6.0), gstreamer1.0-plugins-good (>= 1.10), gstreamer1.0
-x
```

Ubuntu - Captura de pantalla (Elaboración propia)

## La herramienta apt de gestión de paquetes Linux

La orden `apt` (herramienta avanzada de empaquetado) es una herramienta que permite instalar paquetes a través de la red y resolver los conflictos en las dependencias de paquetes que se puedan presentar. Su ejecución requiere privilegios de administrador del sistema. `apt` dispone de dos interfaces en línea de órdenes:

`<strong>apt-get</strong>` es la herramienta en línea de órdenes usada para el manejo de paquetes.

`<strong>apt-cache</strong>` realiza una serie de operaciones sobre la caché de paquetes de `apt`. Permite operaciones de búsqueda en la información de los paquetes.

Las opciones de formato más importantes de la orden `apt-get` son las siguientes:

Opción	Descripción
update	Obtiene una nueva lista de paquetes actualizando las descripciones de los paquetes que hay en la base de datos local.
upgrade	Actualiza el sistema con actualizaciones disponibles. No elimina paquetes previamente instalados, ni instala paquetes previamente no seleccionados.
dist-upgrade	Actualiza todo entre ramas de desarrollo. Con <code>-u</code> vemos todos los paquetes a actualizar.
install	Instala un paquete y sus dependencias.
install --reinstall	Reinstala un paquete ya instalado.

install -f	Para solucionar problemas de dependencias de paquetes ya instalados o a medio instalar.
remove [-purge]	Elimina un paquete (borra archivos configuración).
source	Baja los paquetes fuente (los fuentes, los parches y los archivos de control de Debian) de un paquete de las listas y descomprime.
clean	Los paquetes instalados se guardan en /var/cache/apt/archives/ y este directorio se puede limpiar con esta opción recuperando espacio.
check	Comprobación de posibles problemas en el estado de los paquetes.

Ejemplo: para actualizar la lista de paquetes, no los propios paquetes instalados en el sistema:

Ubuntu - Captura de pantalla (Elaboración propia)

Ejemplo: para instalar nuevas versiones de los paquetes ya instalados en el sistema:

```
# apt-get upgrade
```

Ubuntu - Captura de pantalla (Elaboración propia)

Ejemplo: para instalar un paquete desde la red resolviendo las dependencias que pudiera presentar:

```
# apt-get install totem
```

Ubuntu - Captura de pantalla (Elaboración propia)

Las opciones de formato más importantes de la orden `apt-cache` son las siguientes:

Opción	Descripción
show <paquete>	Muestra la descripción del paquete dado.
search <cadena>	Busca la 'cadena' en la base de datos de paquetes.

stats	Muestra estadísticas sobre apt-cache.
depends	Muestra de qué paquetes depende otro paquete.
showpkg	Muestra información sobre el paquete.

Ejemplo: queremos conocer información sobre un paquete que tenemos instalado:

```
#apt-cache show totem
```

Ejemplo: queremos conocer información sobre un paquete que no tenemos necesariamente instalado (en este caso el paquete gnome-do):

```
#apt-cache search gnome-do
```

Las fuentes de búsqueda del repositorio se encuentran en el archivo /etc/apt/sources.list que indican a dónde se debe ir a buscar los paquetes que se instalan con la orden apt-get install, es decir, las fuentes o lugares que suministran toda las aplicaciones disponibles de la distribución.

Al instalar la variante Ubuntu el sistema dispone del siguiente archivo /etc/apt/sources.list que reproducimos en parte:

```
root@prueba:/# cat /etc/apt/sources.list
#deb cdrom:[Ubuntu 20.04 LTS _Focal Fossa_ - Alpha amd64 (20200324)]/ focal main
restricted

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://es.archive.ubuntu.com/ubuntu/ focal main restricted
# deb-src http://es.archive.ubuntu.com/ubuntu/ focal main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://es.archive.ubuntu.com/ubuntu/ focal-updates main restricted
# deb-src http://es.archive.ubuntu.com/ubuntu/ focal-updates main restricted
```

Ubuntu - Captura de pantalla (Elaboración propia)

Conviene tener siempre el sistema actualizado con las últimas versiones de los paquetes instalados, siempre que correspondan a una versión estable de la distribución. La orden apt-get permite, tanto tener actualizada la lista de paquetes disponibles en el sistema como su actualización a las últimas versiones.

Hay que tener en cuenta que hay que actualizar el sistema siempre que:

Introducimos cambios en el archivo /etc/apt/sources.list  
Queremos actualizar la lista de paquetes del sistema

Para ello hay que ejecutar la orden:

```
#apt-get update
```

Se puede llevar a cabo una actualización general del sistema a la última versión disponible sin necesidad de marcar y actualizar cada paquete de forma manual. Para ello hay dos métodos:

Actualización predeterminada: el método de actualización por defecto actualiza solo los paquetes instalados. La actualización no podrá ser marcada si la última versión del paquete depende de paquetes no instalados o hay conflictos con paquetes ya instalados.  
Actualización inteligente o Dist-Upgrade: Este método intenta resolver conflictos entre paquetes de forma inteligente. Este método desde una terminal es equivalente a la ejecución de la orden:

```
#apt-get dist-upgrade
```

## La herramienta tar de gestión de paquetes Linux

El comando tar es capaz de comprimir y empaquetar paquetes de ficheros y directorios que forman una aplicación. Generalmente las aplicaciones en código fuente vienen en este formato. Será necesario su descompresión y desempaquetamiento para pasar al proceso de enlazar las librerías necesarias para el funcionamiento de la aplicación y la compilación de la aplicación en ejecutable (pasar el código fuente a código binario). Los paquetes fuente suelen estar comprimidos en ficheros con las siguientes extensiones: .tgz (o **tar.gz**) y .bz2. Es conveniente buscar en las fuentes algún fichero de nombre 'readme' o 'install' para ver las instrucciones de instalación, aunque casi siempre se reducen a cuatro pasos.

1.- Descomprimir las fuentes y situarnos en el directorio donde se han descomprimido :

```
# tar zxvf nombre_archivo.tgz
```

2.- Debemos posicionarnos en la carpeta donde hemos descomprimido las fuentes (previamente se habrá creado con el comando mkdir directorio\_ruta):

```
# cd /ruta
```

3.- Tecleamos:

```
# ./configure
```

4.- Compila las fuentes y crea los binarios necesarios. Tecleamos:

```
# make
```

5.- Para instalar el programa tecleamos:

```
# make install
```

Las opciones de formato más importantes del comando tar son:

Opción	Descripción
c	Crear un paquete
t	Comprueba los archivos almacenados en paquete
x	Comprueba los archivos almacenados en paquete
v	Muestra el proceso de empaquetamiento
p	Conserva los permisos de los ficheros
r	Añade archivos al paquete
f	Especifica el nombre del paquete
Z	Comprime o descomprime (compress)
z	Comprime o descomprime (gzip)

Para crear un fichero **.tar**:

```
# tar -cvf ejemploArchivo.tar /home/ejemploArchivo
```

El directorio `/home/ejemploArchivo` es el directorio que necesita ser comprimido creando `ejemploArchivo.tar`

Para crear un fichero **.tar.gz**:

```
# tar -cvzf ejemploArchivo.tar /home/ejemploArchivo
```

La opción z representa compresión **gzip**.

Para descomprimir un fichero **.tar**:

```
# tar -xvf ejemploArchivo.tar
```

Si queremos extraer tus archivos a un directorio diferente, podemos usar la opción **-C**. Aquí tienes un ejemplo:

```
# tar -xvf ejemploArchivo.tar -C /home/Extraerfichero/
```

Para descomprimir archivos **<strong>.tar.gz</strong>**:

```
# tar -xvf ejemploArchivo.tar.gz
```

Si queremos extraer tus archivos a un directorio diferente, podemos usar la opción **-C**. Aquí tienes un ejemplo:

```
# tar -xvf ejemploArchivo.tar.gz -C /home/Extraerfichero/
```

## La herramienta de gestión de paquetes rpm de Linux

Si no encontramos el paquete en formato debian (\*.deb) ni encontramos las fuentes, sino que tan solo lo encontramos en formato \*.rpm (formato usado por otras distribuciones de Linux, como Red Hat, Suse o Mandriva). Rpm permite la instalación, modificación, actualización y borrado de los paquetes del sistema, en el proceso de instalación, realiza comprobaciones de dependencias y versiones registrando los paquetes instalados en la base de datos de RPM. Tiene el problema de que no es capaz por sí sólo de obtener los ficheros de dependencias. El administrador será el encargado de realizar dicha tarea de localizar las librerías e instalarlas previamente al paquete que se quiere instalar.

El problema anteriormente citado se soluciona con herramientas que automatizan el proceso de implantación de aplicaciones en RPM como son: aptrpm, YUM, urpmi, update, Yast2.

Ejemplo: para instalar un paquete en binario llamado aplicación.rpm, tecleamos:

```
# rpm -i aplicación.rpm
```

Ejemplo: para instalar un paquete en código fuente llamado aplicaciónfuente.src.rpm:

Tecleamos:

```
# rpm -ivh aplicaciónfuente.src.rpm
```

Procesar el archivo .spec con:

```
# rpmbuild -bb aplicaciónfuente
```

Esto creará un .rpm en **/usr/src/RPMS/i386** compilado y adaptado al sistema.

Las opciones de formato más importantes del comando rpm son:

Opción	Descripción
--------	-------------

i	Instalar el paquete
v	Muestra información detallada del proceso
U	Permite la actualización de paquetes en el sistema si no existe lo instala
test	Prueba el comando rpm aplicado antes de ejecutarlo
iq	Obtenemos toda la información del paquete
qc	Obtener los archivos de configuración del paquete
qd	Muestra la documentación existente del paquete
qf	Obtenemos un listado de todos los paquetes instalados
af	Conocemos a qué paquete pertenece un determinado archivo
rebuilddb	Reconstruye la base de datos rpm que contiene los paquetes instalados en el sistemas

## La herramienta de gestión de paquetes Gzip y Bzip de Linux

La orden <strong>Gzip</strong> es un comando que permite comprimir, nombre del archivo se identifica con la extensión .gz y debe de ser descomprimido antes de ser usado. Para comprimir más de un archivo se empaqueta antes con el comando tar (la versión última del comando tar permite comprimir con el parámetro z).

Ejemplo: para comprimir el archivo documento.doc

```
# gzip documento.doc
```

El resultado generará un archivo llamado documento.doc.gz, y el fichero documento.doc será sustituido por documento.doc.gz

```
# gzip -d documento.doc.gz
```

También podemos utilizar el comando:

```
# gunzip documento.doc.gz
```

La orden <strong>Bzip </strong>permite comprimir archivos. Comprime más que gzip (el algoritmo utilizado es más completo). La extensión que crea es .bz2

Ejemplo: para comprimir el archivo documento.doc

```
# bzip2 documento.doc
```

El resultado generará un archivo llamado documento.doc.bz2, el fichero documento.doc será sustituido por documento.doc.gz

Ejemplo: para descomprimir el fichero creado en el ejemplo anterior.

```
# bzip2 -d documento.doc.bz2
```

También podemos utilizar el comando:

```
# gunzip documento.doc.bz2
```

## La herramienta de gestión de paquetes Zip de Linux

La orden Zip comprime y descomprime archivos permitiendo la transparencia y portabilidad con otros entornos de sistemas como Windows. Con el comando Zip se comprime y con unzip se descomprime.

Ejemplo: para comprimir todos los archivos que se encuentran en el directorio actual en un fichero llamado copia.

```
# zip backup *
```

El \* se utiliza como comodín y representa a todos los archivos.

Ejemplo: para comprimir el archivo /home/documentos/examen.doc y /home/documentos/apuntes.doc en un fichero llamado copia.

```
# zip copia /home/documentos/examen.doc /home/documentos/apuntes.doc
```

Ejemplo: para descomprimir el archivo copia generado en los ejemplos anteriores.

```
# unzip backup.zip
```

Ejemplo: para descomprimir el archivo copia en el directorio /home/documentos/ejemplos/ generado en los ejemplos anteriores.

```
# unzip backup.zip -d /home/documentos/ejemplos/
```

## Autoevaluación

**Indicar si la siguiente afirmación es verdadera o falsa:**

El comando tar es capaz de descomprimir y desempaquetar aplicaciones para poder pasar a realizar el proceso de linkaje y compilación con el fin de crear de los ficheros fuentes el fichero ejecutable de la aplicación.

Sugerencia

- Verdadero  Falso

**Verdadero**

El comando tar es capaz de comprimir y empaquetar paquetes de ficheros y directorios que forman una aplicación.

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 2.- Actualización de Sistemas Operativos y aplicaciones: Instalación de parches.

### Caso práctico

**Noiba, Naroba y Jana** continúan con la tarea encomendada y entienden que los sistemas operativos y aplicaciones necesitan de un sistema de actualización automatizado por el propio sistema para mejorar las prestaciones de seguridad y servicios. Deben comprobar como gestiona el sistema operativo las actualizaciones sin que le resulten costosas en tiempo y forma.

—En Windows, la aplicación que se encarga de las actualizaciones es Windows Update —afirma **Jana**.

—Sería interesante investigar como funciona Windows Update, —añade **Naroba**.

—Buena propuesta, —dice **Noiba**.

### Actualización de sistemas operativos y aplicaciones en Windows

Para solucionar problemas de errores del sistema, mejorar la la seguridad de posibles vulnerabilidades críticas (agujeros de seguridad) y de mejoras de rendimiento (por ejemplo actualizar los controladores del ordenador), Microsoft desarrolla parches que arreglan estos problemas. Los parches se pueden instalar automáticamente por Internet mediante la aplicación Update o se pueden descargar en modo fichero denominado Service Pack del portal oficial de Internet o instalar desde CD u otro dispositivo de almacenamiento.

Con el programa de instalación de la propia aplicación se puede :

Instalar cualquier aplicación incluida en el paquete de software.

Añadir componentes a aplicaciones actuales instaladas.

Actualizar los archivos y configuraciones de las aplicaciones instaladas.

Actualizar a una nueva versión de distribución, siempre que las aplicaciones instaladas lo permitan, es decir, sean compatibles.

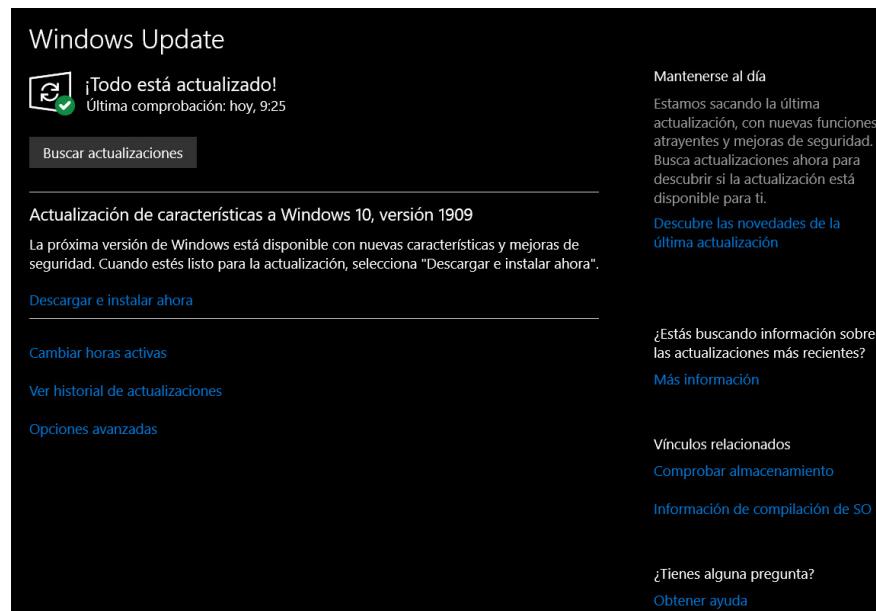
La actualización de aplicaciones dispone del sistema de actualización online (con conexión a Internet) que permite modificar los programas directamente con el fabricante desde Internet. Normalmente se configura para que nos avise mediante una ventana de diálogo si deseamos realizar la actualización o no.

Para **actualizar aplicaciones en Windows**, podemos realizarlo desde la herramienta de actualización que aporta Windows:

*Inicio-Panel de control-Programas.*

Aparece la lista de programas instalados. Seleccionar un programa y, a continuación, hacer clic en *Desinstalar o cambiar*. Algunos programas incluyen la opción de cambiar o reparar el programa además de desinstalarlo, pero muchos solo ofrecen la opción de desinstalación. Para cambiar un programa, hacer clic en Desinstalar o cambiar.

**En el sistema operativo se activa la actualización automática del sistema para mejoras de seguridad y de rendimiento mediante los denominados Service Pack aportados por el distribuidor Microsoft, se necesita conexión a Internet.** Dicha utilidad se encuentra en *Inicio-Configuración-Windows Update*



Microsoft - Captura de pantalla (Elaboración propia)

Cada opción realiza lo siguiente:

*Windows Update*: aparece una ventana con una serie de opciones. Las tareas que se pueden realizar son:

*Buscar actualizaciones*: realiza un proceso de consulta de las actualizaciones disponibles para nuestro sistema, notificándonos del número e invitándonos a realizar la actualización.

*Ver el historial de actualizaciones*: aparece un listado de las actualizaciones realizadas en el equipo con informes del nombre, fecha y situación (en caso de informar error conviene volver a actualizar). Podemos realizar varias acciones:

Desinstalar las actualizaciones.

Opciones de recuperación: podemos restablecer el ordenador, reiniciar el equipo e iniciar en modo avanzado y otras opciones de recuperación.

*Opciones avanzadas*: podemos activar o desactivar diferentes opciones como: ofrecer actualizaciones para productos de Microsoft, descargar automáticamente las actualizaciones, avisar cuando el sistema se vaya a reiniciar, pausar actualizaciones y elegir cuando se instalarán las actualizaciones.

*Cambiar horas activas*: establece las horas activas para que sepamos cuándo sueles usar el dispositivo y el ordenador no se reiniciará automáticamente durante estas horas sin comprobar que se está usando.

A la derecha tenemos una columna que nos muestra información sobre las actualizaciones: novedades de la última actualización, mas información, comprobar almacenamiento, etc.

Para instalar actualizaciones de Windows sin conexión a Internet, deberemos de disponer de las actualizaciones en un medio de almacenamiento, como puede ser un CD/DVD, pendrive, por red o cualquier otro dispositivo de almacenamiento desde otro ordenador conectado a Internet. En la dirección [Catálogo de Microsoft Update](#) podemos buscar la actualización que deseemos.

Pasos para descargar las actualizaciones de Windows para instalarlas offline desde el catálogo de Microsoft Update:

Abrimos el catálogo de Microsoft Update: [Catálogo de Microsoft Update](#)



Microsoft - Captura de pantalla (Elaboración propia)

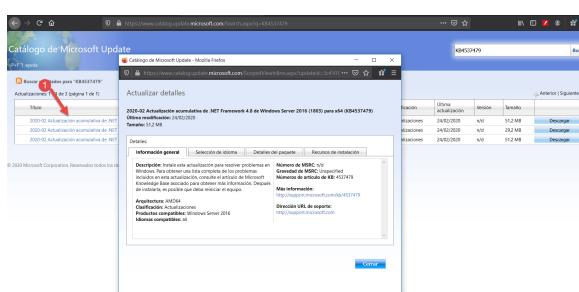
Escribimos el nombre de la actualización, por ejemplo: **KB4537479** en el cuadro de dialogo y pulsamos en el botón *buscar*.

Nos aparecen los resultados de la búsqueda.



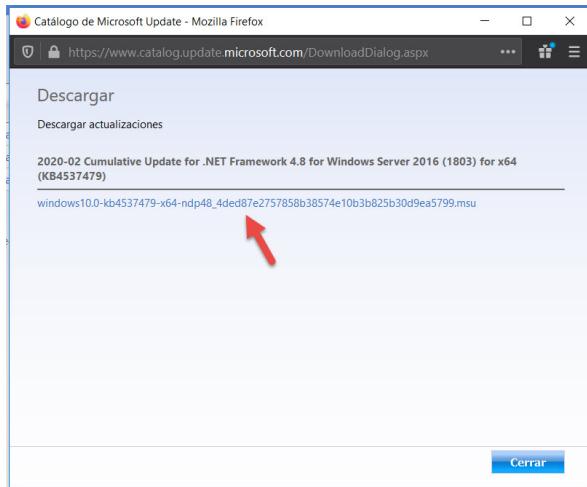
Microsoft - Captura de pantalla (Elaboración propia)

Podemos pulsar sobre la actualización, para ver la descripción de la misma.



Microsoft - Captura de pantalla (Elaboración propia)

Si hemos abierto la ventana de información anterior, pulsamos en el botón cerrar. Pulsamos en descargar. Haremos clic en el enlace que nos aparece para que comience la descarga.



Microsoft - Captura de pantalla (Elaboración propia)

Esperamos a que termine la descarga. Ya podemos utilizarla para actualizar cualquier ordenador sin necesidad de conexión a Internet.

Antes de que se instale una actualización o parche deberemos seguir las siguientes instrucciones:

Asegurarse de que ha descargado e instalado las actualizaciones anteriores (esto lo podemos comprobar en el historial de actualizaciones).

Instalar las últimas actualizaciones para Windows desde el sitio web de Windows Update.

Comprobar si el equipo está ejecutando la versión de 32 o de 64 bits de Windows.

**Para actualizar el equipo de una versión de Windows a otra Windows** podemos realizarla haciendo un **upgrade** (actualización), donde se migrarán todos nuestros datos, aplicaciones, configuraciones, etc. Pero es recomendable realizar un backup donde guardar todos nuestros datos, información importante, etc, para evitar que se pierda cualquier información importante. Para realizar esta actualización tendríamos que tener un equipo con Windows 7 con una licencia y con sus actualizaciones.

Para realizar una actualización de Windows 7 a Windows 10 hacemos lo siguiente:

Entramos en esta página [Actualización de Windows 7 a Windows 10](#) que nos abre un asistente que nos guía en la actualización a Windows 10.

Pinchamos en el botón *Descargar ahora la herramienta*. La ejecutamos y comprueba si nuestro ordenador cumple con los requisitos para poder actualizar a Windows 10.

Aceptamos los términos de licencia.

Nos pregunta ¿Qué deseamos hacer?

Actualizar este equipo ahora: elegimos esta opción si la actualización la vamos a realizar en el mismo ordenador.

Crear medio de instalación (unidad flash USB, DVD o archivo ISO) para otro ordenador. Esta opción se elige si queremos crear un medio de instalación para realizarla en otro equipo.

Preparado el equipo para la instalación, nos muestra un pantalla con lo que va a realizar: instalar Windows 10 y conservar los archivos personales y las aplicaciones. Nos aparece también el enlace *Cambiar los elementos que se van a conservar*, esta opción nos permite cambiar aquellos elementos que deseamos conservar. Pulsamos en el botón *Instalar*.

Empieza el proceso de instalación (se reiniciara varias veces durante el proceso).

Termina el proceso de instalación y puedes iniciar sesión con la misma cuenta que tenías en Windows 7. Tendríamos activado Windows 10, conserva configuraciones como el mismo nombre de equipo y otras configuraciones.

## Para saber más

La dirección de Microsoft donde puedes obtener información sobre Windows Update para el Sistema Operativo Windows 10:

[Windows Update: preguntas frecuentes](#)

Configurar Windows Update:

[Configurar Windows Update](#)

Service Pack y centro de actualizaciones:

[Service Pack y centro de actualizaciones](#)

Migrar un equipo de Windows 7 a Windows 10:

[Actualizar un equipo de Windows 7 a Windows 10](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 2.1.- Actualización de sistemas operativos y aplicaciones en Linux.

Linux mediante la actualización del su núcleo o Kernel realiza la corrección de errores y agrega nuevas mejoras en el rendimiento del sistema. Cada versión soporta un número determinado de versiones de núcleo. Debemos considerar que cada núcleo se adapta a los requisitos de hardware y que cada aplicación está diseñada para un determinado Kernel, es decir, que si modificamos el núcleo también hay que actualizar las aplicaciones.

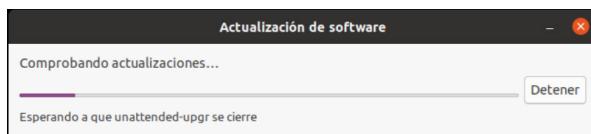
La distribución Ubuntu dispone en su entorno gráfico de un sistema de actualización automática del sistema que lanza la aplicación **Update Manager** (<i>/usr/bin/update-manager</i>). Para ello:

Pulsamos en *Mostrar aplicaciones* y escribimos actualizaciones. Nos aparece el icono de *Actualización de Software*.



Ubuntu - Captura de pantalla (Elaboración propia)

Esperamos mientras comprueba las actualizaciones.



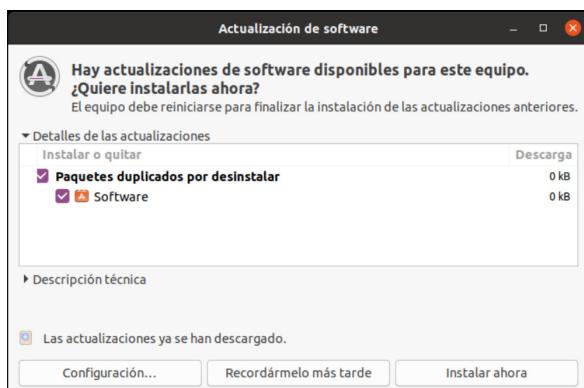
Ubuntu - Captura de pantalla (Elaboración Propria)

Se abre la ventana que nos muestra las actualizaciones disponibles. Si pulsamos en:

*Instalar ahora*: instalamos ahora mismo las actualizaciones.

*Recordármelo más tarde*: se instalan mas tarde.

*Configuración*: se abre la pantalla donde podemos configurar como se realizan las actualizaciones, como se notifican nuevas versiones del sistema operativo y cuando se comprueban las actualizaciones en otras cosas.



Ubuntu - Captura de pantalla (Elaboración propia)

Esta aplicación detecta la disponibilidad de nuevas versiones de los paquetes instalados y propone la actualización de la lista de paquetes así como la descarga de estas nuevas versiones. La aplicación **Update Manager** no puede ser ejecutada si existe lanzada otra instancia de **Synaptic** y pedirá que se cierre una de ellas. A partir de este momento la actualización es inmediata y, dependiendo del volumen de paquetes a actualizar y de la velocidad de la conexión a Internet, tardará cierto tiempo.

**El archivo /etc/apt/sources.list indica a dónde se debe ir a buscar los paquetes que se instalan** con la orden <i>apt-get install</i>, es decir, las fuentes o lugares que suministran toda las aplicaciones disponibles de la distribución y los paquetes que actualizan el sistema operativo. Al instalar la variante Ubuntu el sistema dispone del siguiente archivo /etc/apt/sources.list que reproducimos en parte:

```
root@prueba:/# cat /etc/apt/sources.list
#deb cdrom:[Ubuntu 20.04 LTS _Focal Fossa_ - Alpha amd64 (20200324)]/ focal main
restricted

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://es.archive.ubuntu.com/ubuntu/ focal main restricted
# deb-src http://es.archive.ubuntu.com/ubuntu/ focal main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://es.archive.ubuntu.com/ubuntu/ focal-updates main restricted
# deb-src http://es.archive.ubuntu.com/ubuntu/ focal-updates main restricted
```

Ubuntu - Captura de pantalla (Elaboración propia)

Recordar que las líneas que empiezan con la almohadilla "#" están comentadas y no se procesan.

Conviene tener siempre el sistema actualizado con las últimas versiones de los paquetes instalados, siempre que correspondan a una versión estable de la distribución. La orden <i>apt-get</i> permite tanto tener actualizada la lista de paquetes disponibles en el sistema como su actualización a las últimas versiones.

Hay que tener en cuenta que hay que actualizar el sistema siempre que:

Introducimos cambios en el archivo /etc/apt/sources.list  
Queremos actualizar la lista de paquetes del sistema

También desde el modo comando (entrando a una consola de edición de comandos) podemos actualizar el sistema mediante la orden apt-get, siendo usuario administrador o root, para ello hay que ejecutar la orden:

Ubuntu - Captura de pantalla (Elaboración propia)

Se puede llevar a cabo una actualización general del sistema a la última versión disponible sin necesidad de marcar y actualizar cada paquete de forma manual. Para ello hay dos métodos:

Actualización predeterminada: el método de actualización por defecto actualiza solo los paquetes instalados. La actualización no podrá ser marcada si la última versión del paquete depende de paquetes no instalados o hay conflictos con paquetes ya instalados.

Actualización inteligente o *Dist-Upgrade*: el método dist-upgrade intenta resolver conflictos entre paquetes de forma inteligente. Este método desde una terminal es equivalente a la ejecución de la orden:

```
root@prueba-VirtualBox:/home/prueba# apt-get dist-upgrade
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
root@prueba-VirtualBox:/home/prueba#
```

Ubuntu - Captura de pantalla (Elaboración propia)

Las opciones más importantes de apt-get referentes a la actualización del sistema son las siguientes:

Opción	Descripción
<span style="width: 10.402%;" lang="en">update</span>	Obtiene una nueva lista de paquetes actualizando las descripciones de los paquetes que hay en la base de datos local.
<span style="width: 10.402%;" lang="en">upgrade</span>	Actualiza el sistema con actualizaciones disponibles. No elimina paquetes previamente instalados, ni instala paquetes previamente no seleccionados.
dist-upgrade	Actualiza todo entre ramas de desarrollo. Con -u vemos todos los paquetes a actualizar.

Ejemplo: para actualizar el sistema e instalar nuevas versiones de los paquetes ya instalados en el sistema (la lista de paquetes, no los propios paquetes):

Ubuntu - Captura de pantalla (Elaboración propia)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 2.1.1.- Actualizar el núcleo o Kernel de Linux.

---

Para saber que versión de Kernel tenemos en nuestro sistema podemos escribir el siguiente comando en la terminal:

```
prueba@prueba-VirtualBox:~$ uname -a  
Linux prueba-VirtualBox 5.4.0-21-generic #25-Ubuntu SMP Sat Mar 28 13:10:28 UTC  
2020 x86_64 x86_64 x86_64 GNU/Linux
```

Ubuntu - Captura de pantalla (Elaboración propia)

También lo podemos ver usando el siguiente comando:

```
prueba@prueba-VirtualBox:~$ uname -mrs  
Linux 5.4.0-21-generic x86_64
```

Ubuntu - Captura de pantalla (Elaboración propia)

Otra opción es usando el comando:

```
# cat /proc/version
```

El resultado que muestra puede ser el siguiente o parecido según la versión:

```
Linux 5.8.0-53-generic x86_64
```

Donde:

**Linux** es el nombre del Kernel

**5.8.0-53-generic** es el número de versión del Kernel

**x86\_64** es la arquitectura de nuestro ordenador.

Hay que tener en cuenta que:

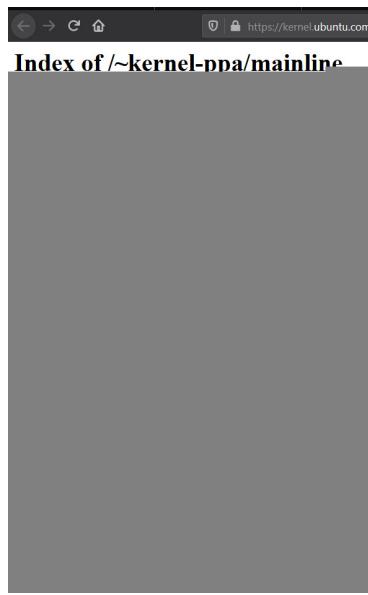
32 bits = i386

64 bits = x86\_64

### Actualizar el Kernel de Linux

Para actualizar el Kernel de Linux en modo consola deberemos de seguir los siguientes pasos:

Ir a la pagina de Ubuntu donde se encuentran los .deb del kernel: [Descargar los Kernel de Linux](#)



Captura de pantalla (Elaboración propia)

Hay que tener en cuenta que las versiones que terminan en rc (ejemplo: v5.4-rc1/) es una versión en pruebas, por lo tanto no es una versión estable.

Elegir la versión de Kernel a la que se desea actualizar. Vamos a elegir la versión **v5.4.1** para la arquitectura de 64 bits (BUILD.LOG.amd65). Hacemos clic en el enlace de la versión elegida. Tenemos que descargar cuatro paquetes:

Si nuestra máquina tiene una arquitectura de 32 bits, tendríamos que elegir los paquetes que estén debajo de BUILD.LOG.i386.

<span style="color: #3366ff;">> linux-headers-5.4.1-050401\_5.4.1-050401.201911290555\_all.deb</span>

<span style="color: #3366ff;">> linux-headers-5.4.1-050401-generic\_5.4.1-050401.201911290555\_amd64.deb</span>

<span style="color: #3366ff;">>linux-image-unsigned-5.4.1-050401-generic\_5.4.1-050401.201911290555\_amd64.deb</span>

<span style="color: #3366ff;">>linux-modules-5.4.1-050401-generic\_5.4.1-050401.201911290555\_amd64.deb</span>

## v5.4.1 mainline build

These binary packages represent builds of the mainline or stable Linux kernel tree at the commit below:

v5.4.1 (794438f37a9a0822333c86acb06a71abc1ce)

To obtain the source from which they are built fetch the commit below:

[git://git.launchpad.net/~ubuntu-kernel-test/ubuntu/+source/linux/+git/mainline-crack](https://git.launchpad.net/~ubuntu-kernel-test/ubuntu/+source/linux/+git/mainline-crack) v5.4.1

and apply the following patches on top in the order below:

0001-base-packaging.patch  
0002-UBUNTU-SAUCE-add-vmlinuz.strip-to-BOOT\_TARGETS1-on-p.patch  
0003-UBUNTU-SAUCE-add-vmlinuz-add-manual-page.patch  
0004-debian-changelog.patch  
0005-config-based-on-Ubuntu-5.4.0-7.8.patch

This is summarised in the [SOURCES](#) file included with these binaries.

Build for amd64 succeeded (see [BUILD.LOG.amd64](#)):

linux-headers-5.4.1-050401-generic\_5.4.1-050401.201911290555\_all.deb

linux-headers-5.4.1-050401-generic\_5.4.1-050401.201911290555\_amd64.deb

linux-headers-5.4.1-050401-lowlatency\_5.4.1-050401.201911290555\_amd64.deb

linux-image-unsigned-5.4.1-050401-generic\_5.4.1-050401.201911290555\_amd64.deb

linux-image-unsigned-5.4.1-050401-lowlatency\_5.4.1-050401.201911290555\_amd64.deb

linux-modules-5.4.1-050401-generic\_5.4.1-050401.201911290555\_amd64.deb

linux-modules-5.4.1-050401-lowlatency\_5.4.1-050401.201911290555\_amd64.deb

Build for armhf succeeded (see [BUILD.LOG.armhf](#)):

linux-headers-5.4.1-050401-generic\_5.4.1-050401.201911290555\_all.deb

linux-headers-5.4.1-050401-generic\_5.4.1-050401.201911290555\_armhf.deb

linux-headers-5.4.1-050401-generic\_5.4.1-050401.201911290555\_armhf.deb

linux-image-unsigned-5.4.1-050401-generic\_5.4.1-050401.201911290555\_armhf.deb

linux-image-unsigned-5.4.1-050401-lowlatency\_5.4.1-050401.201911290555\_armhf.deb

linux-modules-5.4.1-050401-generic\_5.4.1-050401.201911290555\_armhf.deb

linux-modules-5.4.1-050401-lowlatency\_5.4.1-050401.201911290555\_armhf.deb

Ubuntu - Captura de pantalla (Elaboración propia)

Una vez descargados los paquetes. Nos vamos a la carpeta donde los hemos descargado. Utilizamos la herramienta dpkg para instalar los paquetes anteriores. Ejecutamos el siguiente comando:

```
sudo dpkg -i linux*.deb
```

Con el comando anterior se desempaquetan y se instalan todos los paquetes. Se reinicia el sistema. Para que entre en funcionamiento la nueva versión del Kernel.

**Una actualización de versión (upgrade) es el proceso de pasar de una versión anterior de Ubuntu a una nueva.** Saltarse versiones no es recomendable y puede causar muchos problemas en la instalación. Si estás utilizando una versión antigua y quieres "saltar" una versión, la única manera segura es hacer una copia de seguridad de tus datos y hacer una instalación limpia, o una actualización progresiva a las sucesivas versiones. Una manera fácil de comprobar la compatibilidad de una nueva versión de Ubuntu en tu máquina es probando el Desktop CD antes de actualizar.

Generalmente, se recomienda que se instale la ultima versión de Ubuntu ya que siempre se añade nuevo hardware soportado, y normalmente optimizado y más rápido. Para saber qué versión de Ubuntu tienes instalada, ejecuta en una terminal el siguiente comando:

```
lsb_release -a
```

## Autoevaluación

**¿Cuál es el fichero donde se encuentran las fuentes de repositorios en Ubuntu?.**

### Sugerencia

- /usr/bin/update-manager.
- /etc/apt-get update.
- /etc/apt/sources.list
- /etc/dev/sources.list

**Incorrecto! Inténtalo otra vez.**

**Incorrecto!** Este comando permite actualizar la lista de paquetes.

**Correcto.** Muy bien, se nota tu progreso.

**Incorrecto!** No es el directorio correcto.

## Solución

1. Incorrecto
2. Incorrecto
3. Opción correcta
4. Incorrecto

## Debes conocer

Podemos actualizar el Kernel de Linux, utilizando una aplicación gráfica como es Ukuu:

[Como actualizar el Kernel de Linux](#)

## Para saber más

Cómo actualizar el Kernel en Ubuntu con comandos y de forma gráfica usando Ukuu:

[Actualizar el Kernel de Linux con comandos](#)

### 3.- Proceso de arranque: Ficheros de inicio de Sistemas Operativos.

#### Caso práctico

A **Jana** le surge la siguiente duda, que pregunta a Vindio:  
Actualmente, ¿Es importante el papel que juega la secuencia de arranque en los sistemas operativos?.

Es una buena pregunta. Es fundamental conocer el proceso de arranque de los sistemas operativos, pues este influye en su administración y configuración —añade **Vindio**.

Sería muy interesante que investigaras. Esto nos permitirá conocer los servicios, utilidades y programas que forman parte del proceso de arranque de los sistemas operativos —comenta **Jana**.

[Alain Bachellier \(CC BY-NC-SA\)](#)

#### La secuencia de arranque de Windows 10 y Windows Server

**La secuencia de arranque de Windows 10** utiliza un nuevo sistema de almacenamiento y configuración de arranque denominado **BCD store** que sustituye al boot.ini. Este registro se encuentra situado en el directorio \boot\BCD de la partición donde se instala de forma oculta y en binario. Consiste en un repositorio de datos y parámetros necesarios para gestionar el arranque del sistema. También el fichero ntldr se sustituye por **Bootmgr.exe** pasando a ser el encargado del manejador y cargador de arranque (boot loader). La secuencia de arranque sigue los siguientes pasos

- 1.- La BIOS carga el MBR en memoria el cual analiza la tabla de particiones y carga en memoria el sector de arranque de la partición marcada como activa en la que se encuentra el sistema operativo Windows 10.
- 2.- El sector de arranque localiza el bootmgr (el cargador de vista) y se ejecuta, busca el directorio \boot que contiene los programas que inician el sistema.
- 3.- Dentro del directorio se ejecuta el BCD apareciendo un menú en que se selecciona el sistema que se desea arrancar comenzando la carga del núcleo de sistema.

Para poder configurar el BCD de Windows 10 se dispone de la herramienta BCDEdit que permite manipular las opciones de arranque del sistema, reemplaza la utilizada para Windows XP llamada bootcfg. Para utilizarlo debemos abrir una consola ejecutando el comando cmd y ejecutar el programa **<i>BCDEdit.exe</i>** ejecutando en la línea de comandos de la consola la orden **bcdedit**, aparecerá un listado con todas las opciones del menú de arranque con sus correspondientes parámetros de configuración. Para obtener una ayuda sobre su manejo podemos ejecutar la orden:

```
bcddedit.exe /? ó bcddedit.exe /? nombre de un comando
```

Por ejemplo si deseamos cambiar en el sistema operativo que arranque por defecto debemos modificar del listado que genera el menú el parámetro Identifier que es un número en hexadecimal denominado GUID que aparece entre llaves. Seguidamente anotamos el Identifier que nos interesa y tecleamos:

```
bcddedit /default GUID
```

siendo GUID el identificador de la entrada del sistema que se quiere ejecutar por defecto.

Existen programas como **EasyBCD** que te permite editar el cargador de arranque (Windows Boot Manager) y agregar todas las entradas que quieras. Te permite modificar, agregar entradas e incluso arreglar el sector de arranque. Detecta automáticamente los sistemas operativos instalados y lista las entradas de tu cargador de arranque. Podemos añadir nuevas entradas para otras versiones de Windows, Linux, BSD y Mac OS.

**En la secuencia o proceso de arranque Windows Server 2019** debemos de considerar los siguientes aspectos:

El entorno de arranque es compatible con los modelos de firmware BIOS y UEFI. Utiliza un almacén para los datos de configuración de inicio (BCD) que sustituye a Boot.ini y proporciona aplicaciones de arranque que sustituyen a los anteriores Windows (el <i>loader Ntldr.exe</i>) con un nuevo administrados y gestor de arranque de Windows. Utiliza el mismo sistema de arranque que Windows 10.

Al encender un equipo el ordenador funciona con el software de inicio que reside en firmware, llamado *boot loader* que localiza e inicia el núcleo del sistema operativo y prepara el acceso a los dispositivos de hardware antes de que el ordenador esté listo para iniciar la ejecución de aplicaciones.

El gestor de arranque suele ser independiente del sistema operativo. Los administradores del sistema suelen cambiar la configuración de arranque. Para cambiar la configuración de inicio, ejecutamos el arranque Bvdedit.exe que configura la aplicación en una línea de comandos; para evitar riesgos de mal funcionamiento antes de realizar cambios es aconsejable hacer una copia de seguridad de la configuración de arranque actual mediante el uso de la orden <i>bcddedit</i> <i>/exportación</i> de guardar BCD.

Los datos de configuración de arranque (BCD) sustituyen a la anterior (boot.ini). En el almacén BCD el gestor de arranque se presenta como objeto de programa (GUID) en lugar de elementos de texto. Con la herramienta Bcdedit.exe podemos utilizar los comandos básicos para controlar todos los aspectos del proceso de arranque, aunque cada objeto se presenta con un GUID, algunos objetos tienen nombres de alias para uso común, como bootmgr (referido al gestor de arranque) y **default** (gestor de arranque por defecto). Podemos modificar la configuración de inicio con WMI. Podemos utilizar MSconfig.exe para proporcionar una interfaz gráfica para visualizar y modificar un subconjunto de los ajustes de configuración de inicio

Podemos realizar el siguiente **resumen del proceso de arranque**:

- 1.- Se inicia la BIOS
- 2.- Se carga el MBR del disco
- 3.- Carga el sector de arranque
- 4.- Carga el Windows Boot Manager
- 5.- Lee desde el BCD
- 6.- Busca archivos de hibernación
- 7.- Inicia el programa Winload.exe, (carga el kernel del Sistema operativo)
- 8.- Inicia el programa ntoskrnl.exe, (imagen del kernel de Windows)
- 9.- Inicia el programa smss.exe, (administrador de secciones)
- 10.- Inicia el programa winlogon (valida la identificación del usuario en el sistema)

11.- Finaliza iniciando los servicios y la interfaz de login.

**Se puede editar el arranque en Windows 2019 ejecutando el comando msconfig en la pestaña Arranque, modificando las opciones de arranque.** El administrador y cargador del sistema de arranque se edita con el comando bcdedit cuyos comandos se listan:

```
bcdedit /?
```

## Debes conocer

Como usar BCDEdit en Windows 10

[Usar BDCDEDIT en Windows 10](#)

## Para saber más

Ejemplo de uso de EASYBCD:

[Uso de la aplicación EASYBCD](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 3.1.- Variables de entorno generadas y ficheros que actúan en el arranque de Windows.

---

Una **variable del entorno** es un valor dinámico cargado en la memoria, que puede ser utilizado por varios procesos que funcionan simultáneamente. En la mayoría de los sistemas operativos, la ubicación de algunas bibliotecas o de los archivos ejecutables del sistema más importantes puede variar según la instalación. Podemos considerar su estudio en este momento ya que cuando se genera el arranque de los sistemas a la vez se van cagando en memoria una serie de valores en las llamadas variables del sistema o de entorno. Por eso es posible, para un programa dado, remitirse a una ubicación basada en las variables del entorno que definen estos datos.

Para ver todas las variables de entorno lo podemos hacer de dos formas:

Usando PowerShell: Abrimos una consola de PowerShell ISE (x86) en modo administrador y escribimos el siguiente comando:

```
Get-ChildItem Env:
```

Microsoft - Captura de pantalla (Elaboración propia)

Usando CMD: abrimos la consola de símbolo del sistema de Windows en modo administrador y escribimos el comando:

```
set
```

Microsoft - Captura de pantalla (Elaboración propia)

## Trabajar con las variables de entorno usando CMD

En Windows, las variables del entorno se ubican entre los caracteres "%". De esta forma, para mostrar el valor de una variable del entorno sólo se debe escribir el siguiente tipo de comando:

```
echo $NAME_DE_VARIABLE%
```

Ejemplo:

```
echo %HOMEDRIVE%
```

Devuelve:

C:

Devuelve la unidad en la que está el directorio en el que estás actualmente.

En Windows, el comando <i>set</i> permite crear, modificar y mostrar las variables del entorno. Sin embargo, los cambios se efectúan sólo en la sesión en progreso.

Algunas de las operaciones que se pueden realizar con variables del sistema son:

Para que se muestre una variable se utiliza el siguiente comando:

```
set Nombre_de_variable
```

Ejemplo:

```
set USERNAME
```

Nos muestra el usuario con el que hemos iniciado sesión.

Para crear una variable se utiliza este comando:

```
set Nombre_de_variable=valor
```

Ejemplos:

```
set var=10
```

```
set colour=black^&white
```

```
set varname="new&name"
```

Para eliminar una variable se utiliza este comando (el signo igual es el que indica la eliminación):

```
set NombreVariable=
```

Se puede utilizar cualquier carácter como valor, incluso espacios. No obstante, para poder utilizar caracteres especiales (<, >, |, & o ^), simplemente se debe introducir el carácter escape (^) delante de éstos o comillas a su alrededor, con lo que formarán parte del valor debido a que se tomarán en cuenta todos los caracteres después del signo =. Por ejemplo, para definir "negroyblanco" como valor:

```
set colour=black^&white
```

## Trabajar con las variables de entorno usando PowerShell

Algunas de las operaciones que se pueden realizar sobre las variables de entorno en PowerShell son:

Mostrar el contenido de una variable de entorno:

```
Write-host $env:nombre_variable_entorno
```

### Ejemplo:

```
Write-host $env:windir
```

Mostramos el contenido de la variable de entorno **windir** (muestra el directorio de instalación de Windows, c:\windows)

También lo podemos hacer de esta forma:

```
Get-ChildItem Env:\nombre_variable_entorno
```

### Ejemplo:

```
Get-ChildItem Env:\windir
```

Crear una variable de entorno:

```
$env:nombre_variable="Contenido de la variable"
```

### Ejemplo:

```
$env:cadena="Ejemplo de variable de entorno"
```

Creamos la variable de entorno **cadena** con el contenido "Ejemplo de variable de entorno"

Eliminar el contenido de una variable de entorno:

```
Remove-Item Env:\nombre_variable
```

**Ejemplo:**

```
Remove-Item Env:\cadena
```

## **Lista de variables de entorno de Windows 10**

Variable de entorno	Valor
%SystemDrive%	Contiene la letra de la unidad en la que se ubica el sistema (por lo general C:).
%SystemRoot%	Contiene la ruta de acceso al directorio raíz del sistema.
%WINDIR%	Contiene la ruta de acceso al directorio del sistema (generalmente C:\WINDOWS o C:\WINNT).
%HOMEDRIVE%	Contiene la letra de la unidad en la que está ubicado el directorio actual del usuario.
%HOMEPATH%	Contiene la ruta de acceso completa al directorio actual del usuario.
%USERPROFILE%	Contiene la ubicación del perfil de usuario de la cuenta actual.
%APPDATA%	Muestra una ruta de acceso al directorio predeterminado que contiene los programas del usuario: C:\Users\USUARIO\AppData\Roaming
%ALLUSERSPROFILE%	Localización de los perfiles de los Usuarios: C:\ProgramData
%PROGRAMFILES%	Carpeta donde se instalan los programas: C:\Program Files
%PROGRAMFILES(X86)%	Se refiere a la carpeta C:\Program Files (x86) en los sistemas de 64 bits.
%PROGRAMDATA%	Ruta donde los programas almacenan los datos: C:\ProgramData
%TEMP%	Contiene la ruta de acceso al directorio temporal para las aplicaciones: C:\Users\USUARIO\AppData\Local\Temp
%LOCALAPPDATA%	Carpeta donde se guardan los archivos temporales: C:\Users\USUARIO\AppData\Local
%PUBLIC%	Carpeta donde se guardan los datos compartidos: C:\Users\USUARIO\AppData\Local

%COMMONPROGRAMFILES%	Carpeta donde guardan los programas los archivos comunes: C:\Program Files\Common Files
%COMMONPROGRAMFILES(x86)%	Ubicación del directorio de Archivos comunes utilizado por los programas de 32 bits. Solo se usa en versiones de 64 bits. C:\Program Files (x86)\Common Files

## Debes conocer

Como personalizar y añadir variables de entorno en Windows 10:

[Utilizar variables de entorno en Windows 10](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 3.2.- Proceso de arranque del sistema operativo Linux.

Una de las características más importantes de Linux es el método altamente configurable que se utiliza para el inicio del sistema operativo. El administrador es libre de configurar muchos aspectos del proceso de arranque, incluyendo qué programas se lanzarán en el momento del arranque. De forma parecida, la parada del sistema finaliza los procesos de forma organizada y configurable, aunque la personalización de este proceso casi nunca es necesaria. Entender el funcionamiento del proceso de arranque y parada no solo le permite personalizarlo, sino que también facilita resolver problemas relacionados con el inicio y el cierre del sistema.

**En la secuencia o proceso de arranque linux debemos de considerar los siguientes aspectos:**

Tras encender el ordenador, la BIOS (Sistema básico de Entrada/Salida) buscará el cargador de arranque, lo almacenará en memoria y le dará el control. Se inicia y ejecuta el cargador de arranque MBR.

El MBR se encuentra en el primer sector de la unidad de arranque. Normalmente en /dev/sda o /dev/hda. Este guarda toda la información sobre cualquier cargador de arranque instalado en el sistema. Si usamos GRUB u otro gestor de arranque como LILO, el MBR debe cargar y ejecutar el gestor GRUB.

En el GRUB se almacena toda la información sobre la imagen del sistema operativo que se debe cargar y ejecutar. Si tenemos más de un sistema operativo instalado, las entradas estarán en este archivo y podremos elegir el sistema operativo con el que arrancar el ordenador. Transcurridos unos segundos, si no elegimos ningún sistema operativo, se seleccionará el que tengamos definido por defecto en el archivo grub.cfg.

Al iniciar el Kernel, se montan el sistema de archivos raíz y se ejecuta /sbin/init.

**Init:** comprueba el archivo e indica el nivel de ejecución (este puede variar dependiendo de la carga del sistema). En Linux existen los siguientes niveles de ejecución (del 0 al 6):

### Niveles de ejecución

0	Halt o apagado de la máquina
1	Usuario exclusivo
2	Modo multiusuario sin acceso a NFS (sistema de archivos de red)
3	Modo multiusuario sin restricción
4	Reservado (no se utiliza salvo excepciones)
5	Modo multiusuario completo con inicio gráfico (similar al nivel 3)
6	Shutdown y reboot (Se apaga la máquina para reinicio)

Init verifica e identifica el nivel de inicio predeterminado del archivo y carga las herramientas y aplicaciones necesarias dependiendo del nivel de ejecución. Para ver qué nivel está siendo

ejecutado en nuestro sistema podemos usar el comando:

```
# runlevel
```

El comando anterior normalmente muestra lo siguiente:

```
N 5
```

Dependiendo del nivel de ejecución, el sistema iniciará las aplicaciones desde los siguientes directorios:

```
Run level 0 - /etc/rc0.d/  
Run level 1 - /etc/rc1.d/  
Run level 2 - /etc/rc2.d/  
Run level 3 - /etc/rc3.d/  
Run level 4 - /etc/rc4.d/  
Run level 5 - /etc/rc5.d/  
Run level 6 - /etc/rc6.d/
```

Actualmente se está utilizando más el GRUB que el LILO por ser mucho más flexible en su configuración, además, dispone de un interfaz más potente ya que dispone de un intérprete de comandos desde el que después de cambiar alguna de sus opciones no se necesita ejecutar nada (al contrario que el LILO). GRUB no puede usarse en el sistema de ficheros XFS o JFS.

## Para saber más

**Gestores de arranque:**

[Información sobre gestores de arranque en Linux](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 3.2.1.- Configuración del gestor de arranque Grub de Linux.

---

GRUB (Gran gestor de arranque unificado). Es un gestor de arranque que permite múltiples opciones de inicio del sistema. Carga todo lo necesario para que se inicie nuestro sistema operativo de forma correcta, como el Kernel y sus opciones predefinidas.

Podemos comprobar la versión de nuestro GRUB escribiendo el siguiente comando:

```
# grub-install --versión
```

Nos devolverá algo como esto:

```
grub-install (GRUB) 2.04-1ubuntu24
```

No necesita ser ejecutado después de cada actualización del núcleo. Sabe cómo leer los sistemas de archivos y como encontrar la ubicación del núcleo en el disco. Se instala en el MBR del primer disco, solo tenemos que ejecutar el siguiente comando:

```
grub-install /dev/sda
```

La configuración de GRUB2 está almacenada en /boot/grub/grub.cfg, pero este archivo (en Debian) es generado a partir de otros. Hay que tener cuidado de no modificarlo a mano ya que perderá dichas configuraciones locales la próxima vez que se ejecute update-grub (que puede ocurrir al actualizar algunos paquetes).

Las modificaciones más comunes del archivo /boot/grub/grub.cfg (agregar parámetros al núcleo o cambiar el tiempo que se mostrará el menú por ejemplo) se realizan a través de variables en /etc/default/grub. Podemos agregar elementos al menú creando un archivo /boot/grub/custom.cfg o modificando el archivo /etc/grub.d/50\_custom. Para configuraciones más complejas podemos modificar otros archivos en /etc/grub.d o crearlos; éstos scripts deben devolver porciones de configuración, posiblemente utilizando programas externos. Estos scripts son los que actualizarán la lista de núcleos a iniciar:

10\_linux tiene en cuenta los núcleos Linux instalados  
20\_linux\_xen tiene en cuenta sistemas virtuales Xen  
30\_os-prober listará otros sistemas operativos (Windows, OS X, Hurd).

### Configurar el GRUB desde la consola

El GRUB genera de forma automática una configuración que es válida para la mayoría de los usuarios. Pero esta configuración se puede modificar, para hacer esto tenemos que modificar su fichero de configuración que lo podemos encontrar en <em>/boot/grub/grub.cfg</em>, o en otras distribuciones esta en /etc/grub.conf.

En Ubuntu 20.04 LTS editamos el fichero de configuración de GRUB, <em>/etc/default/grub</em>, usando el comando:

```
# sudo nano /etc/default/grub
```

Siempre es recomendable hacer una copia de seguridad de ficheros que contenga una configuración importante, para evitar problemas.

Ubuntu - Captura de pantalla (Elaboración propia)

En este fichero tenemos muchas configuraciones, explicamos algunas de ellas a continuación:

**GRUB\_DEFAULT= 0:** Con la opción 0 hacemos que se seleccione por defecto la primera entrada (sistema), con la opción 1, la segunda... así sucesivamente. Si escribimos `saved`, hacemos que siempre se seleccione el último sistema al que se accedió.

**GRUB\_TIMEOUT=90:** Esta línea indica el tiempo de espera (en segundos) hasta iniciar el sistema que tenemos como default. Con un valor de -1, se desactiva la cuenta atrás y el valor será infinito.

**GRUB\_TIMEOUT\_STYLE=hidden:** con la opción oculta o cuenta regresiva, antes de mostrar el GRUB esperará a que expire el tiempo de espera establecido por `GRUB_TIMEOUT`. Si se presiona ESC durante ese tiempo, mostrará el menú y esperará la entrada.

**GRUB\_DISTRIBUTOR=`lsb\_release -i -s 2> /dev/null || echo Debian`:** determina el nombre de la entrada del menú.

**GRUB\_CMDLINE\_LINUX\_DEFAULT="quiet splash":** quiet sirve para agrupar las entradas iguales, mientras que splash sirve para que nos muestre la imagen de carga en vez de los mensajes del Kernel.

**GRUB\_CMDLINE\_LINUX="Opciones":** Las entradas en esta línea se agregan al final de la línea de comando 'linux' (línea "kernel" de GRUB legacy) para los modos normal y de recuperación. Se utiliza para pasar opciones al kernel.

**#GRUB\_GFXMODE=640x480** significa que no se tiene en cuenta y se aplicará el modo gráfico más adecuado a tu gráfica y monitor.

**#GRUB\_DISABLE\_LINUX\_RECOVERY="true"** Descomenta esta línea para que no aparezca la opción de recovery mode en el menú.

Veamos un ejemplo de configuración:

Supongamos que queremos iniciar con Ubuntu 20\_04 y este lo tenemos en la entrada (línea 3. En este caso lo configuraremos como:

```
GRUB_DEFAULT= 3
```

Si quisieramos que siempre se inicie nuestra ultima selección, escribiríamos lo siguiente:

```
GRUB_DEFAULT= saved
```

Si queremos cambiar el tiempo de espera hasta iniciar el sistema de forma que este sera 4 segundos, escribimos lo siguiente:

```
GRUB_TIMEOUT=4
```

Si escribimos

```
GRUB_TIMEOUT=-1
```

deshabilitamos la cuenta atrás y el valor será infinito.

Una vez hecho todos los cambios, tenemos que actualizar el GRUB para ello escribimos en el terminal:

```
$ sudo update-grub && sudo update-grub2
```

## Configurar el GRUB usando la herramienta GRUB Customizer

Es una herramienta gráfica muy intuitiva con la que podemos administrar el gestor de arranque de Linux. Permite configurar el sistema operativo predeterminado, agregar o eliminar entradas de inicio, establecer una imagen de fondo, etc.

Sus características principales son:

- Añadir, organizar, eliminar, y modificar el nombre de las entradas del menú.
- Mostrar o ocultar el menú al iniciar el sistema.
- Cambiar la entrada de arranque predeterminada.
- Editar los parámetros del Kernel.
- Modificar el tiempo de retraso hasta que carga la entrada predeterminada.
- Cambiar los colores del texto y la imagen de fondo.
- Reinstalar el cargador de arranque GRUB en el MBR.

Para instalarlo, escribimos en el terminal lo siguiente:

```
# apt-get install grub-customizer
```

Lo abrimos pinchando en *Mostrar aplicaciones*, escribimos

```
grub customizer
```

y pinchamos sobre el ícono con el mismo nombre. Nos pide la contraseña de administrador y se abre la aplicación.

En la imagen siguiente podéis ver la pantalla principal de Grub Customizer, donde podemos configurar las entradas.

Ubuntu - Captura de pantalla (Elaboración propia)

En la pantalla de *Configuración general* podemos configurar ciertos aspectos de la aplicación.

Ubuntu - Captura de pantalla (Elaboración propia)

En la pantalla de *Configuración de la apariencia* podemos elegir la resolución de la pantalla del GRUB, el color, o añadir una imagen de fondo.

Ubuntu - Captura de pantalla (Elaboración propia)

Desde la pantalla de configuración general, tenemos acceso a la configuración avanzada del GRUB, para ello, hacemos clic en el botón *Configuración avanzada*. Nos muestra la ventana de configuración avanzada, donde podemos configurar las diferentes opciones del GRUB.

Ubuntu - Captura de pantalla (Elaboración propia)

Una vez realizados los cambios, le damos a guardar. Reiniciamos la máquina y los cambios se habrán habilitado.

## Para saber más

**Configurar el menú de arranque GRUB2 en Debian, Ubuntu y derivados:**

[Configurar grub](#)

**Restaurar la configuración del GRUB2:**

[Restaurar la configuración de GRUB2](#)

**Modificar el menú de arranque con GRUB-CUSTOMIZER**

[Modificar el menú de arranque con GRUB-CUSTOMIZER](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 3.2.2.- Configuración de los Runlevels y el programa Init del sistema Linux.

**Los niveles de ejecución o runlevels** son los distintos modos de funcionamiento de init. Linux siempre se ejecuta en un determinado runlevel de ocho niveles posibles nombrados del 0 al 6 y el llamado S. La única diferencia entre un nivel y otro son los comandos o procesos que el administrador root configura en cada uno de ellos que indicarán las acciones a realizar al reiniciar o apagar el. Los procesos que se ejecutan en cada nivel dependerán de la distribución utilizada, aunque hay ciertos niveles estándar para todas ellas como son:

Runlevel 0	Apaga el sistema.
Runlevel S	Es el inicial en el que arranca cuando el kernel carga el init para que posteriormente se pasa a cargar el runlevel que está configurado por defecto.
Runlevel 1	Llamado modo monousuario que arranca el sistema sin iniciar la red ni ejecutar servicios (usado para reparar o depurar el sistema), es como el runlevel S pero pensado para salir de un runlevel y volver al estado inicial.
Runlevel 2	Para arrancar en modo texto
Runlevel 3	Multiusuario con red en modo texto
Runlevel 4	Libre para configurar por root
Runlevel 5	Multiusuario con red en modo gráfico con X-Window
Runlevel 6	Reinicia el sistema

Un sistema Linux no se arranca o detiene, sino que simplemente se cambia su nivel de ejecución. Algunos comandos que nos permiten gestionar los niveles de ejecución del sistema son los siguientes ejecutados con usuario root:

# init 0 # telinit 0	Para apagar el sistema
# init u	Para hacer que init se vuelva a ejecutar
# init q	Para que se vuelva a leer /etc/inittab y se apliquen los cambios necesarios
# kill -l	Podemos enviar una señal SIGHUP a init
# halt # shutdown -h	Una forma de forzar la salida de un runlevel es apagando o reiniciando el sistema, haciendo que init cierre los procesos del nivel que se está ejecutando y pase a ejecutar el runlevel 0

# reboot	Salir del nivel actual y pasar al nivel 6 que reinicia el sistema
# runlevel # who -r	Podemos saber en qué nivel de ejecución estamos ejecutando
/sbin/init nivel	Cambia al nivel especificado
/sbin/runlevel	Indica el nivel de ejecución previo y el actual
# init = /bin/sh	Desde el cargador grub puede expresarse el nivel de ejecución deseado pasándole como parámetro al kernel el nivel de ejecución

Ya sabemos que del Post pasamos al arranque del sistema operativo, cargando el kernel que creará todas las estructuras de memoria necesarias para la memoria virtual, posteriormente accede al disco duro y recupera el programa que ocupa el fichero /sbin/init, situándolo en memoria RAM, seguidamente lo ejecuta empezando por la primera página que ocupa, y arrancando el primer proceso del sistema. **El fichero que configura el init es <i>/etc/inittab</i>**, en el que se indica que debe ejecutar init y en que runlevel. El nivel de ejecución por defecto para el sistema está definido en el fichero /etc/inittab. Cada línea del fichero está formada por cuatro campos separados por dos puntos en los que se indica:

El identificador: es un código de una a cuatro letras que distingue las entradas de /etc/inittab.

El runlevel: es una lista de todos los runlevels seguidos en los que se ejecutará la entrada.

La acción: indica el comportamiento de init al ejecutar esta línea. Como pueden ser:

*wait*: para esperar la ejecución de otros comandos.

*once*: indica que sólo se ejecutará el comando una vez al entrar en el runlevel.

*respawn*: indica a init que el proceso debe ser reiniciado cada vez que termine. Usada para mantener activos servicios del sistema.

*initdefault* indica a init cual será el nivel de ejecución por defecto, no necesita de ningún comando. Por ejemplo para un runlevel por defecto de 3: <span>id:3:initdefault:</span>

Existen tres acciones que permiten indicar que hacer durante el arranque del sistema e ignorar los runlevels que se especifican en la línea y son *<i>sysinit</i>*, *<i>boot</i>* y *bootwait* (en la que init espera a que el proceso termine antes de seguir procesando el fichero).

Con la combinación de teclas Control+Alt+Suprimir se reinicia el sistema, se puede realizar con la línea de comandos:

```
Kill -2 l
```

El proceso: es el comando o programa que se ejecutará.

### **3.2.3.- Configuración de los ficheros rc del arranque de Linux.**

---

Los ficheros rc son aquellos que contienen instrucciones para el inicio de un programa. En algún momento init lanza los procesos definidos por el sistema rc.

En las distribuciones Linux existen dos estilos sobre el uso del sistema rc:

- 1.- BSE
- 2.- SysV.

Linux usa SysV.

init ejecutara un programa script denominado rc que a su vez ejecuta los script de inicio de los programas que hacen funcionar el sistema. La ubicación de los rc encuentra en <i>/etc/init.d/rc0 </i>y en /sbin/rc, dependerá de la distribución.

De forma general, existirá un directorio <i>/etc/rc<x>.d/</i> , por cada nivel de ejecución definido por el sistema, donde se encuentran los servicios que deberán ser lanzados y parados en ese nivel de ejecución. Cuando entra en un determinado nivel de ejecución realiza las siguientes acciones:

- 1.- Ejecuta, por orden de nombre, todos los scripts que comienzan por **K** en el directorio correspondiente al nivel, utilizando como argumento para dicho script la opción **stop**.
- 2.- Ejecuta, por orden de nombre, todos los scripts que comienzan por **S** en el directorio correspondiente al nivel, utilizando como argumento para dicho script la opción **start**.

A título de ejemplo, a continuación se muestra un listado del directorio que corresponde al nivel multiusuario con red (<i>/etc/rc5.d</i>).

Ubuntu - Captura de pantalla (Elaboración propia)

Como se puede apreciar, ninguno de los scripts que inician y apagan los servicios están localizados en el directorio /etc/rc5.d/. Todos los ficheros en /etc/rc5.d/ son enlaces simbólicos apuntando a los scripts localizados en el directorio /etc/init.d/. Los enlaces simbólicos se usan en cada uno de los directorios rc de manera que los niveles de ejecución puedan ser reconfigurados al crear, modificar y eliminar los enlaces simbólicos sin que afecte a los scripts actuales a los que se refiere.

El nombre de cada enlace simbólico empieza con **K** o **S**. Como ya habíamos comentado, los scripts que empiezan por **K** son procesos candidatos a ser parados en ese nivel de ejecución, mientras que aquellos que empiezan por **S** son procesos candidatos a ser iniciados. El nombre de los enlaces de estos directorios puede ser poco intuitivos al principio, pero veamos su significado, la sintaxis es:

[K | S] + nn + [string]

Donde:

La primera letra del nombre puede ser una K o una S, seguidas de un número de dos dígitos, del 01 al 99 y por último una cadena de texto.

La K significa que el servicio será detenido al entrar al runlevel (Kill), la S para iniciararlo (Start).

El número indica la prioridad del servicio dentro del runlevel, por ejemplo, S02apache y S01php iniciará primero php y luego apache. Si dos servicios tienen el mismo orden de prioridad numérico, se procede en orden alfabético.

## Agregar un servicio a un runlevel

Para agregar un servicio a un runlevel deberemos usar el comando

```
update-rc.d
```

Por ejemplo, si quieras que nginx o Apache se ejecuten en cada inicio del sistema, basta con agregarlos a los runlevel 2-5, correspondientes al modo multiusuario:

```
# update-rc.d nginx start 90 2 3 4 5 . stop 01 0 1 6
```

Donde:

90 es el número de prioridad para el inicio (**S90nginx**) aplicado a los runlevel 2-5.

01 para la prioridad de detención (**K01nginx**) en los runlevel 0 1 y 6.

## Para saber más

Podemos obtener más información sobre el comando rc para la distribución de Ubuntu en:

[Manual de Ubuntu sobre update-rc](#)

Entendiendo los runlevel en Debian:

[Información sobre los runlevels en Debian](#)

Como va update-rc-d

[Información sobre update-rc.d](#)

## 3.2.4.- Variables de entorno generadas en el proceso de arranque de Linux.

Una **variable del entorno** es un valor dinámico cargado en la memoria, que puede ser utilizado por varios procesos que funcionan simultáneamente. En la mayoría de los sistemas operativos, la ubicación de algunas bibliotecas o de los archivos ejecutables del sistema más importantes puede variar según la instalación. Podemos considerar su estudio en este momento ya que cuando se genera el arranque de los sistemas a la vez se van cagando en memoria una serie de valores en las llamadas variables del sistema o de entorno

Por eso es posible, para un programa dado, remitirse a una ubicación basada en las variables del entorno que definen estos datos.

En sistemas Linux las variables del entorno están precedidas por el carácter "\$" y se ubican entre corchetes, aunque éstos no son obligatorios. Debido a la variedad en los sistemas Linux (particularmente a las diferentes distribuciones Linux), para que se muestre el valor de una variable del entorno sólo se debe escribir uno de los siguientes comandos:

```
echo $VARIABLE
```

ó

```
echo ${VARIABLE}
```

Ejemplo:

```
echo $SHELL
```

Resultado:

```
/bin/bash
```

A continuación se proporciona una lista incompleta de las principales variables del entorno en un sistema Linux:

Variable de entorno	Descripción
\$ARCH	Contiene la descripción de la arquitectura del equipo.
\$DISPLAY	Contiene la identificación de la terminal de visualización que se utilizará en el administrador de ventanas (x11).
\$HOME	Muestra la ruta de acceso al directorio actual del usuario.
\$HOST	Muestra el nombre del equipo.

\$LANG	Muestra el código del idioma predeterminado.
\$PATH	Muestra una lista de rutas de acceso a los directorios que contienen archivos ejecutables, separadas por punto y coma.
\$PRINTER	Contiene el nombre de la impresora predeterminada.
\$SHELL	Indica la ruta del intérprete de comandos utilizado.
\$USER	Muestra la identificación del usuario actual.

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 4.- Registro de sistema.

### Caso práctico

**Vindio** le comenta a **Noiba** que en muchas ocasiones, para solucionar problemas referentes al funcionamiento del sistema, los administradores realizan consultas por Internet buscando la ayuda de otros usuarios que nos permita encontrar una solución.

Puede que la solución se encuentre en la configuración de la base de datos de registro del sistema, —le comenta **Noiba** a **Vindio**.

Efectivamente **Noiba**, esto es así, —responde **Vindio**.

En ese caso, deberemos documentarnos sobre su manejo para encontrar posibles errores ocasionados, como por ejemplo, la infección del sistema por un virus.

[Alain Bachellier \(CC BY-NC-SA\)](#)

### Registro de sistemas operativos Windows

El registro de Windows permite guardar datos de configuración del sistema operativo para hacer referencia del mismo cuando se realizan tareas como perfil de usuario, de hardware, de software instalado en máquina, puertos del sistema, etc. Es decir, el **registro del sistema**, o registro de Windows, es una base de datos que almacena las configuraciones y opciones del sistema operativo Microsoft Windows en sus versiones de 32 bits, 64 bits y Windows Mobile; contiene información y configuraciones de todo el hardware, software, usuarios, y preferencias del ordenador. Si un usuario hace cambios en las configuraciones del "Panel de control", en las asociaciones de ficheros, en las políticas del sistema o en el software instalado, los cambios se reflejan y almacenan en el registro.

El registro reemplaza los archivos de inicialización y configuración legados de Windows 3.x y MS-DOS (.ini), autoexec.bat y config.sys. Los datos de registro se almacenan en archivos binarios, en las carpetas:

\%SYSTEMROOT%\system32\config  
%USERPROFILE%\NTUSER.DAT donde %USERPROFILE% es  
c:\WINDOWS\nombreusuario. Cada usuario dispone de un archivo NTUSER.DAT que almacena datos sobre su configuración de registro personal.

En Windows las variables de sistema aparecen identificadas en los símbolos %. Podemos visualizar sus valores desde una consola de entrada de comandos con el comando

SET

Por ejemplo:

%SYSTEMROOT% indica el path del directorio raíz del sistema y %USERPROFILE% el path del usuario).

Para entrar en el editor de registro de Windows:

Escribimos regedit desde Inicio-Buscar y seleccionamos el ícono *Editor de registro*. También pulsando la combinación de teclas **Windows+R** y escribimos regedit.

Microsoft - Captura de pantalla (Elaboración propia)

El registro de Windows se muestra en estructura jerárquica de árbol de claves. Tiene las siguientes ramas:

HKEY\_CLASSES\_ROOT  
HKEY\_CURRENT\_USER  
HKEY\_LOCAL\_MACHINE  
HKEY\_USERS  
HKEY\_CURRENT\_CONFIG.

Estas ramas principales se subdividen en claves que pueden tener subclaves igual que las carpetas tienen subcarpetas. En la imagen inferior 1 es una rama y 2 es una clave o subclave.

Microsoft - Captura de pantalla (Elaboración propia)

Cada registro de clave contiene subcarpetas o subclaves que aparecen en el lado izquierdo de la ventana. Cada uno de ellos puede tener uno o varios valores (1 en la imagen inferior). Cada

valor puede tener un tipo de datos (2 en la imagen) y el dato que contiene (3 en la imagen).

Microsoft - Captura de pantalla (Elaboración propia)

Los siguientes archivos del Registro se encuentran en %SystemRoot%\System32\Config\ y las claves se vinculan con los siguientes archivos auxiliares:

HKEY\_LOCAL\_MACHINE\SAM: con Sam, Sam.log, Sam.sav  
HKEY\_LOCAL\_MACHINE\SECURITY: con <i>Security</i>, <i>Security.log</i>, <i>Security.sav</i>  
HKEY\_LOCAL\_MACHINE\SOFTWARE: con Software, Software.log, Software.sav  
HKEY\_LOCAL\_MACHINE\SYSTEM: con <i>System</i>, <i>System.alt</i>, <i>System.log</i>, <i>System.sav</i>  
HKEY\_USERS\DEFAULT: con Default, Default.log, Default.sav  
HKEY\_CURRENT\_CONFIG: con System, System.alt, System.log, System.sav, Ntuser.dat, Ntuser.dat.log

## Para saber más

**Información del registro de Windows:**

[Información del registro de Windows \(Microsoft\)](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 4.1.- Operatividad con el registro del sistema operativo Windows.

Existen varias utilidades para optimizar el registro de Windows como TuneUp Utilities, CCleaner, etc. Además de optimizar el registro la mayoría de estos programas reparan errores provocados por la instalación o desinstalación de programas, valores guardados, etc. También se puede operar con el registro del sistema con las directivas de grupo gpedit.msc, con directivas del sistema <i>reg.exe</i> y ficheros con extensión reg. Para modificar el registro de Windows sabemos que las *HKEY* son de dos tipos:

Las que se crean cada vez que se realizar una operación de administración como instalar o desinstalar aplicaciones, crear usuarios, cambiar configuración de programas de arranque, etc.

Las que se crean cada vez que se inicia el sistema.

Para operar con claves nos situamos sobre una de ellas (pantalla lista variables lado izquierdo) y hacemos doble clic con el ratón:

Microsoft - Captura de pantalla (Elaboración propia)

Las posibles operaciones con claves son:

Para crear una subclave seleccionamos del menú *Edición-Nuevo-Clave* o *Edición-Nuevo-Tipo de valor*.

Si deseamos modificar se selecciona la clave *Edición-Cambiar Nombre*.

Para borrar clave o valor seleccionamos la clave y pulsamos en *Edición -Eliminar*.

Podemos buscar claves y valores desde menú *Edición-Buscar* en la ventana que aparece escribimos el valor a buscar.

Editar el registro se desaconseja en general por la poca trazabilidad de las modificaciones, siendo recomendable realizar una copia de seguridad antes de la modificación. Para realizar una copia vamos *Archivo-Exportar* damos un nombre para archivar la copia podemos hacer una copia de todo el registro (opción *Todo*) o de una parte seleccionada (opción *Rama seleccionada*). La copia es un fichero con el nombre dado y con la extensión *reg*(compuesta por texto plano y se puede editar desde un editor de textos). Para restaurar la copia vamos al fichero y damos doble clic sobre él, la copia se instala y se fusiona en el registro actual, pero es más recomendable realizar desde la aplicación de regedit de la siguiente manera, para restaurar todo o parte del registro desde el menú en *Archivo-Importar* se introduce el nombre del archivo que contiene la copia y se presiona en el botón Abrir, que se sobrescribirá el registro y habrá que reiniciar el sistema.

Para poder configurar el registro de Windows se tiene que haber iniciado sesión como administrador, de esta manera el usuario administrador puede dar permisos para cada clave del sistema, para ello se selecciona una clave se hace clic con el botón derecho del ratón y se

pulsa en la opción *Permisos*, podemos dar permiso de *lectura* y de *control total* (el usuario puede crear, modificar o eliminar valores), también podemos administrar permisos especiales para el usuario o grupo de usuarios pulsando en el botón de *Opciones avanzadas*.

## Modificar el registro desde la línea de comandos

Con REG se pueden administrar todos los parámetros del Registro desde la Línea de comandos. Se pueden agregar claves, modificar claves, valores, exportar ramas, etc.

El comando REG se compone de varios subcomandos, cada uno para un uso completamente diferente, de ellos REG QUERY, REG ADD, REG DELETE y REG COPY se pueden utilizar para hacer modificaciones en equipos remotos en la red. Solo es necesario agregar \\NombreEquipo suponiendo que el servicio Registro Remoto (RemoteRegistry) este ejecutándose.

Comandos con REG	Uso
QUERY	Busca el valor de una clave.
ADD	Agrega una clave o valor
DELETE	Elimina una clave o valor
COPY	Copia claves y valores desde y hacia otro equipo o en el equipo local
SAVE	Guarda claves del registro en archivos .HIV locales (archivo de subárbol)
RESTORES	Restaura archivo de subárbol que han sido guardados
LOAD, REG UNLOAD	Transfieren y mueven claves dentro del registro
COMPARE	Compara claves y valores dentro del registro
EXPORT	Exporta claves, subclaves y valores a un archivo REG
IMPORT	Importa los datos

exportados  
previamente desde un  
archivo REG

Abriendo el símbolo del sistema (CMD), podemos buscar ayuda del comando reg, usando el siguiente comando:

```
reg /?
```

Si queremos obtener ayuda de cada uno de los comando de reg podemos escribir el siguiente comando:

```
REG COMANDO /?
```

Ejemplo:

```
REG ADD /?
```

Nos muestra ayuda para agregar una clave o valor con REG ADD.

Algunos ejemplos para trabajar con el registro usando el comando reg:

Ejemplo: **crear la clave** *nombre* dentro la subclave HKLM\Software con el tipo de datos REG\_SZ y con el dato *Mi nombre es Juan*

Para ello abrir el interprete de comando en modo administrador y escribimos lo siguiente:

Microsoft - Captura de pantalla (Elaboración propia)

Ejemplo: **eliminar la clave** creada anteriormente (*nombre*).

En el interprete de comandos escribimos:

Microsoft - Captura de pantalla (Elaboración propia)

## Para saber más

Trabajar con el registro en Windows:

[Guía básica para trabajar con el registro de Windows.](#)

Cómo agregar, modificar o eliminar subclaves y valores del registro mediante un archivo .REG:

[Página de Microsoft: Agregar, modificar y eliminar subclaves del registro](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 4.2.- Registro de sistema operativo Linux.

La alternativa de Linux con respecto a manejar un registro del sistema es la de **no disponer de una base de datos de registro** de sistema que permita realizar cambios referentes a procesos de administración y configuración ya que se corre el riesgo de perder la seguridad del sistema, en muchos casos Windows ha recibido ataques al sistema mediante el acceso a su registro haciendo que el sistema caiga y sea irrecuperable el fallo.

En conclusión, en **Linux no hay registro**, las configuraciones del sistema se editan de otro modo (desde una terminal de comandos, como algo similar al editor de registro); de manera que **todo en Linux es un fichero**, incluso los dispositivos, cada concepto está asociado a un fichero que el administrador root del sistema puede editar en cualquier momento y modificar el valor de sus directivas (variables y parámetros) de configuración. La mayoría de los servicios se pueden adaptar a las necesidades de uso en el servidor. A la hora de instalar una aplicación podemos obtener el código fuente de los programas y cambiar su código para que posteriormente se compile y cree el fichero ejecutable que se puede ejecutar en el directorio poniendo `./nombre_programa`.

Casi todas las procesos de configuración del sistema se pueden realizar desde modo consola o modo entrada de línea de comandos. De manera que se puede ejecutar una aplicación o **Script o demonio** ( proceso informático que se ejecuta en segundo plano (símbolo & al final del comando) este tipo de programas se ejecutan de forma continua hasta que se paren, normalmente comando `stop` o se inicien con el comando `start`) indicando algún parámetro de entrada de ejecución (podemos consultar la ayuda mediante el comando `man`).

En otra unidad temática veremos que en un sistema GNU/Linux todo queda registrado, median el demonio `syslogd`. Toda información relevante (mensajes de arranque, accesos, errores de conexión en los servidores, ...) se guarda en unos archivos. Son los *logs* o ficheros de sucesos y eventos del sistema (se encuentran en el directorio `/var/log/`).

Para acceder al fichero `/var/log` y ver los fichero logs hacemos lo siguiente:

```
# cd /var/log  
# ls *.log
```

Ubuntu - Captura de pantalla (Elaboración propia)

Podemos ver el contenido de los ficheros **.log** usando cualquier editor de texto, por ejemplo:

```
# cat /var/log/auth.log
```

En modo gráfico podemos ver los logs usando la aplicación *registros*. Para abrirla, pinchamos en mostrar aplicaciones, escribimos registros, pinchamos sobre el ícono registros. Esta aplicación nos muestra de forma amigable la información correspondiente a los logs del sistema con la posibilidad de separarlos por tipo y agrupados por registro.

El archivo de configuración general de `<em style="font-size: 1em;">syslog</em>` es `<i style="font-size: 1em;">/etc/syslog.conf</i>`. También puede aparecer `rsyslog` y el archivo de configuración es `/etc/rsyslog.conf`. La cantidad de carpetas y de archivos que encontraremos en `<i style="font-size: 1em;">/var/log/</i>` dependerá en gran medida de los servicios que haya en nuestra máquina. Así, por ejemplo, los

mensajes referentes a nuestro servidor web los encontramos en <i style="font-size: 1em;">/var/log/httpd</i>, y los de nuestro servidor de correo en <i style="font-size: 1em;">/var/log/mail/</i>.

Ubuntu - Captura de pantalla (Elaboración propia)

## Debes conocer

Como revisar los registros logs del sistema en Linux:

[Revisar los registros logs del sistema Linux](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 5.- Actualización y mantenimiento de controladores de dispositivos.

### Caso práctico

**Vindio, Noiba, Naroba y Jana**, han utilizado el ordenador "caja de herramientas", para instalar todos los sistemas operativos y aplicaciones que utilizarán los ordenadores de la empresa.

Seguramente, los ordenadores dispondrán de diferentes dispositivos externos e internos conectados. Comenta **Jana**.

Efectivamente confirma **Laro**. ¿Sabéis como se gestiona la administración de los programas que permiten el correcto funcionamiento de todos los componentes físicos en un ordenador?

[Alain Bachellier \(CC BY-NC-SA\)](#)

Sería interesante investigarla. Esto nos permitirá conocer qué utilidades y servicios se utilizan en la administración de los controladores de los dispositivos, comenta **Jana**.

El **controlador de un dispositivo** o driver es un programa informático que permite al sistema operativo interactuar con un dispositivo, haciendo una abstracción del hardware y proporcionando una interfaz para facilitar su configuración. Existen tantos tipos de controladores como tipos de dispositivos, y es común encontrar más de un controlador posible para el mismo dispositivo, adaptado a cada distribución del sistema operativo. Por ejemplo, aparte de los oficiales (normalmente disponibles en la página web del fabricante), se pueden encontrar también los proporcionados por el sistema operativo, o también versiones no oficiales hechas por terceros.

Los controladores de dispositivo se adjuntan al núcleo del sistema operativo. Los controladores de dispositivo pueden ser de dos tipos: orientados a caracteres (tales como los dispositivos NUL, AUX, PRN, del sistema) o bien orientados a bloques, constituyendo las conocidas unidades de disco. La diferencia fundamental entre ambos tipos de controladores es que los primeros reciben o envían la información carácter a carácter; en cambio, los controladores de dispositivo de bloques procesan, como su propio nombre indica, bloques de cierta longitud en bytes (sectores).

Elaboración propia (Dominio público)

Los controladores de dispositivo han sido tradicionalmente programas binarios puros, similares a los COM aunque ensamblados con un ORG 0, a los que se les colocaba una extensión SYS.

Sin embargo, no hay razón para que ello sea así ya que un controlador de dispositivo puede estar incluido dentro de un programa EXE, con la condición de que el código del controlador sea el primer segmento de dicho programa.

La firma de controladores gestionada por el sistema hace que los controladores estén autorizados para su instalación en el sistema operativo y proporcionan las ventajas siguientes:

**Más Seguridad en el sistema.** Puesto que los usuarios estándar no pueden instalar controladores de dispositivos que no estén firmados o que estén firmados por un editor que no es de confianza. Mediante el uso de directivas de grupo, un administrador puede proporcionar a todos los equipos cliente de una organización los certificados de los editores que se consideren de confianza, permitiendo la instalación de los controladores sin intervención del usuario, para comprobar que se trata de una firma digital de confianza.

**Más fiabilidad en el funcionamiento del dispositivo.** Los usuarios sólo podrán instalar los dispositivos que hayan sido probados y admitidos por la organización.

**Funcionamiento automático.** Cuando el usuario conecta el dispositivo al equipo el dispositivo funciona automáticamente sin acción alguna por parte del usuario.

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 5.1.- Gestión de controladores de dispositivos en el sistema operativo Windows.

---

Cuando se instala un componente de hardware en el sistema, el propio Windows detecta la conexión gracias al sistema **Plug-and-play** (conocida también por su abreviatura **PnP**, es la tecnología para añadir hardware que requiere que un dispositivo se identifique a sí mismo al conectarse al ordenador) e intenta buscar un controlador compatible con el componente dentro de su base de datos de controladores. En el caso de no encontrar uno, solicita la instalación del mismo vía Internet o el controlador alojado en una unidad de almacenamiento (como el DVD o CD del fabricante).

Cuando instalamos un sistema operativo, es aconsejable **comprobar si están instalados todos los dispositivos** correctamente, para ello debemos ir a: *Inicio-Panel de control-Hardware y sonido*. Hacemos clic encima de *Hardware y sonido*.

En la ventana que aparece seleccionamos la opción *Administrador de dispositivos* y aparece la lista de dispositivos conectados en el caso de que aparezca un dispositivo con problemas aparece un símbolo de admiración, interrogación o señal de peligro, al lado de la descripción del componente.

Microsoft (Elaboración propia)

Para **instalar actualizaciones** opcionales, debe revisarlas y luego seleccionarlas de una lista de actualizaciones que Windows encuentra para el equipo. Las actualizaciones opcionales no se instalan automáticamente. Esto se realiza desde Windows Update:

- 1.- *Inicio-cuadro de búsqueda*, escribir *Update* en la lista de resultados y hacer clic en Windows Update.
- 2.- Hacer clic en *Buscar actualizaciones*. Si ves un mensaje que indica que hay actualización es opcionales disponibles, o que te insta a revisar actualizaciones opcionales, haga clic en el mensaje para ver y seleccionar las actualizaciones opcionales que se instalarán. Haga clic en las actualizaciones opcionales de la lista para ver más información acerca de ellas. Active las casillas correspondientes a las actualizaciones opcionales que desea instalar y haga clic en *Aceptar*.

Windows incorpora un sistema de **firma digital para los controladores**. Para que un controlador sea aceptado por Windows sin problemas, éste debe poseer un certificado digital que lo hace compatible con Microsoft. Supuestamente, esto garantiza que no tendremos problemas con el software. Windows, por defecto, bloquea los controladores no firmados, lo que impide la instalación de estos en muchos casos, haciendo imposible el funcionamiento del dispositivo asociado.

Por lo tanto, es necesario **desactivar** temporalmente la **verificación de firmas de controladores** de la siguiente manera:

- 1.- Presionamos [Tecla Windows]+[X]. Seleccionamos la opción *Símbolo del sistema (administrador)*.
- 2.- Escribimos los siguientes comandos:

```
bcdedit -set loadoptions DISABLE_INTEGRITY_CHECKS  
bcdedit -set TESTSIGNING ON
```

3.- Instala los controladores que necesites.

Para activar la seguridad, para comprobar firmas de los controladores y mantener seguro tu equipo. ejecutamos los siguientes comandos:

```
bcDEDIT -set loadoptions ENABLE_INTEGRITY_CHECKS
```

```
bcDEDIT -set TESTSIGNING OFF
```

## Para saber más

Detectar un dispositivo desconocido y encontrar sus controladores:

[Cómo detectar un dispositivo desconocido y encontrar sus controladores](#)

Solución de problemas al instalar y actualizar controladores:

[Cómo solucionar problemas al instalar y actualizar controladores](#)

El administrador de dispositivos:

[Utilización del administrador de dispositivos](#)

Los mejores programas para buscar y actualizar drivers en Windows:

[Los principales programas para buscar y actualizar drivers en Windows](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

## 5.2.- Gestión de controladores de dispositivos en el Sistema operativo Linux.

Linux trata a los drivers o controladores prácticamente como una aplicación más, de hecho podemos instalarla desde el mismo repositorio que instalamos cualquier otro software, sin necesitar ningún requisito adicional. Por ello, cuando queramos actualizar nuestros controladores (para la mayoría de ellos) solo tendremos que pinchar en mostrar aplicaciones y escribir Actualizaciones para que los controladores incluyan dentro de esta búsqueda y se instalen si es necesario. Cuando instalamos las actualizaciones al mismo tiempo que se instalan las aplicaciones o actualizaciones del sistema, se actualizan nuestros controladores.

Por temas legales, nuestro sistema Ubuntu no puede llevar integrados directamente drivers propietarios de los fabricantes de hardware y ya que no tienen la misma licencia que Ubuntu. Sin embargo, nada nos prohíbe acudir a las webs de los fabricantes de hardware y obtener los controladores, para poder instalarlos y configurarlos nosotros.

Para comprobar si estamos utilizando nuestra tarjeta gráfica al máximo rendimiento con los controladores adecuados, o si existe un controlador propietario para alguno de nuestros elementos hardware de nuestra computadora, pinchamos en *Mostrar aplicaciones* y escribimos *controladores*. Nos aparece un ícono con el nombre *Más controladores*, pinchamos sobre él, y se nos abre la ventana *Más controladores* de la aplicación Software y actualizaciones.

Ubuntu - Captura de pantalla (Elaboración propia)

El contenido de la pestaña *Más controladores* cambiará con cada ordenador, pero en general siempre veremos un controlador genérico y otros drivers propietarios, algunos testeados y otros no, con su número de versión. Es mejor elegir el driver testeado con un número de versión más alto.

Es conveniente tener actualizado el repositorio de búsqueda de paquetes de software, tanto para la actualización del sistema como para la actualización de aplicaciones y programas instalados en los que podemos incluir controladores y aplicaciones que operan con los elementos de hardware. Para ello desde el entorno gráfico, pinchamos en *Mostrar aplicaciones* y escribimos *Synaptic*. Posteriormente entramos en la configuración de línea de registro de repositorios desde la pestaña Software de terceros, y podemos añadir repositorios (lugares de servidores de Internet descarga de software).

Para gestionar la actualización de programas y ficheros controladores de componentes, debemos actualizar el fichero de donde se encuentran los repositorios o las direcciones de búsqueda de servidores que ofrecen la posibilidad de descargar de aplicaciones y programas. El fichero que contiene dichos datos es *sources.list*, para gestionar su actualización deberemos realizar los siguientes pasos:

- 1.- Abrimos un terminal.
- 2.- Entramos como usuarios root:

```
$ su root
```

3.- Realizamos una copia del fichero, por seguridad (para no perder el original). Siempre que realizamos modificaciones en ficheros fuente que permiten configurar el sistema es conveniente realizar una copia de seguridad del fichero original. Usamos la orden:

```
# cp /etc/apt/sources.list /etc/apt/sources.list_copia
```

4.- Editamos el fichero sources.list para realizar modificaciones:

```
# gedit /etc/apt/sources.list
```

5.- Buscamos las líneas que empiezan por #deb... y quitamos el # (para descomentar la línea). Para mejorar la transferencia es mejor elegir un mirror de tu país.

6.- Guardar el fichero editado.

7.- Actualizar la lista de paquetes, con la orden:

```
# apt-get update ó # aptitude update
```

Seguidamente, después de actualizar y añadir repositorios para que se pueda descargar programas y controladores con la orden <span lang="en">aptitude</span>. Podemos instalar drivers o controladores desde la línea de comandos realizando los siguientes pasos:

1.- Abrimos un terminal.

2.- Ejecutamos el comando:

```
$ sudo aptitude install nombre_fichero_controlador
```

En el caso que descarguemos el fichero en código fuente *comprimido* y *empaquetado*. Para su instalación debemos realizar los siguientes pasos:

1.- Despues de obtener el fichero en un directorio, accedemos al directorio donde hemos descargado el fichero, y como usuario root, ejecutamos la orden para desempaquetar y descomprimir:

```
# tar zxvf fichero_comprimido.tar.gz
```

2.- Entrar al directorio que se crea con el mismo nombre que la aplicación (se recomienda leer el archivo readme).

3.- Ejecutar las siguientes ordenes:

```
# ./config
```

```
# make
```

```
# make install
```

#### 4.- Reiniciar el sistema con la orden:

```
# reboot
```

Para comprobar el hardware que tenemos instalado en nuestro sistema, podemos recopilar esta información con aplicaciones gráficas o por medio de comandos.

### Comprobar el hardware del sistema en modo gráfico

Podemos utilizar aplicaciones como hardinfo ó **Lshw-gtk**.

**Hardinfo:** muestra un detalle del hardware utilizado pero, a diferencia de lshw, muestra también algunos datos interesantes sobre el sistema operativo: la resolución de la pantalla y otra información relacionada, la versión del kernel, el nombre de la computadora y del usuario actual, el entorno de escritorio, el tiempo de ejecución, los módulos del kernel activos, los idiomas disponibles, información sobre el sistema de archivos, etc. Permite también ejecutar varias pruebas de rendimiento (benchmarks). No está instalada por defecto en Linux.

Instalación en Debian/Ubuntu:

```
sudo apt-get install hardinfo
```

Instalación en Fedora:

```
sudo yum install hardinfo
```

Abrir **hardinfo**, pinchamos en *mostrar aplicaciones*, escribimos hardinfo y pinchamos sobre su ícono.

**Lshw-gtk:** es la interfaz gráfica de lshw, una herramienta de línea de comandos que sirve para mostrar información sobre el hardware en uso.

Ubuntu - Captura de pantalla (Elaboración propia)

Instalación en Debian/Ubuntu:

```
sudo apt-get install lshw-gtk
```

Instalación en Fedora:

```
sudo yum install lshw-gui
```

Para abrir lshw, abrimos un terminal y escribimos:

```
lshw-gtk, o bien como administrador, sudo lshw-gtk
```

# Comprobar el hardware del sistema en modo comando

## Listado general del hardware de un equipo

Podemos usar el comando:

```
$ sudo lshw
```

Con este comando obtenemos un resumen general de todo el hardware detectado. El listado es muy grande y detallado, por lo que es conveniente guardarlo en un fichero.

## Información del procesador

En el fichero /proc/cpuinfo nos da información detallada de nuestra CPU:

```
$ cat /proc/cpuinfo
```

ó

```
$ lscpu
```

## Información de la memoria

Podemos usar el comando general sobre el hardware del sistema *lshw* y de esta extraer la información de la memoria.

```
$ sudo lshw
```

## Información de los discos duros

Con fdisk nos muestra información de los dispositivos de almacenamiento detectado en nuestro equipo.

```
$ sudo fdisk -l
```

## Para saber más

Para consultar un ejemplo de instalación de un controlador puedes acceder a:

[Ejemplo de instalar un controlador en Linux](#)

Reconocer hardware en Ubuntu:

[Reconocer hardware en Ubuntu](#)

## Debes conocer

Instalar y usar Hardinfo:

[Instalar y usar Hardinfo](#)

Instalar y usar <abbr title="Hardware Lister"><span title="Hardware Lister" lang="en">Lshw</span></abbr>:

[Ver información del hardware usando Lshw](#)

Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](#)

# Condiciones y términos de uso de los materiales

Materiales desarrollados inicialmente por el Ministerio de Educación, Cultura y Deporte y actualizados por el profesorado de la Junta de Andalucía bajo licencia Creative Commons BY-NC-SA.

Antes de cualquier uso leer detenidamente el siguiente [Aviso legal](#)

## Historial de actualizaciones

**Versión: 01.00.01**

**Fecha de actualización: 29/09/21**

Actualización de materiales y correcciones menores.

**Versión: 01.00.00**

**Fecha de actualización: 23/07/20**

Versión inicial de los materiales.

