

## **Ud. 4: Administración del sistema operativo.**

Primeros pasos con el sistema operativo	3
1.1. Interacción del usuario con el sistema operativo	3
1.2. Mediante la interfaz gráfica	3
1.3. Intérprete de comandos	4
1.4. Editores de texto	6
1.5. Redirección	7
1.6. Histórico de comandos y expansión de teclado	7
2. Archivos y directorios	8
2.1. Directorios o carpetas: introducción	8
2.2. Características de los directorios	8
2.3. Crear carpetas o directorios	11
2.4. Eliminar directorio o carpetas	12
2.5. Ver el contenido de las carpetas y desplazarse por ellos	12
2.6. Visualizar la estructura de directorios	13
3. Gestión de usuarios	24
3.1. Usuarios y grupos en Windows	24
3.1.1. Perfiles de usuario	26
3.1.2. Opciones de seguridad	27
3.1.3. Directiva de contraseña	27
3.1.4. Directiva de bloqueo de cuentas	28
3.2. Usuarios en Linux	28
3.2.1. Usuarios en modo gráfico	29
3.2.2. Alta de usuarios con comandos	29
3.2.3. Gestión de contraseñas	30
3.2.4. Modificación y baja de usuarios locales	30
3.2.5. Archivos de configuración	31
3.2.6. Comandos para la administración de usuarios	33
3.2.7. Grupos de usuarios	33
4. Monitorización de procesos y servicios	35
4.1. Procesos en primer y segundo plano	35
4.2. Gestión de procesos en Windows	35
4.3. Comandos de Linux para la gestión de procesos	39
4.4. Iniciar y detener servicios en Linux	40
4.5. Niveles de ejecución o runlevel en Ubuntu	43
5. Conexión de la impresora	46
5.1. Conexión física	46
5.2. Conexión lógica	46
5.3. Conexión cliente servidor	50
5.4. Colas de impresión	51
5.5. Servicios de impresión	53
6. Instalación de software	54
6.1. Instalación en Windows	54
6.2. Instalación de software en Linux	56
6.2.1. Desde Internet	56
6.2.2. Centro de software de Ubuntu	58
6.2.3. Synaptic	59
6.2.4. Modo texto apt y apt-get	61
6.3. Desde paquetes preconfigurados	62
6.3.1. Paquetes Autopackage (de extensión .package)	63

6.3.2. Archivos binarios (.bin)	63
6.3.3. Archivos Run (.run)	63
6.4. Desde código fuente	63

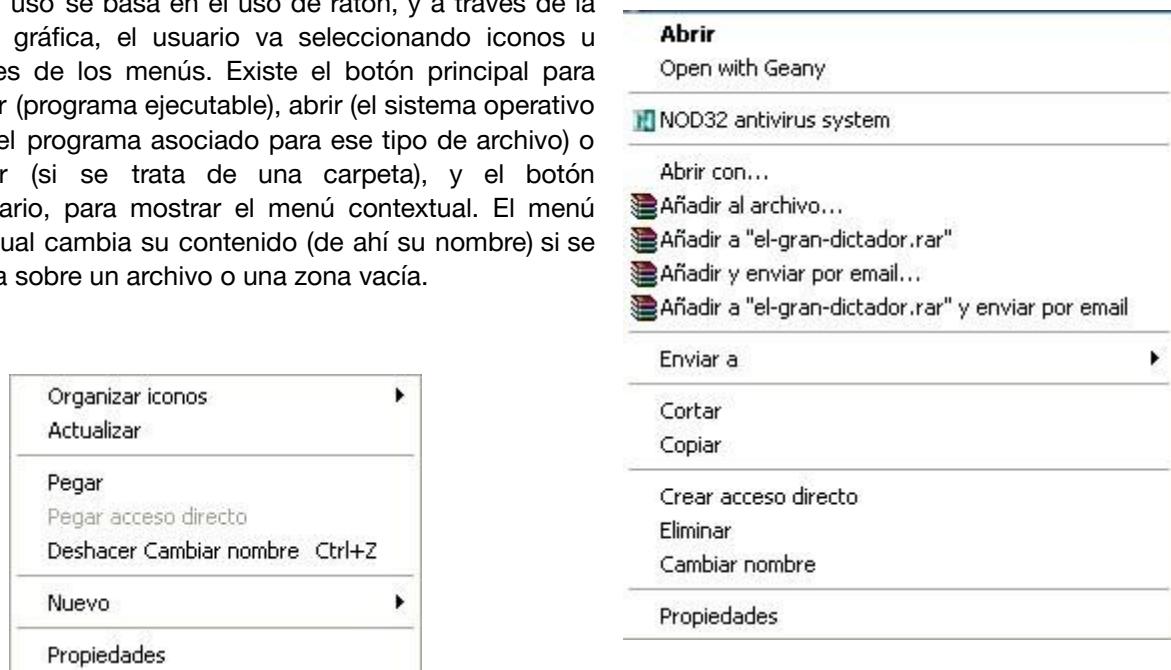
## 1. Primeros pasos con el sistema operativo.

### 1.1. Interacción del usuario con el sistema operativo.

Una de las funciones más importantes de un sistema operativo es la interacción del usuario con el sistema operativo. Esta interacción se puede realizar de 2 maneras, utilizando la interfaz gráfica o el intérprete de comandos.

### 1.2. Mediante la interfaz gráfica.

Su uso se basa en el uso de ratón, y a través de la interfaz gráfica, el usuario va seleccionando iconos u opciones de los menús. Existe el botón principal para ejecutar (programa ejecutable), abrir (el sistema operativo busca el programa asociado para ese tipo de archivo) o acceder (si se trata de una carpeta), y el botón secundario, para mostrar el menú contextual. El menú contextual cambia su contenido (de ahí su nombre) si se muestra sobre un archivo o una zona vacía.



Menú contextual sobre una zona vacía en Windows. Menú contextual sobre un archivo en Windows.

Para ejecutar aplicaciones en Windows (una vez iniciada la sesión) existen 4 maneras de lanzar aplicaciones:

1. El más habitual, es desde inicio, e irse desplazando hasta encontrar la aplicación deseada.
2. Muchas aplicaciones acostumbran a incluir un ícono en nuestro Escritorio. Para arrancar esas aplicaciones solo es necesario hacer doble click.
3. Otra forma es navegando por nuestro PC hasta encontrar con el programa deseado.
4. Por último, podemos ejecutar aplicaciones desde el menú de inicio y el cuadro de ejecutar.

Para ejecutar aplicaciones en Linux (una vez iniciada la sesión), es muy parecido a Windows y depende del entorno de escritorio. En Unity, el entorno utilizado en la distribución de Ubuntu, se puede arrancar:

1. Desde dash o tablero, y buscar la aplicación deseada.
2. Desde terminal, tecleando el nombre del ejecutable.

3. En Linux no es usual que las aplicaciones incluyan iconos en el escritorio al

instalarse. Ahora bien, podemos incluirlos nosotros:

El programa en Windows que gestiona los menús se llama Explorer.exe. En Linux existen varios entornos como nautilus (entorno gnome), Konqueror (entorno kde) o Thunar (entorno xface).

### **1.3. Intérprete de comandos.**

Se entiende por intérprete de comandos (también denominado Shell, consola) el sistema encargado de leer órdenes que teclea el usuario y convertirlas en instrucciones que el sistema operativo interpreta. A pesar de que el empleo de interfaces gráficas en la interacción del usuario con el sistema operativo ha ganado terreno gracias a lo intuitivo de su uso, el intérprete de comandos, que fue el elemento central de los primeros sistemas operativos de usuario, sobrevive hoy por razones tanto históricas como de seguridad y eficiencia.

El funcionamiento del intérprete de comandos es, con salvedades que ya veremos, siempre el mismo: el comando, las opciones y los parámetros que queremos ejecutar. Los campos de opciones y los parámetros son optativos y no tienen por qué aparecer. El comando siempre se acaba con la orden Intro (Enter) y el sistema procede a ejecutarlo.

Algunos comandos tienen el mismo nombre en Windows y en Linux, pero hemos de ser cautelosos puesto que siempre tendrán diferencias, en ocasiones sutiles.

Por ejemplo, es común a ambos sistemas, los comandos Ctrl+C para abortar el comando y el comando exit para salir de una sesión de comandos.

Para obtener la lista de comandos internos de Windows, te teclea help. Para obtener información detallada de los comandos y algunas de las herramientas se teclea la opción /?. Por ejemplo ping /? (ping no es un comando es un programa).

En Linux también se muestran los comandos de sistema con el comando help. Para mostrar información, incluso muy detallada, se teclea man seguido del comando (o aplicación), del que desea información.

### **Inicio y fin de sesión.**

En Windows el inicio y fin de sesión no tiene mucha complicación, durante el inicio, se van cargando procesos hasta que se inicia el proceso winlogon.exe, que es responsable de manejar la "Secuencia de atención segura" (SAS), cargar el perfil de usuario al inicio de sesión, y opcionalmente bloquear al sistema cuando un protector de pantalla se está ejecutando (requiriendo otro paso de autenticación).

El fin de sesión consiste en acabar con todos los procesos de usuario, ya sea para cerrar sesión, cambiar de usuario, apagar o reiniciar. En el modo suspender, el equipo baja de prestaciones, bajando el consumo y apagando el disco duro y los periféricos.

Para apagar Windows, existe la orden shutdown.exe, que necesita una serie de parámetros, entre los que destacan:

/l Cierra sesión.

/s Apaga el equipo.

/t XXX Espera XXX segundo a apagar el equipo.

/a Anula el apagado pendiente.

/r Apaga y reinicia el equipo.

-s -t 12

El proceso de arranque en Linux se realiza a través del proceso init. Es el primer proceso que se ejecuta en el sistema, es el más importante, del que dependen el resto de todos los demás procesos. A este tiene siempre asignado el PID número 1. Este programa, ahora como proceso, cargará los subprocessos necesarios para la puesta en marcha del sistema. Cuando init haya terminado de cargarse vaciará el subdirectorio /tmp y lanzará a getty que es el encargado de permitir a los usuarios hacer login en el sistema.

En Linux existen varias formas de apagar o reiniciar la máquina. Así para apagar existe:

halt

shutdown

n

poweroff

init 0

Y para reiniciar existen varias formas

reboot

shutdown -r

now init 6

Otra forma es terminar con el proceso init, esencial para el funcionamiento del sistema; matar este proceso rearrastra el sistema inmediatamente, pero en algunos sistemas el kernel entra en pánico. Es un método salvaje; usar shutdown o reboot.

Existen en ambos sistemas, herramientas de recuperación que se utilizan para reparar problemas de inicio, ejecutar diagnósticos o restaurar el sistema. Esta opción solo está disponible si las herramientas están instaladas, lo que normalmente ocurre en una instalación por defecto.

El modo seguro inicia Windows en un estado básico, que usa un conjunto limitado de archivos y controladores. En este modo puedes solucionar problemas del equipo. Por ejemplo, si el problema no se produce en el modo seguro, sabrás que la configuración predeterminada y los controladores básicos del dispositivo no están causando el problema.

En Windows 7, las opciones aparecen pulsando durante el inicio F8.

En Windows 10, para acceder en modo seguro se debe reiniciar el equipo. Cuando se muestre la pantalla de inicio de sesión, presiona la tecla Mayús mientras

#### Configuración de inicio

Presione un número para elegir entre estas opciones:

Use las teclas de número o las de función F1-F9.

- 1) Habilitar depuración
- 2) Habilitar el registro de arranque
- 3) Habilitar vídeo de baja resolución
- 4) Habilitar modo seguro
- 5) Habilitar modo seguro con funciones de red
- 6) Habilitar modo seguro con símbolo del sistema
- 7) Deshabilitar el uso obligatorio de controladores firmados
- 8) Deshabilitar protección antimalware de inicio temprano
- 9) Deshabilitar reinicio automático tras error

Presione F10 para ver más opciones

Presione Entrar para volver al sistema operativo

seleccionas Inicio/Apagado Icono de inicio/apagado y reiniciar.

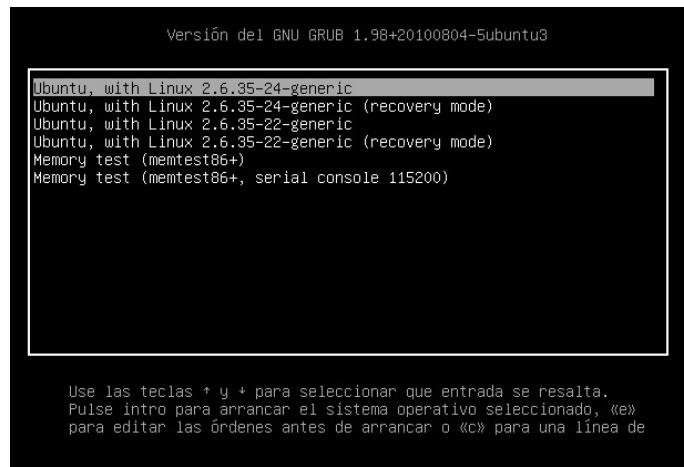
Cuando el equipo se reinicie, en la pantalla Elegir una opción, selecciona Solucionar problemas > Opciones avanzadas > Configuración de inicio > Reiniciar.

Una vez reiniciado el equipo, verás una lista de opciones. Selecciona 4 o F4 para iniciar el equipo en Modo seguro. O si necesitas usar Internet, selecciona 5 o F5 para el Modo seguro con funciones de red.

En Linux normalmente aparecen 3 opciones:

**Iniciar el sistema normalmente:**  
Para ellos seleccionamos las opciones que hacen referencia a la versión del kernel de nuestro sistema operativo.

**Iniciar el sistema en modo recuperación:** Podemos arrancar el equipo para trabajar en modo comando, de tal forma que no se carguen controladores que pudieran provocar un mal inicio del sistema. En este modo podremos realizar las operaciones de instalación, desinstalación de hardware, software o lo que pueda provocar que el equipo arranque mal.



**Test de memoria:** con esta opción, realizamos sencillamente un chequeo de la memoria RAM del equipo para confirmar que todo está correcto. Es una opción heredada de las versiones originales de UNIX, en las que el chequeo era muy importante.

#### 1.4. Editores de texto.

Una de las herramientas más elementales que proporciona los sistemas operativos es la posibilidad de editar un archivo de texto. En Windows el editor por defecto es el bloc de notas (notepad), aunque en modo consola no puede ser usado, y se usa el edit.

En Linux existen numerosos editores de texto, que pueden estar instalados o que requieran su instalación. Así como ejemplo están emacs, kate, gedit, kwrtie, etc. Pero de todos, el editor clásico es vi, disponible en todas las versiones de Linux.

#### Búsqueda de cadenas en ficheros de texto.

Para ver el contenido de un fichero de texto, en Windows se utiliza el comando type y en Linux existe el comando cat. En ambos sistemas existe el comando more que permite ver el contenido paginado, es decir, si ocupa más de una página, se detiene hasta que pulsamos una tecla.

Para buscar cadenas de texto tenemos en Windows tenemos el comando find, que busca las líneas que cumplen una condición, así, por ejemplo:

```
find provincias.txt "Andalucía"
```

En Linux el comando se denomina grep, que muestra las líneas que cumplen esa condición. Se utiliza muchas veces con el comando cat como veremos a continuación, pero

también se puede utilizar solo de la siguiente manera:

```
grep patrón fichero.txt
```

### 1.5. Redirección.

En principio, el resultado de un comando se muestra por pantalla. Si el contenido que sale por pantalla lo queremos almacenar en un archivo. Para ello se debe de direccionar. Existen 4 modos de direccionamiento, que son:

> **Direccionamiento de salida:** Hace que cambie la salida hacia el destino que le indicamos, así queda:

```
dir c: >listado.txt
```

>> **Adición a un fichero:** Hace que cambie la salida hacia el destino que le indicamos, agregándolo al final del archivo si el archivo ya existe, así queda:

```
dir c: >> listado.txt
```

< **Redirección de entrada:** Cambia la entrada del

```
comando date < fecha.txt
```

| **Redirección entre comandos:** Sirve para enviar la salida de un comando a la entrada de otro comando, así se puede acotar la salida de un comando y usarlo como entrada del comando more, por ejemplo:

```
dir | more
```

Así para buscar dentro de un archivo Linux un patrón el comando seria:

```
cat fichero.txt | grep "patron"
```

### 1.6. Histórico de comandos y expansión de teclado.

Tanto en Linux como en Windows podemos beneficiarnos del histórico de comandos y de la expansión de teclado.

El histórico de comandos se refiere a la posibilidad de consultar los comandos que llevamos ejecutados. Simplemente tenemos que teclear la flecha arriba ↑ del teclado, y en cada pulsación retrocedemos un comando de nuestro historial.

Otra gran utilidad que en sus últimas versiones también incluye Windows (únicamente en los path de los archivos), es la expansión de teclado. Esta función se utiliza mediante la tecla de tabulación y funciona de manera distinta en Windows, que en Linux. En Linux si se escribe los primeros caracteres del comando muestra todos los comandos con ese comienzo. Si lo que se teclea es una ruta, y solo existe un camino se completa. En Windows solo funciona con rutas y va mostrando todas las posibles soluciones.

## 2. Archivos y directorios.

### 2.1. Directorios o carpetas: introducción.

Dentro de un sistema informático, un directorio es un conjunto de archivos y/o directorios que se agrupan según su contenido, su propósito o cualquier otro criterio. Un directorio no es ni más ni menos que un tipo de archivo, que almacena información acerca de los archivos y directorios que contiene.

La figura del margen, podemos ver una estructura típica de almacenamiento de carpetas en el sistema de archivos en un entorno Windows.

En los sistemas operativos que manejan el entorno gráfico, que es lo habitual actualmente, el directorio recibe el nombre de carpeta.

#### Caminos o trayectorias.

Una ruta (en inglés *path*) es la forma de referenciar un archivo informático o directorio en un sistema de archivos. Una ruta señala la localización exacta de un archivo o directorio mediante una cadena de caracteres concreta. Ésta puede ser de diversas formas dependiendo del sistema operativo y del sistema de archivos en cuestión. Estos nombres estarán separados por un carácter delimitador que usualmente será "\\" en sistemas MS-DOS/Windows (Microsoft) o "/" en sistemas UNIX/Linux.

En la mayoría de los sistemas operativos y sistemas de archivos una ruta se puede expresar en forma relativa o en forma absoluta:

- Las **rutas absolutas** señalan la ubicación de un archivo o directorio desde el directorio raíz del sistema de archivos. Por ejemplo, es una ruta absoluta **/home/dir1/arc1.jpg** que señala la ubicación de **arc1.jpg** desde la raíz del sistema de archivos en Linux.
- Las **rutas relativas** señalan la ubicación de un archivo o directorio a partir de la posición actual del sistema operativo en el sistema de archivos. Por ejemplo, es una ruta relativa **dir1/arc1.jpg** que señala al archivo **arc1.jpg** dentro del directorio **dir1** en la ubicación actual. En sistemas tipo UNIX, la ruta **~/** es una ruta relativa que lleva al directorio personal del usuario que ha insertado la ruta relativa; por ejemplo, si el usuario **Juan** tiene una imagen en su directorio personal, esta imagen podría tener dos rutas de acceso, una relativa y una absoluta.

En Linux para conocer el directorio actual se utiliza el comando **pwd**.

### 2.2. Características de los directorios.

Al igual que los archivos, los directorios o carpetas tienen características que los clasifican, catalogan y ubican dentro del sistema de archivos. En general una carpeta o directorio queda determinado por las siguientes características:

1 – **Identificación.** Cada directorio está identificado por un nombre. El nombre es obligatorio y sus reglas quedan determinadas por el sistema de archivos de la versión del sistema operativo utilizado. Originalmente MS-

Un archivo no puede contener ninguno de los siguientes caracteres:  
\\ : \* ? " < > |

DOS tenía un límite de 8 caracteres, UNIX de 11 caracteres. En la actualidad, el nombre puede ser mayor, dependiendo del sistema de archivos, por lo general 255 caracteres. Existen una

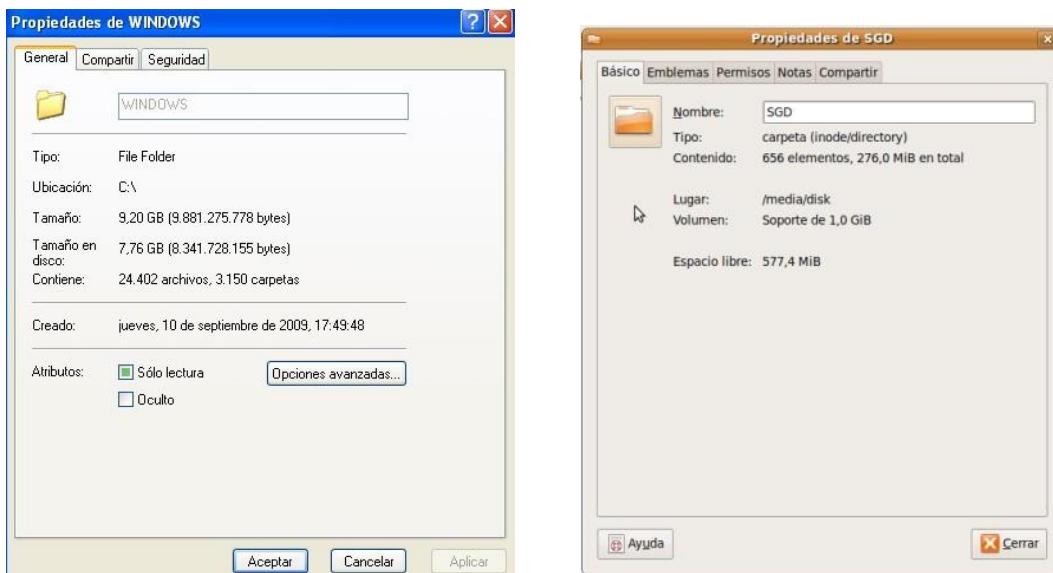


serie de caracteres no se pueden utilizar en el nombre.

La principal diferencia en Linux, ya en el nombre de los directorios, como de los archivos, como de los comandos, son diferentes en mayúsculas y en minúsculas. Es decir, pueden existir diferentes directorios (o archivos) en la misma ubicación con los nombres carpeta1,

Carpeta1, CARPETA1, etc. Se recomienda utilizar siempre minúsculas.

## 2 – Propiedades.



Las propiedades suelen ser comunes en ambos sistemas. Así algunos de ellos serían:

- Tamaño:** Los directorios tienen también un tamaño que se mide en bytes, kilobytes, megabytes y gigabytes, dependiendo de la carpeta.

**Ubicación:** Todo directorio tiene que estar almacenado necesariamente dentro de un directorio o subdirectorío. Todos los directorios o carpetas siempre colgarán del directorio raíz y quedará referenciado por la ruta en la que se encuentra.

3 – **Información sobre el directorio:** Cada directorio, al igual que cada archivo, tendrá una fecha de creación.

4 – **Atributos:** Propiedades o permisos que se tienen sobre la carpeta.

Sobre los directorios se pueden realizar una serie de operaciones, que son crear, consultar, actualizar, borrado y renombrado.

### Directarios especiales.

Dentro de todo sistema de archivos, existen 3 directorios que se consideran especiales dentro de la estructura:

- **Directorio raíz:** Directorio inicial de toda la estructura de directorios y del que depende el resto de información almacenada en el soporte. En sistemas Windows, existe un directorio raíz por cada unidad de almacenamiento. En este sistema el directorio raíz viene indicado por una letra seguida de 2 puntos y de la barra que representa el directorio (:). Así las unidades se denominan A:\, C:\, etc.

En Linux (UNIX), la referencia al directorio raíz (root) se escribe con una barra de dividir (/), de la que cuelga el resto de directorios y archivos de la estructura.

- **Directorio actual (.)**. Es un directorio especial que hace referencia al directorio en el que estamos.
- **Directorio padre (..)**. Es un directorio especial que hace referencia al directorio padre del directorio en el que estamos. Si escribimos “..”, nos estaremos refiriendo al directorio padre.

### Directarios importantes.

La información en cualquier sistema informático se ordena por carpetas y con diferentes nombres, pero con funciones similares en Windows y en Linux. Así las principales carpetas en Windows son:

- C:\Archivos de programa (Program files): Será el directorio donde, por defecto, se

instalarán las aplicaciones de usuario. Se suele crear una carpeta para cada aplicación que se instala en el sistema.

- C:\Windows: Es el directorio donde se instala el sistema operativo. Contiene los ficheros ejecutables del sistema operativo como el fichero dir.exe.
- C:\users: Será el directorio donde se guardarán los perfiles locales de usuario. En general, podemos ver que esa carpeta consta de los siguientes subdirectorios.
  - ❖ Administrador (administrador): Correspondiente al superusuario.
  - ❖ All Users: Contiene los elementos comunes a todos los usuarios (por ejemplo, algún ícono del escritorio).
  - ❖ Usuario. Corresponde a un usuario normal del sistema. Existe una carpeta con su nombre para cada usuario.
  - ❖ Default User: es una carpeta oculta que copia cuando se crea un nuevo usuario.

Dentro de cada carpeta de usuario, existe una para el escritorio, menú inicio, mis documentos datos de programa, favoritos, etc.

En Linux esta distribución puede cambiar dependiendo de la distribución. Nombraremos los de Ubuntu por ser la distro utilizada en clase.

**/ (raíz):** Es el nivel más alto dentro de la jerarquía de directorios. De aquí cuelgan el resto de carpetas, particiones y otros dispositivos. Es por esto que donde se instala el sistema, se selecciona la partición deseada y se le indica que el punto de montaje es justamente /.

**/bin (binarios):** Los binarios son los ejecutables de Linux. Aquí tendremos los ejecutables de los programas propios del sistema operativo, entre ellos comandos como cp, mv, cat, chown, etc. No es el único directorio que contiene ejecutables como veremos más adelante.

**/boot (arranque):** Aquí nos encontramos los archivos necesarios para el inicio del sistema, desde los archivos de configuración de Grub Lilo, hasta el propio kernel del sistema.

**/dev (dispositivos):** Linux se basa en la simpleza y en el tratamiento homogéneo de la información. Linux trata los dispositivos como si fueran un fichero más, para facilitar el flujo de la información. En esta carpeta tenéis los dispositivos del sistema, por ejemplo, los usb, sda (o hda) con sus respectivos números que indican las particiones, etc.

**/etc (etcétera):** Aquí se guardan los ficheros de configuración de los programas instalados, así como ciertos scripts que se ejecutan en el inicio del sistema. Los valores de estos ficheros de configuración pueden ser complementados o sustituidos por los ficheros de configuración de usuario que cada uno tiene en su respectivo "home" (carpeta personal).

**/home (hogar):** Este hogar no es más que un directorio que a su vez contiene otros, uno por cada usuario dado de alta en el sistema. Dentro de dichos directorios es donde el usuario tiene su carpeta personal, donde están los ficheros de configuración de usuario, así como los archivos personales del mismo que puede crear, modificar y eliminar bajo su propio criterio.

**/lib (bibliotecas):** Contiene las bibliotecas (también mal conocidas como librerías) del sistema, así como módulos y controladores (drivers).

**/lost+found (perdido y encontrado):** Es una carpeta que nos podemos encontrar en todas las particiones. Cuando por cualquier circunstancia se cierra mal el sistema (un apagón, por ejemplo), cuando éste se reinicie comprobaréis que se llamará al programa fsck para restaurar la integridad del sistema de ficheros. En esta carpeta encontraremos la información que se mal-guardó debido a la incidencia.

**/media (media/medios):** Es donde se montan las unidades extraíbles como los

dispositivos USB, disqueteras, unidades de CD/DVD y en algunas distribuciones, como Ubuntu, las particiones adicionales.

**/mnt (montajes):** Es un directorio que se suele usar para montajes temporales de unidades.

**/opt (opcionales):** Destinado para guardar paquetes adicionales de aplicaciones.

**/proc:** Información para la virtualización del sistema de ficheros de Linux.

**/root:** Es el /home del administrador. Es el único /home que no está incluido -por defecto- en el directorio anteriormente mencionado.

**/sbin (binarios de sistema):** Son los ejecutables de administración, tales como mount, umount, shutdown...

**/srv (servicios):** Información del sistema sobre ciertos servicios que ofrece (FTP, HTTP...).

**/sys (sistema):** Información sobre los dispositivos tal y como los ve el kernel Linux.

**/tmp (temporales):** Es un directorio donde se almacenan ficheros temporales. Cada vez que se inicia el sistema este directorio se limpia.

**/usr:** Es el directorio padre de otros subdirectorios de importancia:

- o **/usr/bin:** Conjunto de ejecutables de la mayoría de aplicaciones de escritorio entre otras (por ejemplo, firefox).
- o **/usr/include:** Los ficheros cabeceros para C y C++.
- o **/usr/lib:** Las bibliotecas para C y C++.
- o **/usr/local:** Es otro nivel dentro que ofrece una jerarquía parecida al propio directorio /usr.
- o **/usr/sbin:** Otra serie de comandos administrativos para el sistema.
- o **/usr/share:** Archivos compartidos como ficheros de configuración, imágenes, iconos, etc.
- o **/usr/src:** Tiene en su interior el código fuente para el kernel Linux.

**var:** Ficheros de sistema como el buffer de impresión, logs...

- o **/var/cache:** Se almacenan datos cacheados para las aplicaciones.
- o **/var/lib:** Información sobre el estado actual de las aplicaciones, modificable por las propias aplicaciones.
- o **/var/lock:** Ficheros que se encargan de que un recurso sólo sea usado por una aplicación determinada que ha pedido su exclusividad, hasta que ésta lo libere.
- o **/var/log:** Es uno de los subdirectorios más importantes ya que aquí se guardan todo tipo de logs del sistema.
- o **/var/mail:** Los correos de los usuarios.
- o **/var/opt:** Datos usados por los paquetes almacenados en /opt.
- o **/var/run:** Información sobre el sistema desde que se inició.
- o **/var/spool:** Datos esperando a que sean tratados por algún tipo de proceso.
- o **/var/tmp:** Otro fichero temporal.

### **2.3. Crear carpetas o directorios.**

Para crear una carpeta en modo gráfico, tanto en Windows como en Linux, simplemente nos posicionamos en el lugar que deseamos la carpeta, y desde el menú contextual crear nueva carpeta en Linux o en Windows nueva ⇒ Carpeta. En ambos casos

nos pide el nombre de la carpeta creada.

Desde línea de comando de Windows para crear una carpeta el comando es mkdir o el comando md, con la siguiente sintaxis:

`mkdir [unidad:] ruta` Así si se desea carpeta llamada prueba, en el directorio activo, se ejecutaría

`mkdir prueba.`

`md [unidad:] ruta` Así si se desea carpeta llamada prueba, en el directorio activo, se ejecutaría

`md prueba.`

Para crear un directorio en Linux también se utiliza el comando mkdir.

`mkdir [unidad:] ruta` Así si se desea carpeta llamada prueba, en el directorio activo, se ejecutaría

`mkdir prueba.`

#### 2.4. Eliminar directorio o carpetas.

Desde cualquier entorno gráfico, para eliminar un directorio, desde el menú contextual, eliminar o mover a la papelera. También podemos seleccionar la carpeta y presionar **Supr**. Aparece un mensaje para confirmar la operación.

Desde la línea de comando en Windows, para eliminar una carpeta el comando es rmdir o su abreviatura rd. Existen 2 restricciones, no se puede borrar el directorio activo y por defecto no borra una carpeta si esta contiene algún archivo o carpeta. Para borrar una carpeta y su contenido será necesario el parámetro /s.

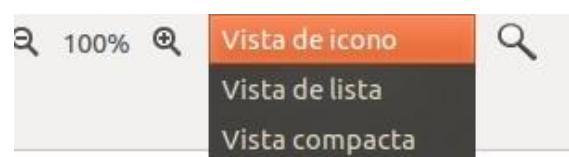
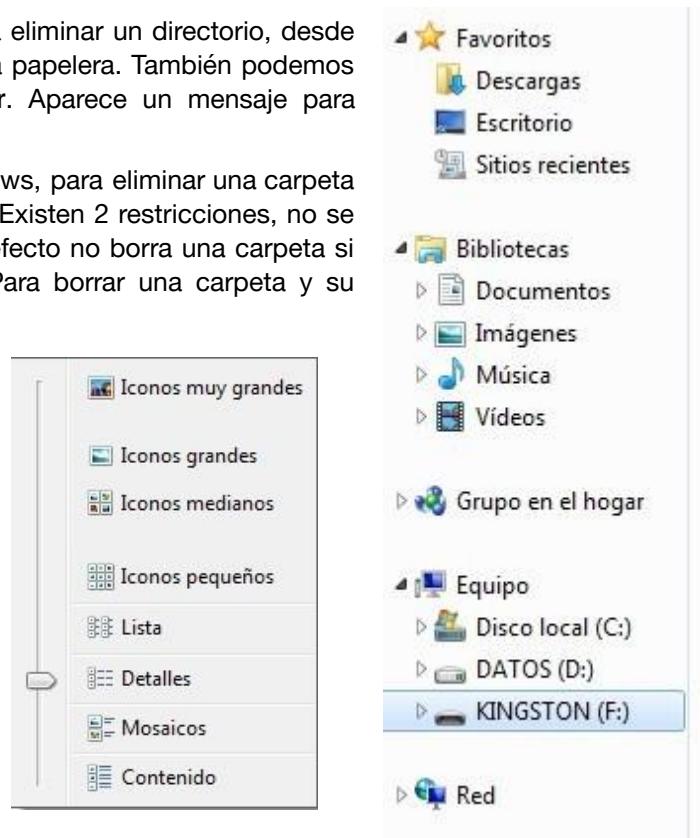
En Linux el comando es también rmdir, tampoco se pueden borrar carpetas con contenido sino se especifica con -R.

#### 2.5. Ver el contenido de las carpetas y desplazarse por ellos.

Desde el modo gráfico, la forma de mostrar el contenido de una carpeta simplemente nos tenemos que posicionar en la carpeta. Lo más interesante es seleccionar los modos de visualización, tanto en Windows como en Linux. Así se quiere ver con detalles, lista, etc.

Para ver el contenido en modo consola, debemos utilizar los comandos dir en Windows y ls en Linux. Estos comandos permiten varios parámetros para ver archivos ocultos, mostrar atributos, para ordenarlo, etc.

Para cambiar de directorio existe tanto en



Linux, como en Windows el comando cd. En Windows se puede utilizar sin abreviar el comando con chdir.

## 2.6. Visualizar la estructura de directorios.

Para visualizar la estructura en modo gráfico, se puede visualizar la estructura desde el explorador. Para ello desde el menú de ver ⇒ Barra del explorador ⇒ Carpetas y en el margen izquierdo aparece la estructura. Cuando en una carpeta aparece el signo + indica que la carpeta consta de más carpetas y se puede expandir. Con el signo – podemos agrupar el contenido.

Para mostrar la estructura en modo consola en Windows, el comando es tree. Así para mostrar la estructura de la carpeta Windows con direccionamiento absoluto:

```
tree c:\WINDOWS
```

Para mostrar la estructura en modo consola en Linux, el comando es ls con el parámetro -R. Así para mostrar la estructura de la carpeta /home con direccionamiento absoluto:

```
ls -R /home
```

## Mover y copiar directorios o carpetas.

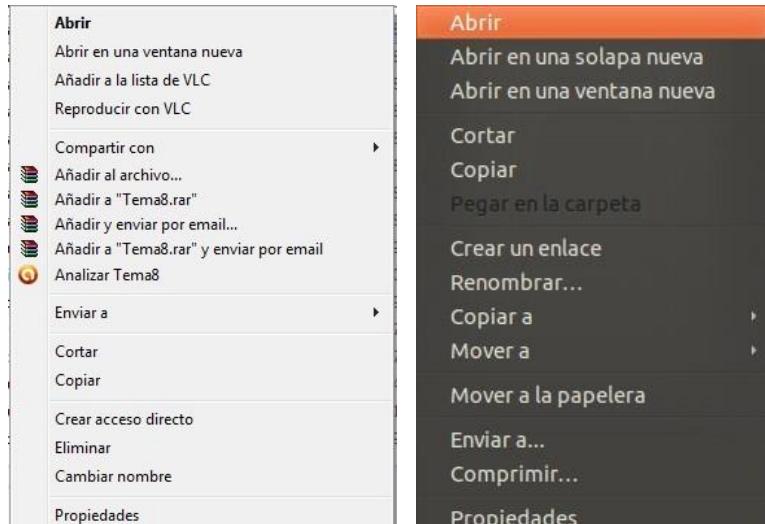
En el modo gráfico, solo es necesario usar el menú contextual desde la carpeta que deseamos mover (cortar) o copiar, y situarnos en la carpeta donde queremos situarla. Desde estos menús también se usarán para renombrar el nombre de la carpeta

Para mover una carpeta en Windows se utiliza el comando move.

```
move  
c:\prueba.txt d:\prueba2.txt
```

Desde la consola de Windows, para cambiar copiar una carpeta se emplea el comando xcopy. En principio solo copia los archivos ubicados en él, para copiar toda la estructura es necesario el parámetro /E. Su sintaxis es la siguiente:

```
xcopy carpetaOrigen carpetaDestino /E
```



Desde Windows vista existe una potente herramienta para replicación de directorios, disponible desde la línea de comandos, llamada robocopy. Esta herramienta por defecto, copia carpetas así para copiar una carpeta el comando seria:

```
robocopy C:\CarpetaOrigen C:\CarpetaDestino
```

/E Para mover una carpeta en Linux comando

```
mv. mv /home/arch1 /miDir/arc
```

Para copiar en Linux la orden es cp. Si se quiere copiar toda la estructura de directorios, es necesario el parámetro -R. Así para copiar la carpeta del usuario pepe a la carpeta del usuario juan

```
cp -R /home/pepe /home/juan
```

En cualquier caso, para el correcto uso de cualquiera de estas órdenes será necesario:

- Conocer donde estamos.
- Cuál es el origen de los datos.

- Cuál es el destino de los datos.
- Cuáles son los parámetros para realizar la operación deseada.

### Cambiar el nombre a un directorio o carpeta.

Para cambiar el nombre (renombrar), desde el entorno gráfico, nos situamos sobre la carpeta, y con el menú contextual, cambiar nombre (en Windows) o renombrar (en Linux).

Desde el modo comandos no es necesario comandos nuevos, ya se utilizan los mismos que mover (el concepto es el mismo), es decir move para Windows y mv para Linux.

En Windows se puede utilizar el comando rename (abreviado ren), para renombrar carpetas.

```
ren NombreAntiguo NombreNuevo
```

### Introducción a los archivos.

Los archivos, también denominados ficheros (o files en inglés), representan una colección de información (datos relacionados entre sí) localizada o almacenada en alguna parte del sistema de archivos.

Técnicamente hablando, un archivo es un flujo unidimensional de bytes tratados por el sistema operativo como una entidad única. Es un conjunto de bits (0 y 1) que refieren a algún tipo de información como un texto, un gráfico, sonido, etcétera.

Las operaciones que se pueden realizar en, son iguales en Windows que en Linux. Estas funciones son:

**Creación:** se genera el fichero, usando un programa de aplicación, y se le asigna simple un nombre y una ubicación.

**Consulta:** consiste en acceder al archivo y ver el contenido.

**Actualización:** los archivos cambian su contenido.

**Borrado:** un archivo que puede dejar de ser útil y debe de eliminarse para liberar el espacio que ocupaba en el disco duro.

**Renombrado:** consiste en cambiar el nombre o la extensión al archivo.

### Características de los archivos.

Todo archivo o fichero, al ser creado, tiene una serie de características:

**Nombre y extensión:** Cada archivo queda identificado por un nombre y una extensión. El nombre es obligatorio, pero la extensión es opcional y su función principal es diferenciar el contenido del archivo de modo que el sistema operativo disponga el procedimiento necesario para ejecutarlo o interpretarlo, sin embargo, la extensión es solamente parte del nombre del archivo y no representa ningún tipo de obligación respecto a su contenido.

**Información sobre el archivo:** Según el sistema de ficheros, de cada archivo o fichero se guarda la fecha de creación, la de modificación y la de último acceso. Al igual las carpetas, los archivos también poseen propiedades llamadas atributos.

**Tamaño:** Los archivos tienen también un tamaño que se mide en bytes, kilobytes, megabytes, gigabytes, etc., y que depende de la cantidad de caracteres (bytes) que contienen.

**Ubicación:** Todo archivo tiene que estar almacenado necesariamente dentro de un directorio o subdirectorío.

### Tipos de archivos.

En cualquier sistema operativo existen distintos tipos de archivo. En Windows se puede decir que existen 3 tipos de archivos:

**Directorios:** Es un contenedor virtual en el que almacenar una agrupación de

archivos de datos y otros subdirectorios, atendiendo a su contenido, a su propósito o a cualquier criterio que decida el usuario. Técnicamente el directorio almacena información acerca de los archivos que contiene: como los atributos de los archivos o dónde se encuentran físicamente en el dispositivo de almacenamiento.

**Archivos ejecutables:** En Informática, un ejecutable o archivo ejecutable, es un archivo binario cuyo contenido se interpreta por la computadora como un programa. Las extensiones comunes de archivos para campos ejecutables en Windows incluyen .exe, .com, .bat.

**Archivos ordinarios:** Son archivos comunes, cuya única misión es contener información en una estructura de datos. Así cualquier archivo .jpg o .xlsx son archivos ordinarios.

En Linux, existen un mayor tipo de archivos, así tenemos:

**Archivo ordinario:** Al igual que en Windows, son archivos de texto, de imágenes, etc.

**Directorios:** Como en Windows, son archivos que contienen referencias a otros archivos. Contiene información sobre archivos ordinarios, subdirectorio, etc.

**Vínculos simbólicos:** Su propósito es vincular archivos de forma simbólica, lo que significa que dichos archivos pueden o no apuntar a un archivo existente. Con mucha frecuencia se los conoce como soft links (en inglés), y están identificados por una 'l'.

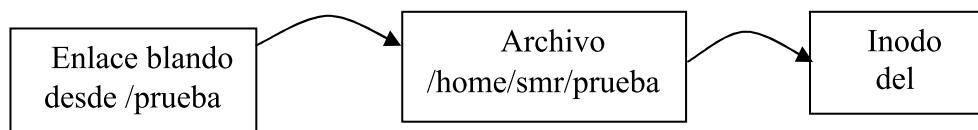
**Archivos especiales:** suelen representar dispositivos físicos, como unidades de almacenamiento, impresoras, terminales, etc. Estos se dividen en:

- o **Archivos de modo carácter** Estos archivos son o bien archivos especiales del sistema (tal como /dev/null), o bien periféricos (puertos serie o paralelo), que comparten la particularidad de que su contenido (si es que tienen alguno) no está en un buffer (es decir, que no se conservan en memoria). Dichos archivos se identifican con la letra 'c'.
- o **Archivos de modo bloque** Estos archivos son periféricos, y a diferencia de los archivos de modo carácter, su contenido está conservado en memoria. Los archivos que entran en esta categoría son, por ejemplo, los discos duros, las particiones de un disco duro, las unidades de disquete, las unidades de CD-ROM y así sucesivamente. Los archivos /dev/hda, /dev/sda5 son un ejemplo de archivos de modo bloque. En la salida de ls -l, estos están identificados por la letra 'b'.
- o **Tuberías nombradas** Estos son muy similares a las tuberías usadas en los comandos del shell, pero con la particularidad que éstas, en realidad, tienen nombre. Sin embargo, son muy raras, y es muy poco probable que vea una durante su viaje por el árbol de archivos. Sólo en caso de que los vea, la letra que las identifica es 'p'.

Dentro de los enlaces existen 2 tipos, enlaces duros (o hard link) y los enlaces simbólicos (blandos o soft link). La diferencia en ellos es la siguiente:

#### **Enlaces simbólicos.**

Un enlace simbólico, en sistemas unix o Linux, indica un acceso a un directorio o fichero que no es real, sino un enlace a otro, que se encuentra en un lugar distinto dentro de la estructura de directorios. Una modificación realizada utilizando este enlace se reflejará en el original; pero, por el contrario, si se elimina el enlace, no se eliminará el auténtico. Se puede asimilar a un acceso directo en Windows.



Es decir, se puede acceder al archivo desde 2 lugares diferentes del sistema de ficheros.

Si se elimina el archivo, el enlace blando no apuntara a una referencia a un archivo que no existe, con lo que no podré acceder.

Para crear un enlace simbólico se utiliza el comando ln con el parámetro -s:

*ln -s archivoOrigen archivoDestino*



**Un ejemplo**

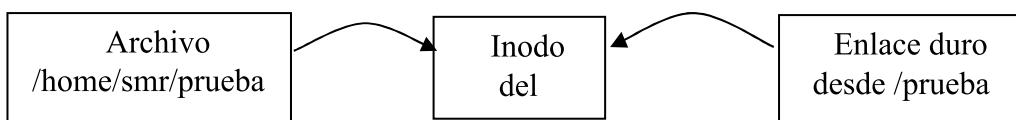
```
In -s /etc/passwd /home/usuario/claves
```

El archivo de /home/usuario/claves se creará y apunta al archivo /etc/passwd

En los enlaces simbólicos se puede utilizar un directorio como archivo origen. Si un archivo es un enlace simbólico, en el entorno gráfico aparece con una flecha en la parte inferior.

**Enlaces duros.**

Un enlace duro es un acceso al inodo y es indistinguible del real. Es decir, accedemos desde 2 ubicaciones distintas al mismo inodo, y por lo tanto al mismo archivo. El archivo no se borrará mientras este referenciado desde un enlace duro. Este tipo de enlace no existe en Windows.



Para crear un enlace duro el comando es ln pero sin parámetro.

```
In archivoOrigen archivoDestino
```

**Un ejemplo**

```
In /etc/passwd /home/usuario/claves
```

El archivo de /home/usuario/claves se creará y apunta al archivo /etc/passwd.

**Atributos y permisos de los archivos.**

Todos los sistemas de archivos proporcionan métodos para proteger los archivos de daños accidentales o intencionados. También otorgan permisos para identificar las operaciones que se pueden realizar con ellos.

**Windows.**

Los atributos en Windows son los siguientes:

Solo lectura (read only): Se especifica con una R y un directorio que tenga este atributo no se podrá borrar ni modificar, solamente se podrá visualizar su contenido.

De sistema (System) Se especifica con una S. Establece el directorio como directorio de sistema, teniendo unas características especiales, como que ésta oculto y que solamente se puede leer. En directorios no es habitual este atributo, pero si en archivos.

Oculto (Hidden): Se especifica con una H. Provoca que el directorio permanezca oculto durante las operaciones normales sobre el directorio.

En Windows, dependiendo de los atributos, se muestran con diferente intensidad. Para conocer los atributos en modo comando, se utiliza el comando attrib, que, sin parámetros, muestra los atributos de todos los ficheros del directorio activo, y con una ruta visualiza los atributos del fichero especificado.

Para asignar o eliminar atributos en modo gráfico, se escoge la carpeta ⇒ propiedades y se modifican el atributo deseado de entre los que deja modificar que son solo lectura y oculto. Por línea de comandos, el formato es el siguiente:

attrib +R Carpeta donde + o – para poner o eliminar el atributo.

atributos R para decidir que atributo queremos cambiar. Existen los siguientes 3

R Solo lectura

S Directorio de sistema.

**H Oculto**

Carpeta el directorio al que se va aplicar.

**Linux**

En Linux, los atributos son más importantes y completos que en Windows. Para ellos se utilizan los permisos, que también son válidos para archivos.

Desde el modo gráfico, se selecciona el archivo (o carpeta), y desde el menú contextual, propiedades y la pestaña de permisos se muestran y modifican los permisos.

Desde el modo texto los atributos en Linux, se muestran con el comando ls, pero usando distintos parámetros, así 2 principales son:

- a      Para mostrar los ocultos.

-l      Muestra un listado en el formato largo, con información de permisos, número de enlaces asociados al archivo, usuario, grupo, tamaño y fecha de última modificación además del nombre.

Los ficheros ocultos en Linux, empiezan por un punto(.)

Al visualizar con -l, se muestran 3 apartados, Propiedades o permisos, atributos y archivos:

```

smr@smr-VirtualBox:~$ ls -l
total 36
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Descargas
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Documentos
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Escritorio
-rw-r--r-- 1 smr smr 179 2011-01-09 19:04 examples.desktop
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Imágenes
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Música
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Plantillas
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Público
drwxr-xr-x 2 smr smr 4096 2011-01-09 19:52 Videos
smr@smr-VirtualBox:~$
```

**Permisos****Permisos.**

Los permisos se definen por una serie de 10 caracteres con el siguiente significado. El primer carácter tiene el siguiente significado.

CARÁCTER	IDENTIFICA
-	Archivo ordinario
d	Directorio
b	Archivo de bloques especiales.
c	Archivo de caracteres especiales.

I	Archivo de vínculo o enlace.
---	------------------------------

p	Archivo especial de cauce
s	Archivo especial de socket

Los siguientes 9 caracteres son los permisos. Se agrupan en grupos de 3 en 3 con el siguiente significado:

Permiso	Significado en archivos	Significado en directorios
-	Sin permiso.	Sin permiso.
r	Permiso de lectura.	Se puede listar su contenido.
w	Permiso de escritura.	Puede crear o borrar archivos en ese directorio.
x	Permiso de ejecución.	Se puede ser el directorio activo.

El primer bloque de 3 determina los permisos del propietario, los 3 siguientes los permisos de grupo del propietario y los 3 últimos los permisos que tendrán el resto de usuarios del sistema.

Así una cadena de permisos

como drwxr-xr-- indica:

d Es un directorio.

rwx El propietario puede leer, escribir y ejecutar.

r-x El resto de miembros del grupo, tiene los permisos de lectura y ejecución. r-- El resto de usuario solo tiene el permiso de lectura.

Para modificar los permisos se utiliza el comando chmod, pero existen 2 formas de ejecutar el comando:

Definir todos los permisos mediante un número en octal. Dependiendo del permiso se le asigna un peso, con la siguiente equivalencia.

r = 4 (lectura)

w = 2 (escritura)

x = 1 (ejecución)

- = 0 (sin permisos)

Ejemplos:

**rw (600)** — Sólo el propietario tiene el derecho de leer y escribir.

**rw-r--r-- (644)** — Sólo el propietario tiene los permisos de leer y escribir; el grupo y los demás sólo pueden leer.

**rw----- (700)** — Sólo el propietario tiene los derechos de leer, escribir y ejecutar el archivo.

**rwxr-xr-x (755)** — El propietario tiene los derechos de leer, escribir y ejecutar; el grupo y los demás sólo pueden leer y ejecutar.

**rwxx--x--x (711)** — El propietario tiene los derechos de lectura, escritura y ejecución; el grupo y los demás sólo pueden ejecutar.

**rw-rw-rw- (666)** — Todo el mundo puede leer y escribir en el archivo. ¡No es una buena elección!

**rwxrwxrwx (777)** — Todo el mundo puede leer, escribir y ejecutar. ¡Tampoco es buena elección!

Así el comando quedaría de la siguiente forma:

`chmod 755 ArchivoCambioPermisos`

La otra forma es añadiendo o eliminando solo el o los permisos. Para ellos habrá que definir el permiso y grupo al que queremos modificarlo.

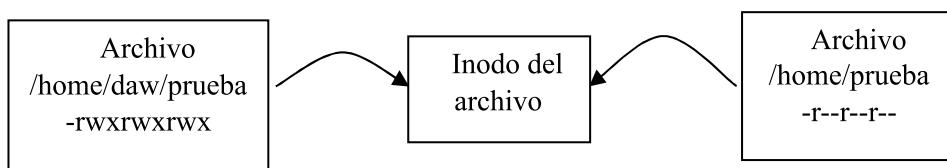
	Símbolo	Descripción
Identidades	u	Es el usuario propietario del archivo o directorio.
	g	Es el grupo al que pertenece el archivo o directorio.
	o	Resto de usuario.
	a	Todos los usuarios.
Permisos	r	Acceso de lectura.
	w	Acceso de escritura.
	x	Acceso de ejecución.
Acciones	+	Añade los permisos.
	-	Elimina los permisos.
	=	Asigna los permisos

Así para asignar el permiso de ejecución al resto de miembros del grupo a un archivo, se utilizaría de la siguiente forma:

`chmod g+x ArchivoCambioPermisos`

Los permisos del sistema de archivo sobre el enlace simbólico no tienen relevancia alguna: los permisos en el objeto al cual referencia el enlace simbólico son los que determinan los permisos de acceso.

En cambio, un enlace duro si tiene relevancia, ya que pueden ser distintos y si accedemos desde una ubicación tener derecho de modificar, y si se accede desde otro no.



Si queremos modificar el fichero desde `/home/prueba` nos va a decir que no tenemos permisos, mientras que, si accedemos al archivo desde la ubicación `/home/daw/prueba`, sí que podremos modificarlo.

```

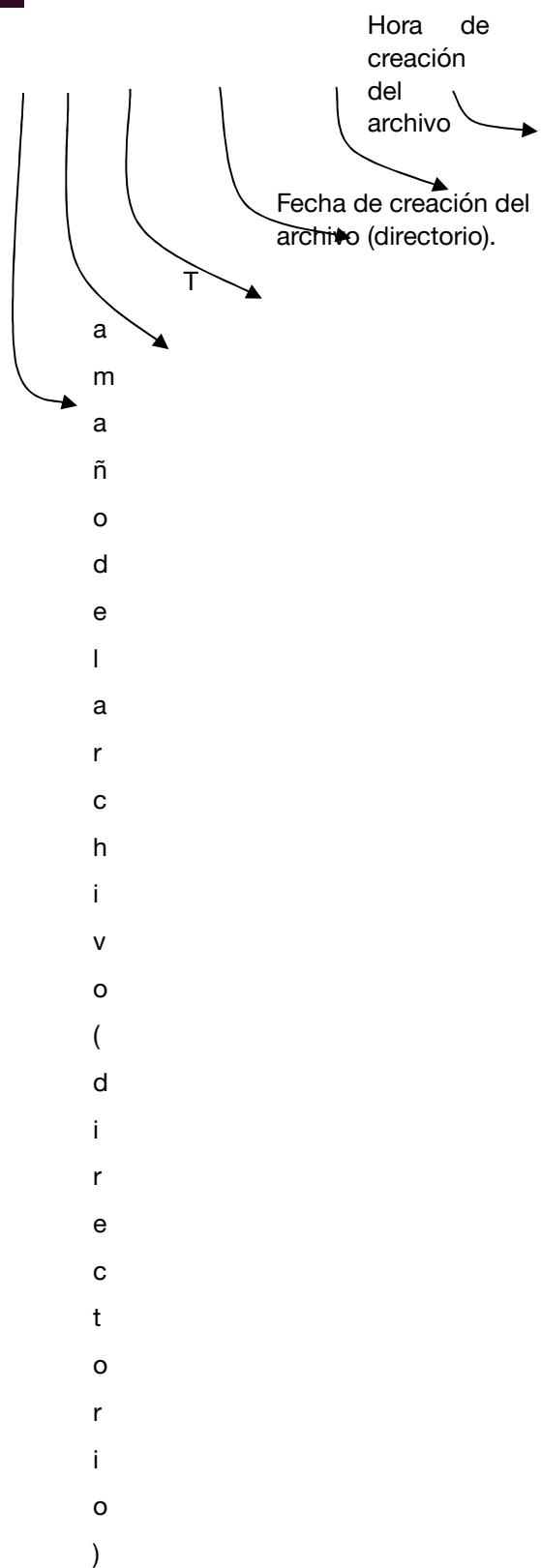
smr@UbuntuSMR:~$ ls -la
total 140
drwxr-xr-x 17 smr smr 4096 feb  8 17:20 .
drwxr-xr-x  4 root root 4096 feb  2 18:28 ..
-rw-r--r--  1 smr smr 3312 feb  2 18:52 .bash_history
-rw-r--r--  1 smr smr  220 ene  8 18:03 .bash_logout
-rw-r--r--  1 smr smr 3771 ene  8 18:03 .bashrc
drwx----- 13 smr smr 4096 ene 27 19:47 .cache
drwx----- 15 smr smr 4096 ene 25 16:05 .config
drwx-----  3 smr smr 4096 ene 25 16:03 .dbus
drwxr-xr-x  2 smr smr 4096 ene  8 18:50 Descargas
-rw-r--r--  1 smr smr   25 ene  8 18:50 .dmrc
drwxr-xr-x  2 smr smr 4096 ene  8 18:50 Documentos
  
```

**Atributos.**

Las siguientes cadenas son los atributos. El significado de cada apartado es el siguiente:

```
13 smr smr 4096 ene 27 19:47 .cache
```

(directorio).



G	p
r	i
u	e
p	t
o	a
d	r
e	i
l	o
p	.
r	Nombre del propietario.
o	Número de enlaces duros.

Por último aparece el nombre del archivo. Cuando se trata de enlaces simbólicos, aparece al lado del nombre del archivo, una flecha y la ubicación al archivo que referencia.

```
lrwxrwxrwx 1 smr smr 13 2011-01-24 23:19 prueba2 -> /home/smr/rrr
```

Los permisos en Linux también se pueden cambiar desde el entorno gráfico, seleccionando las propiedades del archivo, y la pestaña de permisos.

#### Caracteres comodines.

Los comodines se utilizan para identificar varios ficheros de una sola vez. Se llaman así porque pueden representar a uno o varios caracteres en los nombre de fichero. Los caracteres comodín son tanto en Windows como en Linux son:

**Carácter ?:** Representa cualquier carácter válido en el nombre de un fichero. Sólo sustituye o representa a uno de ellos:

**Carácter \*:** Representa uno o más caracteres válidos en el nombre de un fichero. Sustituye caracteres por el principio, por el final o por el centro del nombre especificado.

Estos 2 caracteres comodines, se pueden utilizar tanto en Linux como en Windows. Pero en Linux existen 2 caracteres más.

**[ ]:** Cuando se utilizan como parte de nombre de archivos o directorios representa un solo carácter de los incluidos entre los corchetes, que sustituirán al carácter en el nombre de archivo o directorio en la posición en la que estén estos corchetes. Pueden incluir rango de separador por un guión.

Para mostrar los dispositivos tty que van del 1 al 4, se emplearía el siguiente comando.

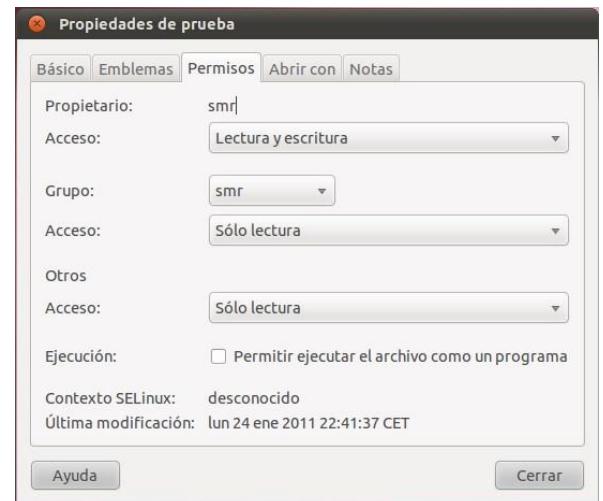
```
smr@smr-VirtualBox:~/Escritorio$ ls /dev/tty[1-4]
/dev/tty1 /dev/tty2 /dev/tty3 /dev/tty4
smr@smr-VirtualBox:~/Escritorio$
```

**!:** Permite la negación o exclusión de caracteres.

Para mostrar los archivos que no empiezan por E:

```
ls [!E]*
```

#### Operaciones con archivos.



### Crear archivos.

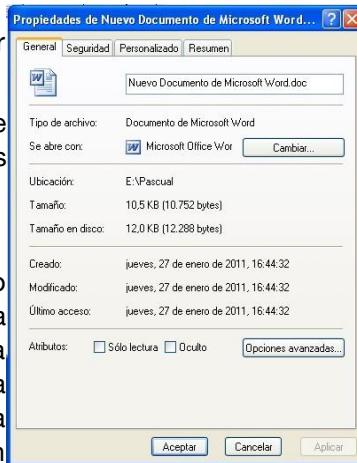
Los archivos en modo gráfico los crean los programas de aplicaciones. No obstante desde el menú contextual se pueden lanzar aplicaciones para generar archivos.

Desde modo consola solo se pueden crear archivos en Linux con los editores.

### Ver el contenido de un archivo.

Para ver el contenido de un archivo es necesario abrir el archivo desde la aplicación asociada. En Windows, para cada extensión tiene asignado una aplicación. Se puede cambiar la aplicación asociada a esa extensión

desde las propiedades del archivo ⇒



Tipo de archivo ⇒ Cambiar y aparece una lista de los programas instalados en el equipo.

También se puede cambiar, cualquier extensión desde explorador ⇒ Herramientas ⇒ Opciones de carpeta ⇒ Tipos de archivo o desde Panel de control ⇒ Opciones de carpeta ⇒ Tipos de archivo.

El modo gráfico de Linux, nos muestra la aplicación por defecto y la opción de abrirla con el resto de aplicaciones del sistema. Para cambiar la aplicación por defecto, Propiedades del archivo ⇒ Abrir con.

Desde consola solo se pueden mostrar los archivos de texto plano. El comando en Windows es type y en Linux cat, o more.

### Cambiar el nombre a un archivo.

Desde modo gráfico, se utiliza el menú textual, cambiar nombre o renombrar.

Desde consola, en Windows el comando es rename (o su abreviatura ren), indicando el nombre antiguo y el nuevo.

`ren NombreAntiguo NuevoNombre`

En Linux, no existe un comando para renombrar, y se utiliza el mismo comando que para mover (mv).

### Copiar y mover archivos.

Para copiar y mover archivos en modo gráfico es igual a copiar y mover carpetas. Se utiliza el menú contextual, y se selecciona copiar o cortar, dependiendo de la operación que se desee realizar.

En modo consola en Windows, el comando es copy. Copia uno o más archivos desde la ubicación inicial a otro lugar. Si no se especifica el destino, la copia se realiza en el directorio actual. Este comando permite la utilización de comodines. Para manejar este comando se deberá de tener 4 cosas en cuenta:

Donde estamos situados.

Donde se encuentra el archivo que se va a copiar y su nombre.

Dónde queremos copiar el archivo.

Cuál es el nombre con el que queremos copiar el archivo, si es el mismo o no.

Para mover archivo se utiliza el mismo comando que para directorios, el comando move.

En Linux los comandos para copiar y mover son los mismos que para los directorios, cp y mv.

**Eliminar archivos.**

En modo gráfico, sea cual sea el sistema operativo, tecleando suprimir o desde el menú contextual, eliminar o mover a la papelera.

Desde Windows en modo consola existe 2 comandos, delete (o su abreviatura del), y el comando erase. El funcionamiento es sencillo, ya que con esta orden se puede eliminar un solo fichero o varios (utilizando los comodines). No se pueden eliminar directorios.

Con el comando del no se pueden eliminar archivos ocultos. Para eliminar archivos ocultos es necesario quitar los atributos necesarios y, después, eliminar el fichero de forma normal.

El comando en Linux para eliminar archivos es rm. Con el parámetro –r podemos eliminar carpetas, como si fuera el comando rmdir.

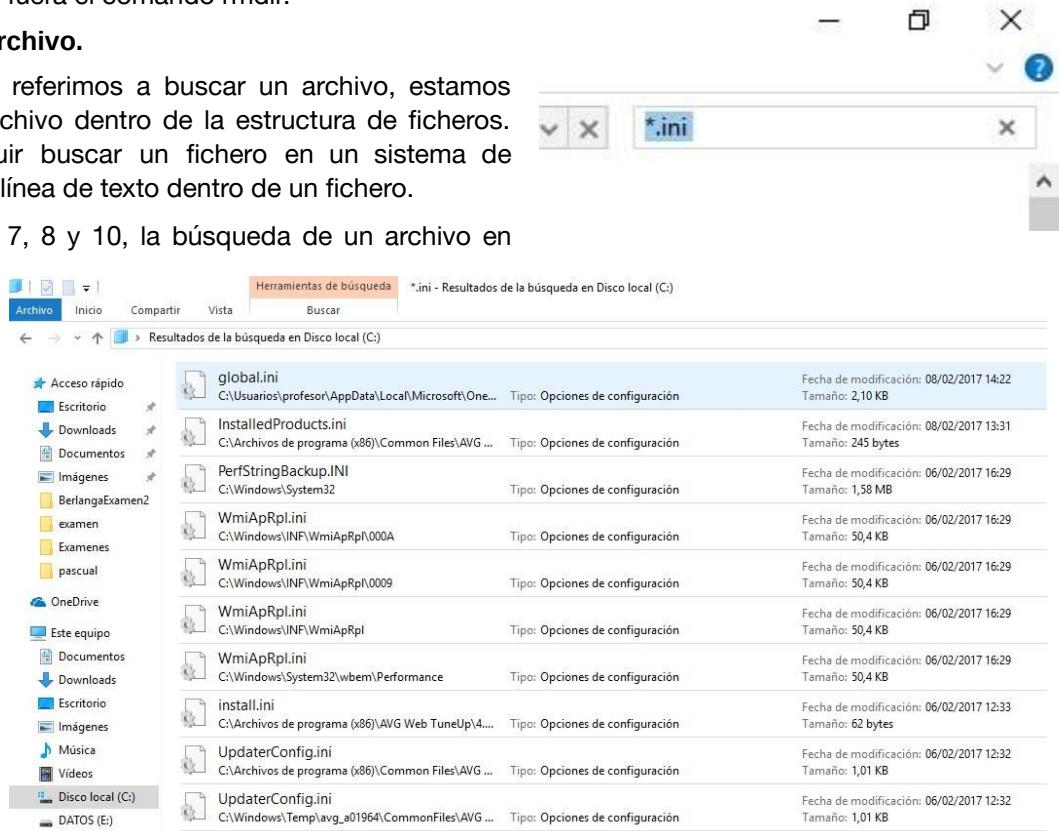
### Buscar un archivo.

Cuando nos referimos a buscar un archivo, estamos localizando un archivo dentro de la estructura de ficheros. Hay que distinguir buscar un fichero en un sistema de ficheros, que una línea de texto dentro de un fichero.

En Windows 7, 8 y 10, la búsqueda de un archivo en modo gráfico

cambia, la búsqueda se realiza dentro de Explorer y dentro de la carpeta que se esté mostrando en ese momento. Dentro de la típica visualización del contenido de la carpeta, en la esquina superior derecha aparece un cuadro de búsqueda, y es

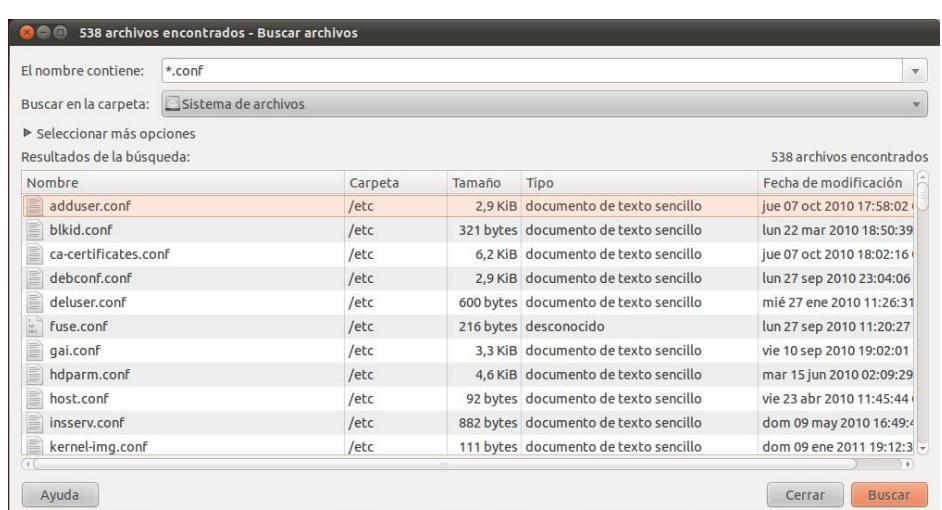
ahí donde se debe seleccionar las opciones de búsqueda. Dentro de las opciones de búsqueda se pueden utilizar los caracteres comodines de Windows.



En Linux también se puede utilizar el modo gráfico para buscar archivos.

En nuestro caso, en el entorno gnome, la búsqueda se realiza desde lugares, buscar archivos.

Aparece una ventana, donde se puede seleccionar el nombre



del archivo que  
deseamos buscar, y la  
ubicación  
donde deseamos realizar  
la

búsqueda.

Al definir el nombre podemos utilizar los caracteres comodines que existen en Linux.

Así se puede buscar los archivos de configuración, que acaban con .conf, o los archivos de configuración que empiezan por un número, con la secuencia de caracteres [0-9]\*.conf.

En Windows no existe un comando para buscar un archivo en un sistema de ficheros, pero con el comando dir, y el parámetro /s, muestra los archivos del directorio especificado y todos sus subdirectorios. Si la salida de ese comando lo direccionamos con el comando find (utilizado en Windows para buscar las líneas de un archivo que contiene un determinada cadena), tendremos la salida deseada. Así para mostrar todos los archivos con extensión exe de la partición c:, teclearíamos:

```
dir c:\ /s /b | find      ".exe"
```

En Linux el comando que se utiliza es find (con distinto funcionamiento que en Windows). Es de los comandos más complicados de Linux.

La sintaxis es la siguiente:

```
find      UbicacionDondeBusqueda Parámetros CondicionBusqueda
```

Dónde:

UbicacionDondeBusqueda, es la ubicación donde se desea buscar el archivo.

Parámetros: se especifica que se desea buscar. Los dos principales parámetros son:

-type f -name nm (nm es el nombre del fichero a buscar).

-type d -name dir (dir es el nombre del directorio a

buscar). CondicionBusqueda, es la condición utilizada para la búsqueda.

Así para buscar todos los directorios llamados

home. find / -type d -name home

Algunos ejemplos de los parámetros del comando find.

```
find / -type f -size +4M -size -7M      Buscamos archivos de más de 4
megas pero menos de 7 megas.
```

find / -type f -atime -1 Ficheros accedidos en las últimas 24 horas.

find / -type f -mmin -5 Ficheros modificados en los últimos 5 minutos.

find / -inum numerodeindo Ficheros con ese número de inodo.

Si tecleamos solo el comando find, nos muestra todos los archivos y utilizando el comando grep y redirecciónamientos, podríamos utilizar para buscar y filtrar.

```
find / | grep .conf  equivale a      find / -type f -name *.conf
```

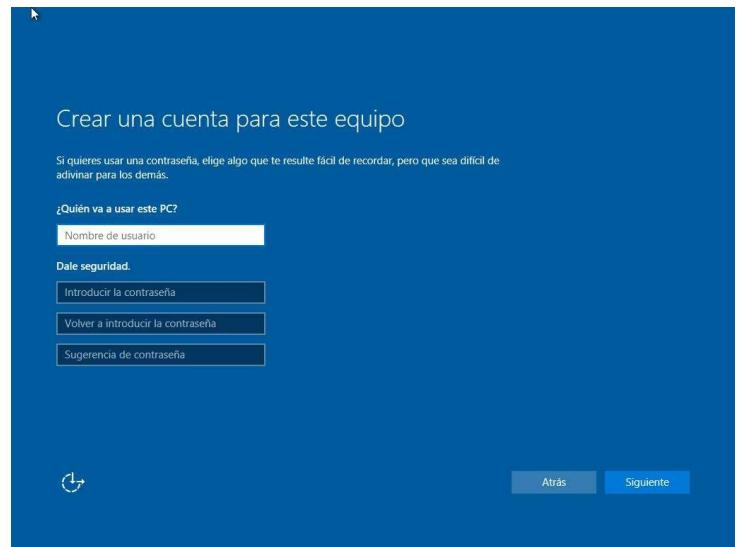
### 3. Gestión de usuarios.

Esta unidad los usuarios que vamos a gestionar son usuarios locales del sistema, es decir, su gestión, alta, baja y modificación solo afectará al equipo en que estemos trabajando y no al resto de equipos de la red. Los usuarios de red se estudiarán posteriormente.

#### 3.1. Usuarios y grupos en Windows.

En los sistemas operativos de Microsoft posteriores a Windows xp, se permite la gestión de usuarios locales y de grupos locales. A diferencia de Windows 9X o ME, necesitan siempre que un usuario se valide ante el sistema para poder trabajar en él. Este tiene que haber sido, además, dado de alta previamente por el único usuario que, en principio, tiene privilegios sobre el ordenador: el administrador.

Siempre que se realiza una instalación del sistema operativo, se instala una configuración específica para que una persona pueda iniciar sesión en el equipo y configurar lo que necesite.



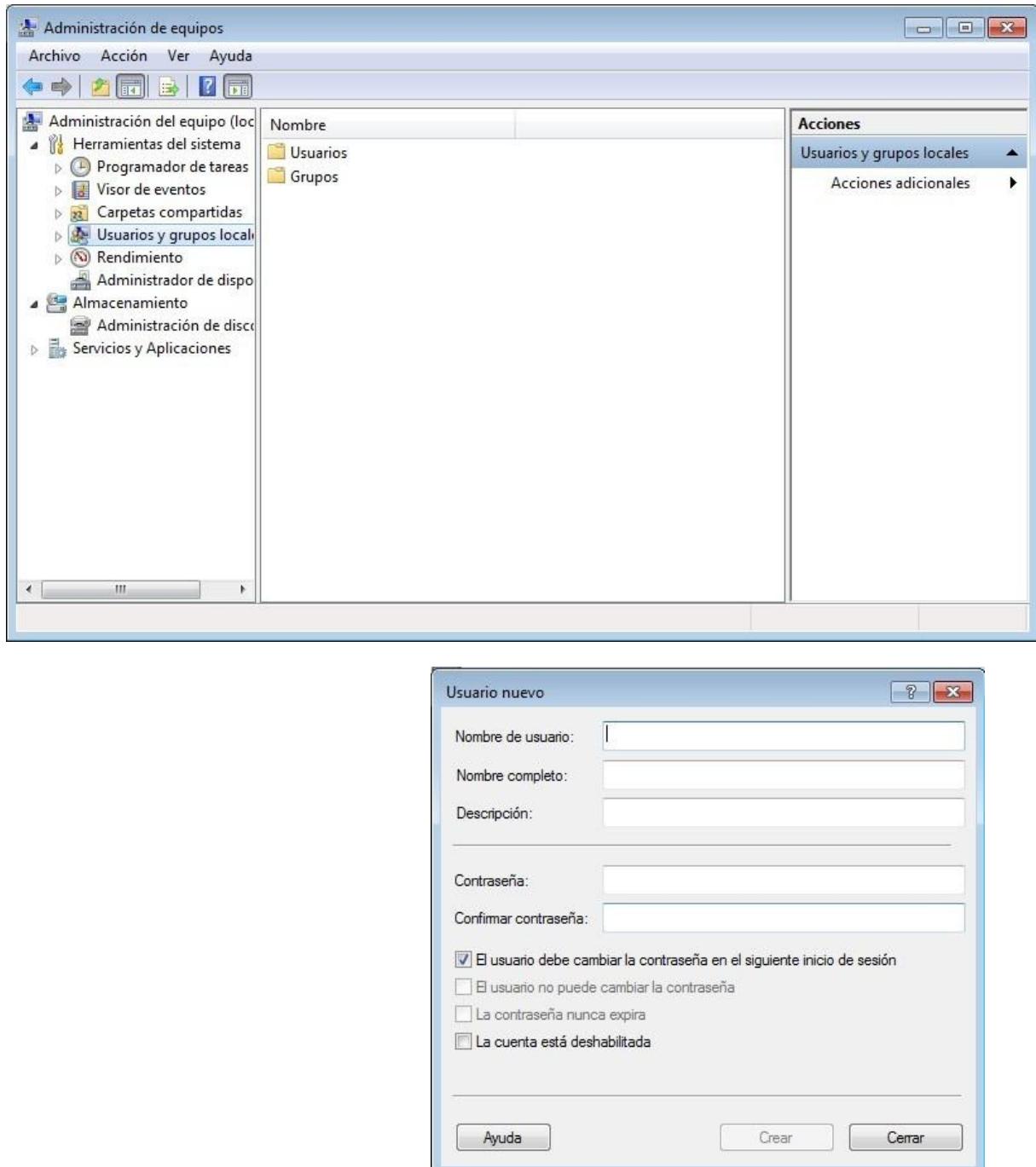
Durante la instalación se crean las credenciales para el usuario administrador del equipo local, que es quien tiene privilegios para realizar la mayoría de las operaciones. Este usuario será el que tiene privilegios sobre el sistema para realizar cuantas configuraciones considere oportunas.

En Windows xp, existía un fallo de seguridad, el usuario Administrador se crea automáticamente con contraseña admin.

En el equipo siempre tiene que existir un usuario administrador. Este usuario será el que inicie sesión por primera vez en el equipo y el tendrá todos los privilegios sobre el sistema para realizar cuantas configuraciones considere oportunas. Posteriormente se pueden crear más usuario con perfil administrador.

Para dar de alta un nuevo usuario local en el equipo, se puede realizar de 2 maneras:

- Desde el administrador del equipo, seleccionamos usuarios, usuario nuevo y llenar los campos necesarios.



- Desde el panel de control, cuentas de usuario, crear un nuevo usuario.

En cualquier caso se puede observar que existen al menos los siguientes usuarios.

- Administrador: Usuario que tiene todo el control sobre el ordenador local y puede crear, modificar y borrar configuraciones del mismo.
- Invitado: Usuario que podrá iniciar sesión para utilizar parte del sistema. No podrá instalar software ni hardware, ni crear, modificar o borrar configuraciones de ningún tipo.

- Usuario inicial: Nombre de usuario introducido durante la instalación del sistema operativo y que tiene casi los mismos privilegios que el administrador.

### 3.1.1. Perfiles de usuario.

Cada vez que se da de alta un usuario en un ordenador y dicho usuario inicia sesión al menos una vez, el propio sistema genera para cada usuario una configuración personal y específica, sobre todo del escritorio, del panel de control y de la utilización de aplicaciones.

Los perfiles se guardan en la partición de sistema (c:\), en xp en la carpeta c:\Documents and Settings y en vista y 7 en la carpeta c:\usuarios. Dentro de esta carpeta existe por cada usuario que está dado de alta en el sistema. Dentro de esta carpeta hay una carpeta denominada default user, que es en la que se almacena por defecto el perfil de cualquier usuario nuevo. La carpeta default user se puede borrar, pero si lo hacemos es probable que los nuevos usuarios no puedan volver a iniciar sesión en modo local al no disponer de perfil de usuario disponible.

En la carpeta All user contiene la información de programas, escritorio y panel de control común para todos los usuarios del sistema, incluido el administrador.

En Windows xp, para ver y gestionar los perfiles de usuario que tenemos en nuestro equipo, desde sistema, opciones avanzadas perfiles de usuario. Los perfiles se utilizan principalmente en configuraciones en red, es decir en red.

Para modificar o dar de baja un usuario local, solo será necesario, acceder al panel de control, cuentas de usuario y modificar (la contraseña por ejemplo) o eliminar el usuario.

Al eliminar un usuario se puede, se puede elegir qué hacer con los archivos de ese usuario, si eliminar o conservar.

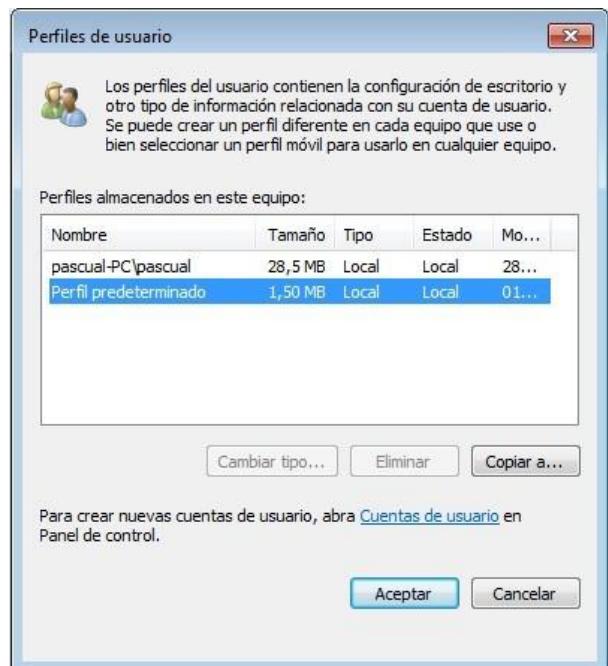
Desde el terminal se pueden crear usuarios desde el comando net user, así se puede utilizar:

net user prueba /ADD, para añadir al usuario prueba al sistema.

La ayuda para el comando:

```
C:\Windows\system32>net user /?
La sintaxis de este comando es:

NET USER
[usuario [contraseña : *] [opciones]] [/DOMAIN]
[usuario {contraseña : *} /ADD [opciones] [/DOMAIN]
[usuario [/DELETE] [/DOMAIN]
[usuario [/TIMES:{tiempos : ALL}]
```

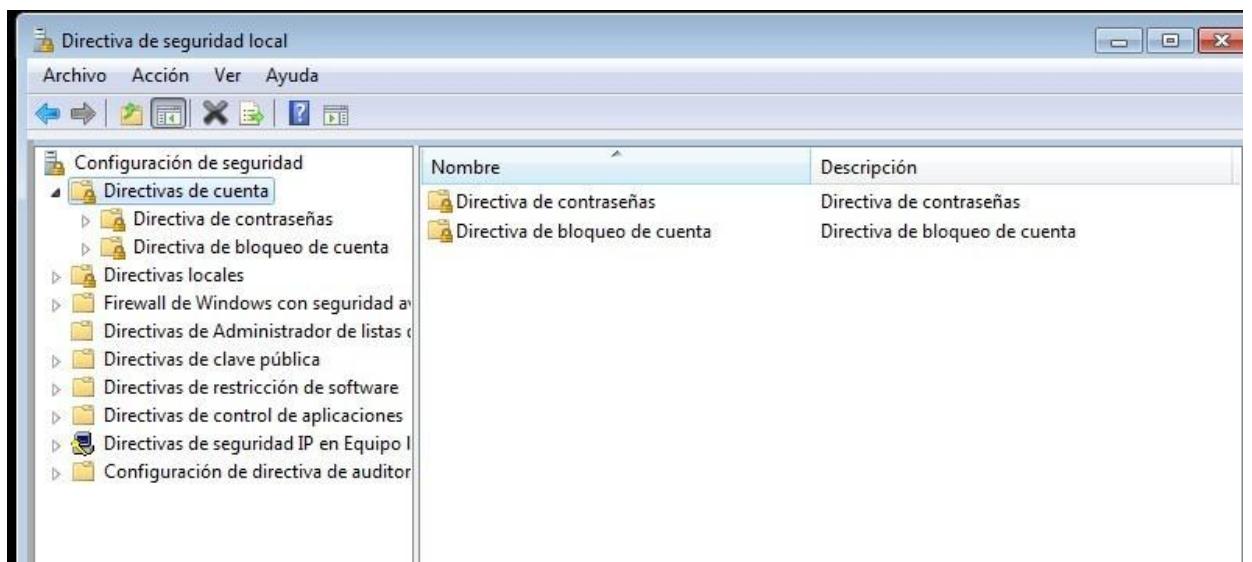


### 3.1.2. Opciones de seguridad.

En Windows 7 podemos ajustar un buen número de directivas locales. Si accedemos a la ventana de *Herramientas administrativas*, podremos pinchar sobre el ícono de *Directiva de seguridad local*. Enseguida, se abrirá una ventana donde, repartidas en diferentes apartados, encontraremos infinidad de opciones para ajustar la seguridad y comportamiento del equipo.

Así, por ejemplo, bajo *Directivas de cuenta* podremos crear una directiva acerca de la calidad de la contraseña u otra que bloquee la cuenta tras una serie de intentos fallidos de acceso. Bajo el apartado de *Directivas locales* localizamos el grueso de las opciones, hasta el punto de controlar el cambio de hora, uso de archivo de paginación o quién puede apagar la máquina.

Para ello, Inicio ⇒ Panel de control ⇒ Herramientas administrativas ⇒ Directivas de seguridad local ⇒ Directivas de cuenta.



### 3.1.3. Directiva de contraseña.

- Forzar El Historial De Contraseñas Esta configuración de seguridad determina el número de nuevas contraseñas únicas que deben asociarse a una cuenta de usuario antes de poder reutilizar una contraseña antigua. El valor debe estar comprendido entre 0 y 24 contraseñas. Esta directiva permite a los administradores mejorar la seguridad ya que garantiza que no se reutilicen continuamente contraseñas antiguas.

- La Contraseña Debe Cumplir Los Requisitos De Complejidad Esta configuración de seguridad determina si las contraseñas deben cumplir los requisitos de complejidad. Si se habilita esta directiva, las contraseñas deben cumplir los siguientes requisitos mínimos:

- No contener el nombre de cuenta del usuario o partes del nombre completo del usuario en más de dos caracteres consecutivos
- Tener una longitud mínima de seis caracteres. Incluir caracteres de tres de las siguientes categorías:
  - Mayúsculas (de la A a la Z)
  - Minúsculas (de la a a la z)

- Dígitos de base 10 (del 0 al 9)

- Caracteres no alfanuméricos (por ejemplo! \$, #, %)

Estos requisitos de complejidad se exigen al cambiar o crear contraseñas.

- Longitud mínima de la contraseña Esta configuración de seguridad determina el número mínimo de caracteres que debe contener la contraseña de una cuenta de usuario.

- Vigencia máxima y mínima de la contraseña Esta configuración de seguridad determina el período de tiempo (en días) en que puede usarse una contraseña antes de que el sistema solicite al usuario que la cambie. Puede establecer las contraseñas para que expiren tras un número de días comprendido entre 1 y 999, o puede especificar que las contraseñas no expiren nunca estableciendo el número de días en 0.

#### **3.1.4. Directiva de bloqueo de cuentas.**

- Duración del bloqueo de cuenta. Esta configuración de seguridad determina el número de minutos que una cuenta bloqueada permanece en este estado antes de desbloquearse automáticamente. El intervalo disponible oscila entre 0 y 99.999 minutos.

- Restablecer recuentos de bloqueo de cuenta tras Esta configuración de seguridad determina el número de minutos que deben transcurrir tras un intento de inicio de sesión incorrecto para que el contador de intentos de inicio de sesión incorrectos se restablezca en 0. El intervalo disponible oscila entre 1 y 99.999 minutos.

- Umbral de bloqueo de cuenta Esta configuración de seguridad determina el número de intentos de inicio de sesión incorrectos que hacen que una cuenta de usuario se bloquee. Una cuenta bloqueada no puede usarse hasta que un administrador la restablezca o hasta que expire su duración de bloqueo. Puede establecer un valor comprendido entre 0 y 999 intentos de inicio de sesión incorrectos. Si establece el valor en 0, la cuenta no se bloqueará nunca.

### **3.2. Usuarios en Linux.**

En sistemas operativos del tipo Unix, root es el nombre convencional de la cuenta de usuario que posee todos los derechos en todos los modos (mono o multi usuario). El usuario root es también llamado superusuario. Normalmente esta es la cuenta de administrador. El usuario root puede hacer muchas cosas que un usuario común no puede, tales como cambiar el dueño o permisos de archivos y enlazar a puertos de numeración pequeña. No es recomendable utilizar el usuario root para una simple sesión de uso habitual, ya que pone en riesgo el sistema al garantizar acceso privilegiado a cada programa en ejecución. Es preferible utilizar una cuenta de usuario normal y utilizar el comando su para acceder a los privilegios de root de ser necesario.

Sin embargo, Ubuntu incluye el usuario root, pero la contraseña de este usuario es generada de manera aleatoria al instalar el sistema. En su lugar, se concede el acceso administrativo a cada usuario individual. La primera cuenta de usuario que se haya creado en su sistema durante la instalación tendrá, de forma predeterminada, privilegios de administración. El resto tiene esta opción desactivada, aunque activarla es sencillo.

Cuando ejecute una aplicación que requiera privilegios de administrador, se le pedirá que escriba su contraseña de usuario. El sistema la recordará durante 15 minutos. Esta característica fue diseñada para permitir a los usuarios realizar varias tareas administrativas sin tener que introducir la contraseña cada vez. Despues volverá a ser un usuario corriente. Se previene así que aplicaciones maliciosas dañen el sistema o estropear algo accidentalmente.

En Linux existe una carpeta personal por cada usuario en /home. El usuario root tiene su propia carpeta en /root.

### 3.2.1. Usuarios en modo gráfico.

Para gestionar a los usuarios en Ubuntu 18.10, en modo gráfico, se realiza desde la



ubicación es Configuración del sistema ⇒

Detalles ⇒ Cuentas de usuario.

En la primera ventana, nos permite añadir y/o eliminar un usuario o modificar características un usuario ya creado. Entre las características que permite modificar el nombre del usuario, el tipo de cuenta (administrador o usuario de escritorio) o la contraseña.

### 3.2.2. Alta de usuarios con comandos.



Para crear un usuario existe un comando, complicado y con una gran variedad de opciones. El comando es useradd, y en principio generaría el usuario, sin carpeta personal, simplemente añade un nuevo usuario al sistema.

Alguna de las opciones del comando useradd

-u : permite especificar el UID.

-c : añade los valores a la sección de comentarios.

-d : permite especificar el directorio de trabajo.

-s : permite establecer el shell.

-m : fuerza la creación del directorio de trabajo, en el directorio estándar (/home).

-k : permite establecer un perfil al usuario normalmente /etc/skel.

Por ejemplo para añadir al usuario prueba, con un perfil de usuario y con carpeta personal, el comando seria:

```
useradd -m -k /etc/skel prueba
```

El comando useradd, requiere de muchas pruebas e intentos para su dominio. En casi todas las distribuciones de Linux existe un script (no un comando) que realiza la tarea de creación de usuarios más mucho más fácil.

Un *script* (cuya traducción literal es 'guión') o archivo de órdenes o archivo de procesamiento por lotes es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano. El uso habitual de los scripts es realizar diversas tareas como combinar componentes, interactuar con el sistema operativo o con el usuario.

El script usado en Linux para la creación de usuarios es adduser, y en Ubuntu está ubicado en /usr/sbin/. Su funcionamiento es fácil, se valida como superusuario, se teclea el script y el nombre que deseamos darle al usuario. Este script añade el usuario, crea su grupo y le asigna ese grupo, crea su directorio personal, le asigna el perfil, solicita un password para ese usuario y pide información complementaria para ese usuario.

```
profesor@PC03: ~
Archivo Editar Ver Terminal Ayuda
profesor@PC03:~$ sudo adduser prueba
Añadiendo el usuario `prueba' ...
Añadiendo el nuevo grupo `prueba' (1002) ...
Añadiendo el nuevo usuario `prueba' (1002) con grupo `prueba' ...
Creando el directorio personal `/home/prueba' ...
Copiando los ficheros desde `/etc/skel' ...
Introduzca la nueva contraseña de UNIX:
Vuelva a escribir la nueva contraseña de UNIX:
passwd: contraseña actualizada correctamente
Cambiando la información de usuario para prueba
Introduzca el nuevo valor, o presione ENTER para el predeterminado
    Nombre completo []: esto es una prueba
    Número de habitación []:
    Teléfono del trabajo []: 978601452
    Teléfono de casa []: 978604152
    Otro []:
¿Es correcta la información? [S/n] S
profesor@PC03:~$ 
```

### 3.2.3. Gestión de contraseñas.

Para cambiar la contraseña de un usuario, además del modo gráfico, también se dispone del comando passwd. Para cambiar la contraseña de un usuario, siempre que se disponga de privilegios, se escribirá el comando y el nombre de usuario que deseemos cambiar. Si no se especifica nada se cambia la contraseña del usuario actual.

passwd usuario

### 3.2.4. Modificación y baja de usuarios locales.

Para modificar por comandos opciones de usuario, se utiliza el comando usermod. La orden usermod dispone de las mismas opciones que la orden useradd, lo que permite por ejemplo, bloquear o desbloquear un usuario, cambiar el directorio personal, el numero id, cambiar el nombre de usuario, etc.

Así para cambiar el nombre del usuario prueba a Juan, el comando sería:

sudo usermod -l Juan prueba

Pero la carpeta personal seguiría siendo la original.

El comando userdel elimina a un usuario del sistema. Para eliminar su carpeta personal, es necesario el parámetro **-r**.

Así para eliminar al usuario Juan y su carpeta:

```
sudo userdel -r Juan
```

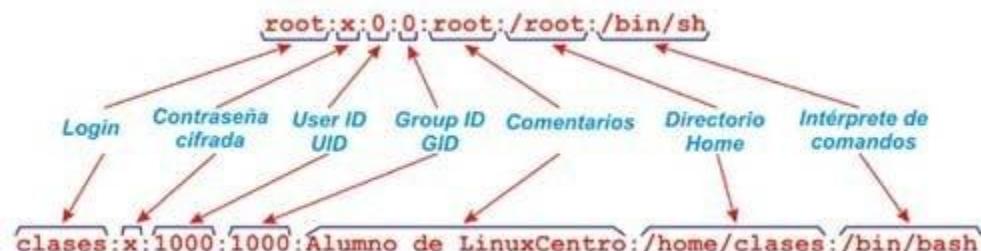
### 3.2.5. Archivos de configuración.

Archivos de configuración de usuarios: **/etc/passwd**.

El archivo passwd almacena los usuarios creados en el sistema y tiene el siguiente formato:

```
gdm:x:113:120:Gnome Display Manager:/var/lib/gdm:/bin/false
smr:x:1000:1000:smr,,,:/home/smr:/bin/bash
pepino:x:1001:1001:pepino,,,:/home/pepino:/bin/bash
smr@smr-VirtualBox:~$
```

Cada línea está separada en campos, el separador de campo son los dos puntos (**:**), y cada campo representa lo siguiente:



**Login:** el nombre del usuario. No puede haber dos nombres iguales.

**Contraseña cifrada:** si no se utiliza el fichero de shadow, las contraseñas cifradas se almacenan en este campo. Si utilizamos el fichero de shadow, todos los usuarios existentes en este fichero deben existir también en el de shadow y en este campo se pone el carácter “x”.

**User ID:** número de identificación del usuario. Es el número con el cual el sistema identifica al usuario. El 0 es el único que está reservado para el root.

**Group ID:** el número de grupo al cual pertenece el usuario. Como un usuario puede pertenecer a más de un grupo, este GID es del grupo primario.

**Comentarios:** campo reservado para introducir los comentarios que queramos sobre el usuario. Se suele utilizar para poner el nombre completo o algún tipo de identificación personal.

**Directorio home:** el directorio home del usuario es donde éste podrá guardar todos sus ficheros, generalmente se encuentran dentro del directorio /home y el nombre de cada

directorio es similar al de cada usuario.

**Intérprete de comandos:** un intérprete de comandos (shell) es un programa que se encarga de leer todo lo que escribimos en el teclado y ejecutar los programas o comandos que le indiquemos. Hay decenas de ellos, aunque el más utilizado es, sin duda, el bash (GNU Bourne-Again SHell). Si en este campo está: /bin/false ó /bin/nologin el usuario no podrá tener acceso a su shell y no podrá ejecutar comandos.

Archivo de contraseñas de usuarios: /etc/shadow.

El archivo shadow se encarga de almacenar las contraseñas cifradas del usuario, y tienen el siguiente formato:

```
gdm:*:14889:0:99999:7:::
smr:$6$DCKG6.Lf$YoYz0LLFw5zAVqqunpabY0Vcbu3xwXloRKyiAc1rsQFlkDI5Q8ngj9qRLACw0TKP
/n/8w3s7L4uh9mPvnV/5j1:14983:0:99999:7:::
pepino:$6$Ljfp4o8j$MczFdW9TFTtk.wno4GFpzGLvs9GOJAuzLbyf9qzUi.PHeNkPP7py158eY5ytx
LNxzo.NrtL.EeGtHd6v7K0V3/:15035:0:99999:7:::
```

Igual que los archivos anteriores cada línea está separada en campos el separador de campo son los dos puntos ( : ), y cada campo representa lo siguiente:



Login: debe ser el mismo nombre que se utiliza en el fichero de passwd.

Contraseña cifrada.

Días que han pasado, desde el 1 de enero de 1970, hasta que la contraseña ha sido cambiada por última vez.

Días que deben pasar hasta que la contraseña pueda ser cambiada.

Días que han de pasar hasta que la contraseña deba ser cambiada.

Días antes de caducar la contraseña en el que se avisará al usuario de que debe cambiarla.

Días que pueden pasar después de que la contraseña caduque, antes de deshabilitar la cuenta del usuario (si no se cambia la contraseña).

Días, desde el 1 de enero de 1970, desde que la cuenta está deshabilitada.

Campo reservado.

En sistemas UNIX es muy común representar las fechas a partir del número de segundos transcurridos desde el 1 de enero de 1970.

### 3.2.6. Comandos para la administración de usuarios.

Comando	Descripción	Ejemplo
<b>adduser</b>	Añade un usuario al sistema pidiendo todos los parámetros de configuración. COMANDO RECOMENDADO. Es un script	<b>adduser alumno</b>
<b>useradd</b>	Añade un usuario sin valores.	<b>useradd alumno</b>
<b>userdel</b>	Elimina un usuario. Para que borre su carpeta personal añadir <b>-r</b> . Este comando no elimina el grupo del usuario por lo que habría que hacerlo a mano.	<b>userdel alumno</b> <b>userdel -r alumno</b>
<b>usermod</b>	Modifica la información de un usuario.	<b>usermod -e 2010-10-20 alumno</b>
<b>passwd</b>	Cambia la contraseña de un usuario.	<b>passwd usuario</b>
<b>su</b>	Cambio de sesión a la del root. Necesita conocer la contraseña de root. También permite iniciar sesión como otro usuario, si se especifica.	<b>su</b> <b>su usuario</b>
<b>sudo</b>	Este programa permite que un usuario pueda ejecutar determinados comandos con privilegios de root. En este caso no es necesario que el usuario conozca la contraseña de root. Estos usuarios y los comandos permitidos para él deben de estar en el fichero <b>/etc/sudoers</b> .	<b>sudo &lt;comando&gt;</b>
<b>chown</b>	Modifica el propietario y/o grupo de un fichero o directorio	<b>chown usuario:grupo file</b>
<b>chgrp</b>	Modifica el grupo de un fichero o directorio	<b>chgrp usuario archivo</b>
<b>id</b>	Muestra la información del usuario a o del usuario actual por definido o defecto	<b>id usuario</b>
<b>who</b>	Muestra los usuarios que actualmente están dentro del sistema	<b>who</b>

### 3.2.7. Grupos de usuarios.

Un grupo de usuarios es una política utilizada generalmente por un sistema operativo para referirse a más de 1 usuario en su configuración y privilegios. En equipos locales, no se suele utilizar, pero en organizaciones con muchos usuarios, su uso es imprescindible. Los comandos para gestionar grupos son:

Comando	Descripción	Ejemplo
<b>addgroup</b>	Añade un grupo al sistema.	<b>addgroup gruposom</b>
<b>groupdel</b>	Elimina un grupo si no tiene usuarios añadidos	<b>groupdel gruposom</b>
<b>groupmod</b>	Modifica el nombre o GID de un grupo.	<b>groupmod -g 701 gruposom</b>

<b>gpasswd</b>	Permite administrar los grupos. Se puede utilizar para añadir y eliminar usuarios, señalar un administrador e	<b>gpasswd</b> <b>gruposom</b> <b>gpasswd</b>	<b>-a alumno</b> <b>-d alumno</b>
----------------	---	---	--------------------------------------

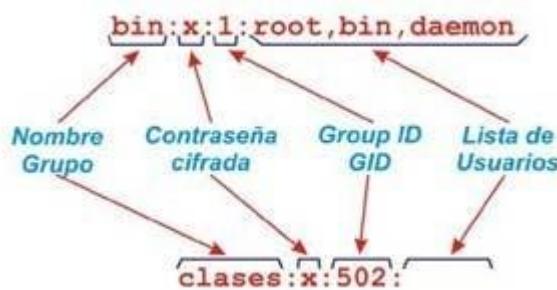
	indicar un password para el grupo.	<b>gruposom</b>
--	------------------------------------	-----------------

Archivo de configuración de grupos: **/etc/group**

El archivo group almacena la información de los grupos del sistema, y tiene el siguiente formato:

```
admin:x:119:smr
gdm:x:120:
nopasswdlogin:x:121:
smr:x:1000:
```

Al igual que el archivo anterior cada línea está separada en campos el separador de campo son los dos puntos (**:**), y cada campo representa lo siguiente:



**Nombre del grupo.** Por defecto con los comandos habituales se crea un grupo con el mismo nombre que el usuario creado, aunque pueden existir otros grupos con nombres específicos.

**Contraseña cifrada:** la contraseña de un grupo se utiliza para permitir que los usuarios de un determinado grupo se puedan cambiar a otro o para ejecutar algunos programas con permisos de otro grupo, siempre que se disponga de la contraseña.

**Group ID:** número de identificación del grupo. Es el número con el cual el sistema identifica internamente a los grupos. El 0 es el único que está reservado para el grupo del root (los administradores).

**Lista de usuarios:** los nombres de los usuarios que pertenecen al grupo, separados por comas. Aunque todos los usuarios deben pertenecer a un determinado grupo (especificado en el cuarto campo del fichero de passwd), este campo se puede utilizar para que usuarios de otros grupos también dispongan de los mismos permisos que tiene el que se está referenciando.

#### 4. Monitorización de procesos y servicios.

##### 4.1. Procesos en primer y segundo plano.

Como ya se vio en el tema anterior, de gestión de procesos, una de las principales tareas del sistema operativo es la gestión de procesos. Cuando un equipo está en funcionamiento van a existir varios procesos ejecutándose en el equipo y el sistema operativo debe de controlarlos a todos, intentando dar servicio a todos los procesos y realizar un control eficiente de los recursos de la máquina.

Un proceso es un programa que está en ejecución, y por lo tanto necesita de unos recursos para que mientras este ejecutándose en el sistema. El sistema operativo será el encargado de crear, destruir, parar y reanudar los procesos y ofrecer mecanismos para que se comuniquen.

Cualquier proceso puede estar en primer o segundo plano. Lo único a tener en cuenta es que solo un proceso estará en primer plano al mismo tiempo y es con el que estemos trabajando e interactuando en ese momento. Un proceso que este en segundo plano no recibirá ninguna señal de parte nuestra, es decir que no nos podemos comunicar con él a través, por ejemplo, del teclado.

Así en Linux si ejecutamos el programa (como puede ser gedit o xeyes), el terminal se quedara en segundo plano, y no podemos seguir trabajando en el terminal hasta que se cierre el programa en cuestión. Para ejecutar un programa en segundo plano (foreground), existen varias soluciones, pero la más sencilla es ejecutar el programa con un ampersand (&) al final.

Así para ejecutar el programa xeyes en segundo plano, se debe realizar con el símbolo &:

```
xeyes &
```

Esto también sirve para comandos que requieren una gran cantidad de tiempo, así para buscar los archivos de configuración del sistema, dejarlos en un archivo y seguir utilizando la terminal, el comando quedaría así.

```
sudo find / | grep .conf > /home/smr/ficherosConf.txt &
```

##### 4.2. Gestión de procesos en Windows.

Para ver los procesos, en Windows tendremos que ejecutar el administrador de tareas, el administrador de tareas de Windows se puede ejecutar desde el menú de inicio ⇒ ejecutar ⇒ taskmgr o desde la unión de las teclas Ctrl+Alt+Supr, desde la versión de Windows aparece directamente, a que pulsar en administrador de tareas.

Aparecen aquí varias pestañas, las importantes para nosotros serán aplicaciones, procesos y rendimiento.

En la pestaña de aplicaciones aparecen las aplicaciones que actualmente están corriendo en el sistema. Desde esta pestaña, botón derecho sobre la aplicación podemos ir al proceso principal de esa aplicación.

En la segunda pestaña aparecen los procesos, una serie de características de los procesos, por defecto, el nombre del ejecutable, el propietario, los ciclos de CPU y el uso de memoria. Desde el menú Ver seleccionar columnas, pueden verse muchos más datos de los procesos como por ejemplo el PID del proceso.

Desde consola el comando tasklist muestra los procesos que están en el sistema. Así si deseamos comprobar si el proceso winword está en ejecución, teclearíamos:

tasklist | find "WINWORD"

TASKKILL es un comando que complementa al anterior. Permite detener tareas o procesos usando el PID (número de identidad del proceso) o el nombre.

taskkill /IM Excel.exe (usando el nombre del proceso)

o taskkill /PID 4325 (usando el PID del proceso)

### **Los servicios en Windows.**

Los servicios no son nada más ni nada menos que programas o aplicaciones cargadas por el propio sistema operativo. Estas aplicaciones tienen la particularidad que se encuentran corriendo en segundo plano (Background).

Por defecto, con la instalación, se instalan y ejecutan una cierta cantidad de servicios. De más está decir, que dependiendo de nuestras necesidades, podemos necesitarlos a todos o no.

Como sabemos, mientras más aplicaciones tengamos ejecutándose consumimos más recursos, por lo tanto, vamos a tratar de deshabilitar lo que no utilizamos.

### **Activación y desactivación de servicios.**

Para visualizar los servicios, o para cambiar algunas de sus opciones y/o estados, debemos abrir la consola de Microsoft.

Esto lo podemos hacer de 2 maneras:

- Inicio / Panel de control / Rendimiento y mantenimiento / Herramientas Administrativas / Servicios

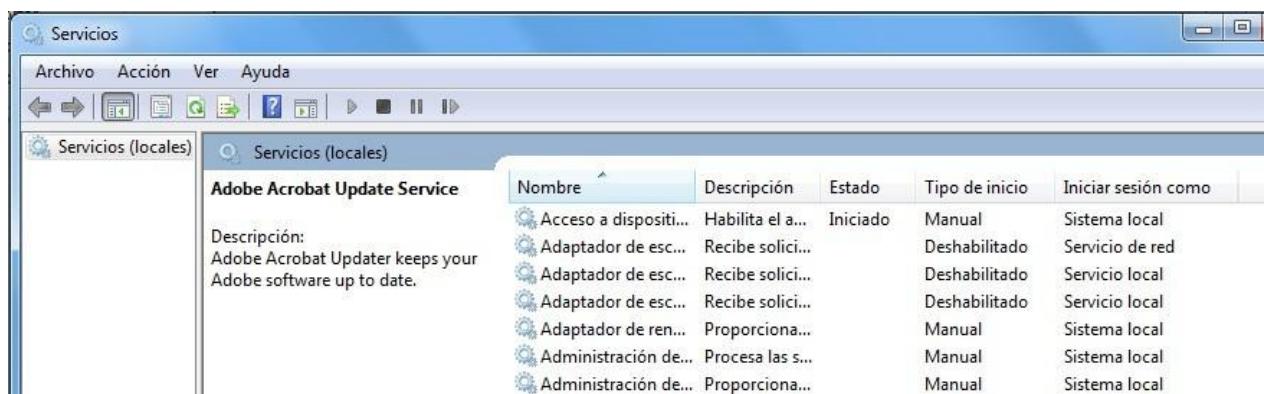
- Inicio / Panel de control / Herramientas Administrativas / Servicios dependiendo de cómo tengamos configurada la vista de Panel de Control.

Estos pasos pueden ser reemplazados por lo siguiente:

Nombre de im...	Nombre ...	CPU	Memoria ...	Descripción
firefox.exe *32	alumno	00	227.924 KB	Firefox
explorer.exe	alumno	00	45.892 KB	Explorador...
WINWORD.E...	alumno	00	20.480 KB	Microsoft ...
soffice.bin *32	alumno	00	12.856 KB	OpenOffi...
FlashPlayerPl...	alumno	00	11.692 KB	Adobe Fla...
dwm.exe	alumno	02	10.408 KB	Administr...
avastui.exe *32	alumno	00	6.836 KB	avast! An...
plugin-contain...	alumno	00	5.188 KB	Plugin Co...
FlashPlayerPl...	alumno	00	3.888 KB	Adobe Fla...
taskhost.exe	alumno	00	3.860 KB	Proceso d...
splwow64.exe	alumno	00	3.672 KB	Print driv...
MSOSYNC.EX...	alumno	00	3.488 KB	Microsoft ...
SnippingTool....	alumno	03	3.208 KB	Recortes
wisptis.exe	alumno	00	2.412 KB	Componen...
taskmgr.exe	alumno	01	2.292 KB	Administr...

Buttons at the bottom: Mostrar procesos de todos los usuarios, Finalizar proceso.

Metrics at the bottom: Procesos: 60, Uso de CPU: 7%, Memoria física: 39%.



Nos dirigimos a Inicio, Ejecutar, escribimos services.msc y presionamos Enter.

Para iniciar o detener un servicio, una vez en la consola, nos posicionamos arriba del servicio que queremos iniciar o detener y haciendo click con el botón derecho vamos a ver las acciones correspondientes.

Otras maneras de iniciar o detener un servicio, es desde la consola, podemos hacerlo utilizando los comandos NET START y NET STOP. Para iniciar y detener un servicio, respectivamente.

El modo de uso es: NET START/STOP NombreDelServicio

Dónde NombreDelServicio es el nombre del servicio completo (entre "" comillas si contiene espacios) o el nombre abreviado.

Ejemplo:

net start BITS

net start "Background Intelligent Transfer Service"

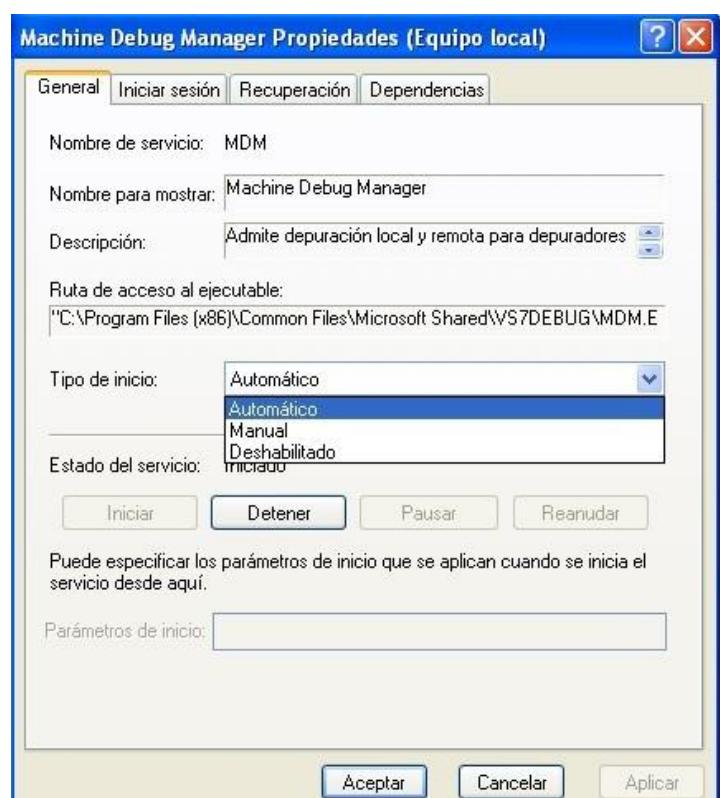
net stop "Automatic

Updates" net start

messenger

Los servicios también se pueden iniciar desde el registro de Windows.

Los servicios pueden encontrarse en dos estados posibles. Pueden estar iniciados, es decir, se encuentra ejecutándose/corriendo o puede estar detenido. Y tenemos tres opciones posibles de inicio:



- **Automático:** Se inician junto con el sistema operativo.
- **Manual:** Podemos iniciarlos manualmente cuando lo deseemos u otro servicio puede hacerlo automáticamente. En un principio estaría detenido.

- **Deshabilitado:** No se puede iniciar manualmente ni otro servicio puede hacerlo.

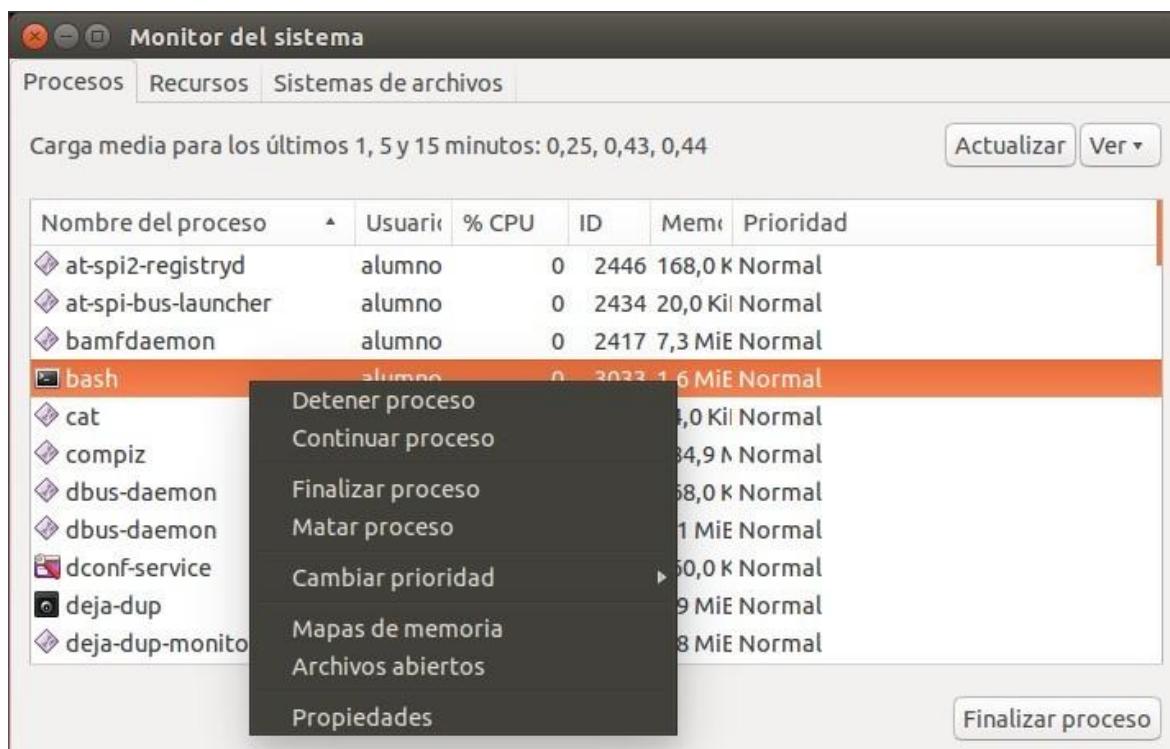
Para cambiar la manera en que se inicia un servicio, debemos dirigirnos a la consola. Una vez ahí elegimos el servicio con el cual vamos a trabajar, hacemos click con el botón derecho del mouse y elegimos propiedades.

A pesar de que los cambios son reversibles, la recomendación es cambiar la configuración de algún servicio, con precaución, ya que podemos causar efectos no deseados sobre nuestro sistema operativo. Por lo tanto hay que estar seguros antes de realizar alguna modificación.

Como recomendación, resta decir que es preferible ir modificándolos de a uno y una vez que vemos que nuestra computadora sigue funcionando de manera adecuada, procedemos con los restantes.

### **Visualización de procesos en Linux, desde el modo gráfico.**

Para gestionar y administrar los procesos en modo grafico en Ubuntu 18.10 es necesario ir al monitor del sistema, y en la pestaña de procesos.



De forma similar a como lo hacíamos en Windows, seleccionando el proceso deseado, y pulsando el botón derecho del ratón, podremos realizar las siguientes acciones:

- Detener un proceso, el proceso se detiene pero sin que termine.
- Continuar el proceso, si es que lo hemos detenido.
- Finalizar el proceso, para que termine naturalmente por sí mismo, cerrándose archivos abiertos y resto de recursos utilizados.
- Matar un proceso para terminarlo incondicionalmente, sobre todo si está bloqueado.
- Cambiar la prioridad. En el control numérico que aparece podemos indicar la prioridad del proceso, siendo 20 la mayor y -20 la menor.

Además nos permite mostrar características del proceso, como ubicación en la memoria, archivos abiertos o tamaño en el espacio de memoria.

#### 4.3. Comandos de Linux para la gestión de procesos.

También existen comandos para gestionar los procesos. Los principales

son: Comando top:

El comando top muestra a tiempo real un listado de los procesos que se están ejecutando en el sistema, especificando además el % de CPU y Memoria que están utilizando, sus IDs, usuarios que lo están ejecutando, etc. Todo esto y mucho más lo hace una de las herramientas imprescindibles para la administración de cualquier tipo de sistema Unix y derivado.

La salida por pantalla de Top, puede dividirse en dos partes, la “cabecera” muestra entre otras cosas, el Uptime del servidor, nº de usuarios conectados y load average. En la siguiente línea podemos ver el nº de procesos ejecutándose en el sistema, así como el uso de disco, memoria y cpus.

Posteriormente podemos observar un listado de procesos, que pueden ser ordenados por uso de CPU o memoria, lo que es una excelente ayuda para detectar procesos que consumen excesivos recursos en el servidor. Este listado, muestra varios detalles de cada uno de los procesos, como pueden ser el PID de proceso, usuario que lo ejecuta, %CPU y memoria que consume, comando que está ejecutando o tiempo de ejecución del proceso entre otros.

```

smr@smr-VirtualBox: ~
Archivo Editar Ver Buscar Terminal Ayuda
top - 16:11:48 up 36 min, 2 users, load average: 1.14, 1.12, 1.14
Tasks: 129 total, 2 running, 126 sleeping, 1 stopped, 0 zombie
Cpu(s): 4.6%us, 41.4%sy, 0.0%ni, 53.6%id, 0.0%wa, 0.0%hi, 0.3%si, 0.0%st
Mem: 476132k total, 408856k used, 67276k free, 58164k buffers
Swap: 999420k total, 9068k used, 990352k free, 190860k cached

PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+   COMMAND
1951 smr      20   0  91596  13m  10m S 23.4  2.9  0:02.06 gnome-terminal
  807 root     20   0  53632  16m  5564 R 22.5  3.5  5:06.67 Xorg
 1975 root     20   0   2620  1112   840 R  1.0  0.2  0:00.17 top
    3 root     20   0     0     0     0 S  0.3  0.0  0:01.31 ksoftirqd/0
1281 smr      20   0  98.5m  10m  7736 S  0.3  2.2  0:11.45 gnome-settings-
1301 smr      20   0  146m  12m  9408 S  0.3  2.6  0:15.93 metacity
1384 smr      20   0 19144  2132  1744 S  0.3  0.4  0:02.59 gvfs-afc-volume
    1 root     20   0  2864  1532  1168 S  0.0  0.3  0:01.62 init
    2 root     20   0     0     0     0 S  0.0  0.0  0:00.04 kthreadd
    4 root     RT   0     0     0     0 S  0.0  0.0  0:00.00 migration/0
    5 root     RT   0     0     0     0 S  0.0  0.0  0:00.00 watchdog/0
    6 root     20   0     0     0     0 S  0.0  0.0  0:01.47 events/0
    7 root     20   0     0     0     0 S  0.0  0.0  0:00.00 cpuset
    8 root     20   0     0     0     0 S  0.0  0.0  0:00.00 khelper
    9 root     20   0     0     0     0 S  0.0  0.0  0:00.00 netns
   10 root     20   0     0     0     0 S  0.0  0.0  0:00.00 async/mgr
   11 root     20   0     0     0     0 S  0.0  0.0  0:00.00 pm

```

Para salir de este comando q (quit).

Comando ps:

El comando ps muestra por pantalla un listado de los procesos que se están ejecutando en el sistema.

Si no añadimos ningún parámetro, ps mostrará los procesos del usuario con el que estamos logueados. Por otra parte, los parámetros más básicos a conocer son los siguientes:

**-A** Lista los procesos de todos los usuarios.

**-u** Lista información del proceso como por ejemplo el usuario que lo está corriendo, la utilización de CPU y memoria, etc.

**-x** Lista procesos de todas las terminales y usuarios.

**-e** Idéntico a -A

**-f** Muestra columnas, con más información.

Lo más común es listarlos con los siguientes parámetros:

sudo ps -aux

Comando kill y killall:

**kill**[opción(es)]proceso-ID

A veces, desafortunadamente, nos encontramos con programas que no se pueden cerrar de forma normal. Con el comando kill, se puede eliminar todos los procesos muertos con ayuda del ID del proceso (PID). Para conocer el PID del proceso utilizaremos los comandos top y ps.

Para ello envía una señal llamada TERM que ordena al programa cerrarse a sí mismo. Si esto tampoco sirve tenemos aún otro parámetro de gran ayuda:

**-9**

Envía en lugar de una señal tipo TERM una señal tipo KILL, lo que ocasiona que el proceso sea liquidado por el sistema operativo, consiguiendo en casi todos los casos que el proceso especificado se termine.

**Killall** [opción(es)] nombreproceso

Este comando funciona de forma equivalente a kill, sólo que para este comando basta con dar el nombre del proceso - y no su ID - para matar todos los procesos que tengan ese nombre.

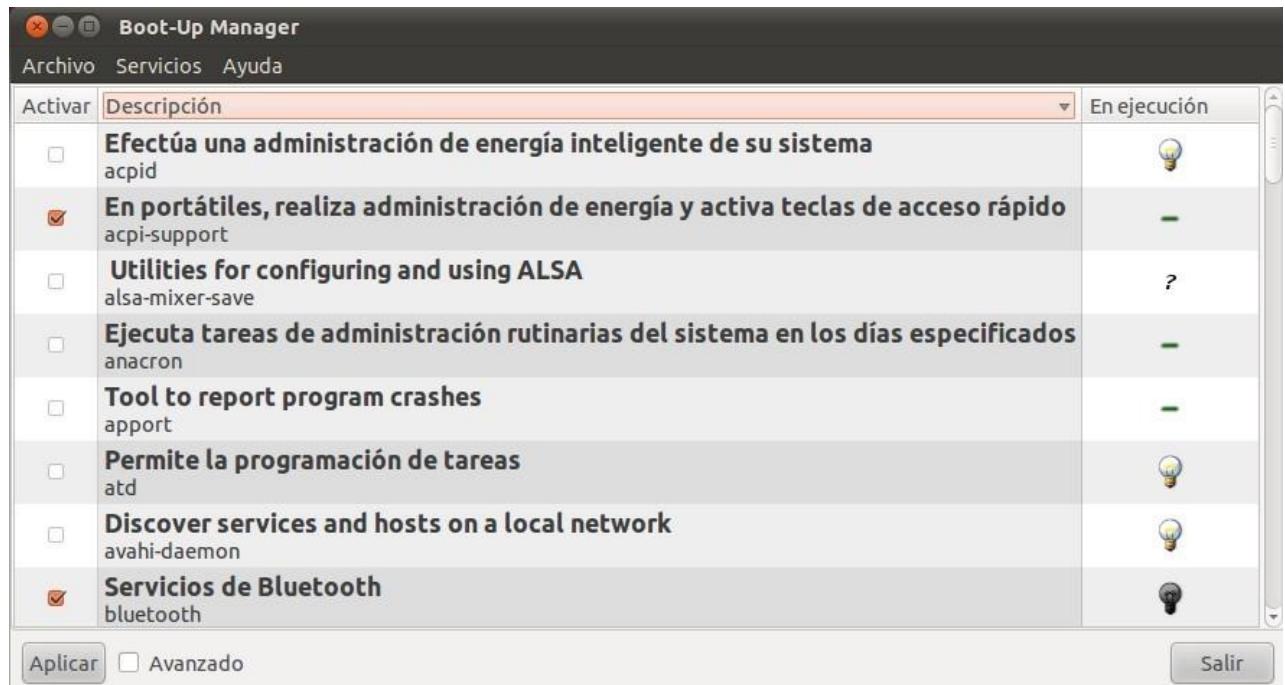
#### 4.4. Iniciar y detener servicios en Linux.

En Linux también existen procesos en segundo plano, lo que en Windows se llaman

servicios, en Linux se llaman demonios (daemon).

Para visualizar los servicios desde el entorno grafico es necesario instalar un software

específico. Para visualizarlo habría que instalar alguna herramienta como puede ser bum (boot-up manager).



Esta sencilla aplicación nos muestra un listado con todos los servicios del sistema y su estado de ejecución, pero lo más importante es que podemos activarlos/desactivarlos con un simple clic en el checkbox que hay a la izquierda de cada servicio.

Los servicios, también llamados demonios en la terminología Unix, son programas que se ejecutan en segundo plano para ofrecer una función concreta de forma independiente del usuario que pueda iniciar sesión en el sistema. Existen servicios que monitorizan el sistema, que comprueban si existen actualizaciones pendientes de instalar, que controlan el reloj del sistema y lo actualizan a través de la red cuando sea necesario, que gestionan la conexión de red y avisarán cuando algo no va bien, etc.

Cuando se instala un programa que lleva asociado algún servicio (como el servidor de impresión CUPS o el servidor web Apache), el proceso de instalación guarda en el directorio /etc/init.d el script necesario para controlarlo. Esto ocurre también con los servicios propios del sistema operativo. En resumidas cuentas, lo que esto significa es que en la carpeta /etc/init.d se encuentran todos los scripts que se utilizan para administrar los servicios que pueden ejecutarse en Ubuntu 14.04 LTS. El número de archivos almacenados dependerá de las funciones para las que esté configurado el sistema y de los dispositivos hardware instalado.

Estos scripts se programan de modo que reconozcan los siguientes argumentos:

- start: Permite iniciar un servicio de forma manual. Por ejemplo, para iniciar el servicio vboxadd escribiremos la siguiente orden:

```
sudo /etc/init.d/cron start
```

- stop: Permite detener un servicio que se está ejecutando en estos momentos. Por

ejemplo, para detener el servicio anterior, escribiremos lo siguiente:

```
sudo /etc/init.d/cron stop
```

- restart: Equivale a realizar las dos operaciones anteriores. Primero detiene el servicio y después lo inicia. Para reiniciar el servicio anterior, escribimos esto:

```
sudo /etc/init.d/cron restart
```

- status: Nos permite consultar el estado en el que se encuentra un servicio. Para comprobarlo, sólo tenemos que escribir lo siguiente:

```
sudo /etc/init.d/cron status
```

Desde la versión 9.10 de Ubuntu, comenzó a implantarse Upstart como sustituto del servicio init, que es quien se encarga de iniciar los servicios durante el inicio del sistema, detenerlos cuando apagamos y controlar su funcionamiento mientras están activos. La ventaja de Upstart es que controla los servicios atendiendo a sus eventos, lo que le permite aumentar su rendimiento. El inconveniente de todo esto es que, dentro de /etc/init.d, además de los scripts tradicionales podemos encontrar los Upstart jobs (o tareas Upstart, en español). Así, si intentamos aplicar con algunos servicios lo que acabamos de aprender, podemos encontrarnos con un error como el de la imagen siguiente:

Desde la versión 9.10 de *Ubuntu*, comenzó a implantarse *Upstart* como sustituto del servicio *init*, que es quien se encarga de iniciar los servicios durante el inicio del sistema, detenerlos cuando apagamos y controlar su funcionamiento mientras están activos. Para arrancar, parar, reiniciar o ver su estado se utilizaba la siguiente orden:

```
sudo stop cron
```

Cuando convivieron ambos sistemas se utilizó el comando service. Este comando permite gestionar de forma indistinta tanto scripts tradicionales como Upstart jobs. Además, cuando usamos scripts, evitamos la necesidad de escribir la ruta completa. La forma de realizar las acciones anteriores utilizando el comando service sería la siguiente:

- service ... start: Permite iniciar la ejecución de un servicio. Por ejemplo, para iniciar el servicio cron, ejecutaremos lo siguiente:

```
sudo service cron start
```

- service ... stop: Detiene un servicio que se está ejecutando. Por ejemplo, para detener el servicio anterior, escribiríamos esto:

```
sudo service cron stop
```

- service ... restart: Equivale a realizar las dos operaciones anteriores. Primero detiene el servicio y después lo inicia. Para reiniciar el servicio anterior, escribimos esto:

```
sudo service cron restart
```

- service ... status: Nos permite consultar el estado en el que se encuentra un servicio. Para comprobarlo, sólo tenemos que escribir lo siguiente:

```
sudo service cron status
```

En este caso también podemos comprobar el estado de todos los servicios utilizando el

siguiente comando:

```
sudo service --status-all
```

Desde Ubuntu 15.04 Vivid Vervet se integró en Ubuntu el gestor de demonios Systemd como reemplazo a Upstart. Systemd es un sistema y administrador de servicios para Linux, compatible con scripts de inicio (init) SysV y LSB. systemd proporciona capacidades de paralelización agresivas, utiliza socket y activación D-Bus para iniciar los servicios, ofrece la puesta en marcha de demonios bajo demanda, realiza el seguimiento de procesos utilizando Linux cgroups, soporta copia instantánea de volumen y la restauración de estado del sistema, mantiene puntos de montaje y automontaje e implementa un elaborado servicio lógico de control transaccional basado en la dependencia. Puede funcionar como un reemplazo para sysvinit.

Para iniciar, parar, reiniciar o ver el estado, se utiliza:

```
sudo systemctl start cron
```

Permite un servicio para iniciarse en el arranque:

```
sudo systemctl enable foo.service
```

Deshabilita un servicio para que no se inicie durante el arranque:

```
sudo systemctl disable foo.service
```

Comprueba si un servicio ya está habilitado o no:

```
sudo systemctl is-enabled foo.service; echo $?
```

#### **4.5. Niveles de ejecución o runlevel en Ubuntu.**

Durante el inicio del sistema se procede al arranque de los servicios. Los servicios que se arrancan dependen del runlevels. El runlevel (del inglés, nivel de ejecución) es cada uno de los estados en que se encuentra el sistema. En el sistema sysvinit, existen 7 niveles de ejecución en total:

- Nivel de ejecución 0: Apagado.
- Nivel de ejecución 1: Monousuario (sólo usuario root; no es necesaria la contraseña). Se suele usar para analizar y reparar problemas.
- Nivel de ejecución 2: Multiusuario sin soporte de red.
- Nivel de ejecución 3: Multiusuario con soporte de red.
- Nivel de ejecución 4: Como el runlevel 3, pero no se suele usar
- Nivel de ejecución 5: Multiusuario en modo gráfico (X Windows).
- Nivel de ejecución 6: Reinicio.

Este sistema de niveles de ejecución lo proporciona el sistema de arranque por defecto de las distribuciones GNU/Linux (init).

En las carpetas **/etc/rc[0-6,S].d** están los enlaces simbólicos a los script (que se

encuentran en /etc/init.d) que se deben ejecutar según el nivel de ejecución.

Para controlar el orden en que se van a ejecutar los scripts debemos controlar los nombres. Todos los enlaces se han de llamar SNN o KNN y primero se ejecutan los K en orden numérico y luego los S también en orden numérico.

Existe una utilidad para línea de comandos que permite cambiar de un nivel de ejecución a otro. Esta es la herramienta init (telinit). Para cambiar de nivel de ejecución sólo hay que ejecutar init seguido del número del runlevel. Por ejemplo:

**init 0:** Cambia al runlevel 0 (se apaga el sistema, equivalente al comando halt).

**init 2:** Cambia al runlevel 2.

**init 6:** Cambia al runlevel 6 (reinicia el sistema, equivalente al comando reboot).

Para reemplazar el sistema de inicio (init) heredado de los sistemas operativos estilo UNIX System V y Berkeley Software Distribution (BSD), se desarrolló systemd. Utiliza targets («objetivos») que sirven a un propósito similar a los runlevels («niveles de ejecución»), pero que tienen un comportamiento un poco diferente. Cada target se nomina, en lugar de numerarse, y está destinado a servir a un propósito específico con la posibilidad de realizar más de una acción al mismo tiempo. Algunos targets son activados heredando todos los servicios de otro target e implementando servicios adicionales. Como hay targets de systemd que imitan los runlevels de SystemV init, es, por tanto, posible pasar de un target a otro utilizando la orden telinit RUNLEVEL.

La orden systemctl se utiliza, en lugar de runlevel.

Conocer los targets presentes

```
# systemctl list-units --type=target
```

Para conocer la runlevel actual:

```
# systemctl get-default
```

Crear un target personalizado

Los niveles de ejecución («runlevels») son asignados a un fin específico de la instalación; 0, 1, 3, 5, y 6; tienen una correlación de 1:1 con un específico target de systemd. Desafortunadamente, no hay una buena manera de hacer lo mismo para los niveles de ejecución definidos por el usuario como son el 2 y el 4. Si se hace uso de estos últimos, se sugiere dar un nuevo nombre al target de systemd como /etc/systemd/system/suTarget que tome como base uno de los runlevels existentes, y cree un directorio

/etc/systemd/system/suTarget.wants, y haga un enlace a los servicios adicionales de /usr/lib/systemd/system/ que desea habilitar.

### Tabla de targets

Runlevel de SysV	Target de systemd	Notas
0	runlevel0.target, poweroff.target	Detiene el sistema.
1, s, single	runlevel1.target, rescue.target	Modalidad de usuario único.
2, 4	runlevel2.target, runlevel4.target, multi-user.target	Definidos por el usuario. Preconfigurados a 3.
3	runlevel3.target, multi-user.target	Multiusuario, no gráfica. Los usuarios, por lo general, pueden acceder a través de múltiples consolas o a través de la red.
5	runlevel5.target, graphical.target	Multiusuario, gráfica. Por lo general, tiene todos los servicios del nivel de ejecución 3, además de un inicio de sesión gráfica.
6	runlevel6.target, reboot.target	Reinicia el sistema.
emergency	emergency.target	Consola de emergencia.

### Cambiar el target vigente

En systemd los targets quedan expuestos a través de «target units». Se pueden cambiar de esta manera:

```
# systemctl isolate/set-default modo
```

Con isolate se realiza una vez y con set-default se cambia el modo por defecto.

```
sudo systemctl isolate graphical.target => cambia una vez a modo
```

gráfico.

```
sudo systemctl set-default multi-user.target => cambia el modo por perfecto.
```

Esto solo cambiará el target actual, y no tendrá ningún efecto sobre el siguiente arranque. Esto es equivalente a las órdenes init 3 o init 5 en Sysvinit.

## 5. Conexión de la impresora.

### 5.1. Conexión física.

En modo local, hoy en día la conexión de la impresora con el equipo se realiza a través de los puertos USB, ya son obsoletos el puerto paralelo con su conector DB25, y los conectores Centronics de las impresoras.

La conexión al ordenador se realiza mediante un conector USB tipo A y la impresora suele ser tipo B.

Muchas impresoras además del conector USB tipo B, tiene también de una conector RJ-45 hembra para conectarse a la red, y a través de esta poder comunicarse con el resto de equipos de la red, y así imprimir.

Si con las conexiones existe algún problema, se debe de consultar el manual que proporciona el fabricante con la impresora.

### 5.2. Conexión lógica.

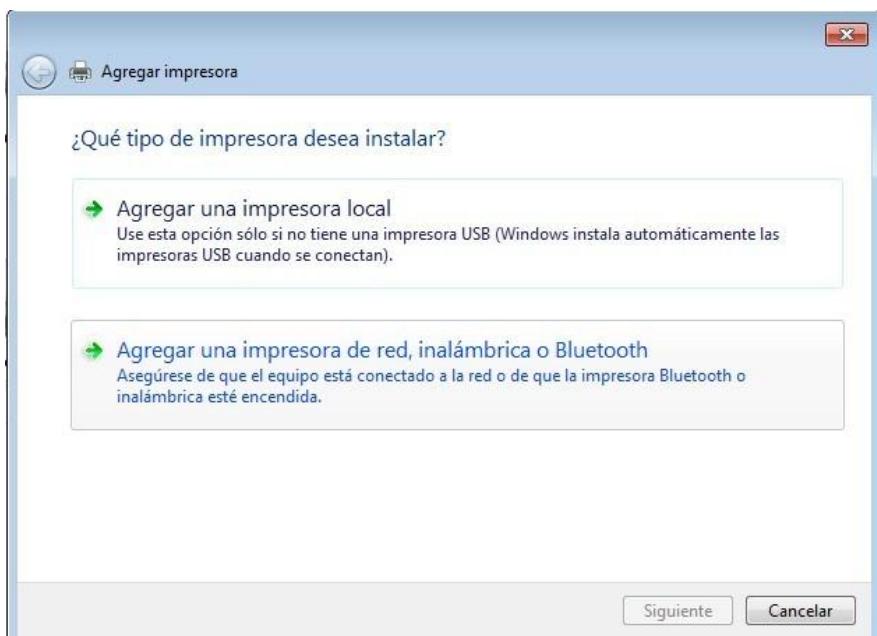
Una vez conectado la impresora, mediante el conector adecuado, será necesario instalar el software de la impresora, denominado normalmente driver. Un driver es un programa que traduce las instrucciones que da el sistema operativo, en otras que entiende la impresora.

El driver suele facilitarlo el fabricante, pero para algún sistema operativo, y no para todos los sistemas operativos.

#### Conección en Windows.

Si conectamos una impresora, el sistema operativo la detectará automáticamente (es Plug and Play), e iniciará el asistente para su instalación. Si se desea instalar sin conectar, se puede instalar el software de la impresora o desde panel de control e impresora y faxes en Windows xp y Dispositivos e impresoras en Windows 7, y desde ahí, agregar impresora.

Si no disponemos de los drivers y conectamos la impresora USB sin haber instalado el software ocurrirá lo mismo que cuando agregamos hardware, y es que aparecerá el asistente para indicarnos que le suministremos los drivers.



La diferencia entre instalar el software completo del fabricante a instalar los drivers de nuestra impresora, es que el software del fabricante incorpora utilidades de configuración, limpieza de cabezales, alineación de papel, etc. Si solo instalamos los drivers, la impresora funcionará, pero no podremos hacer prácticamente nada que no sea lo que por defecto permite el propio sistema operativo.

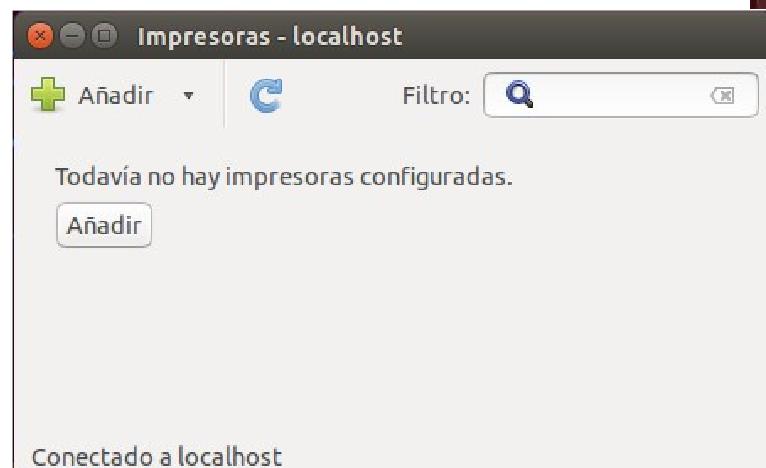
Es fácil que al acabar la instalación, nos coloque la impresora como predeterminada. Si deseamos cambiarla, accederemos al panel de las impresoras, escogeremos las que

deseemos como predeterminada y con el botón derecho, establecer como impresora predeterminada.

### Conexión en Linux.

Linux dispone de una gran cantidad de drivers para las impresoras más comerciales, pero también es cierto que, a veces, los fabricantes no suministran este tipo de drivers de forma libre.

Existen varias formas de instalar, una de ellas es desde



el dashboard de Ubuntu buscar impresoras.

Pulsando en añadir, y el asistente buscara entonces las impresoras conectadas al equipo y las mostrara en la lista.

Si la impresora está conectada, parecerá en la lista la impresora y una lista de impresión por defecto.

También se puede

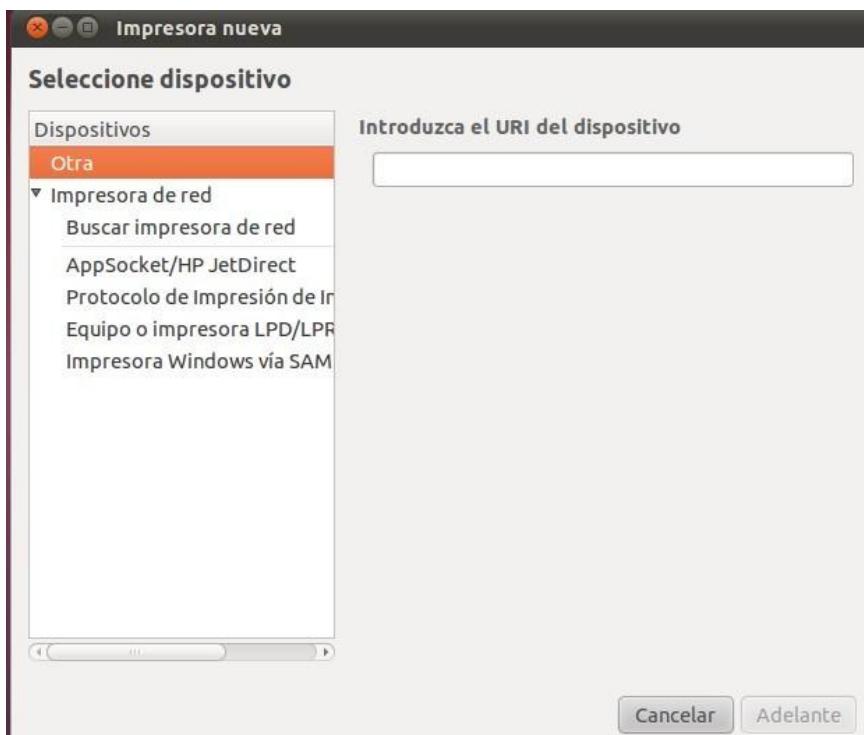
seleccionar las impresoras en red, donde aparecen las siguientes opciones:

- AppSocket/HP JetDirect, deberá indicar la dirección IP de la impresora.

- Protocolo de impresión de Internet IPP. Usa el protocolo IPP para comunicarse con la impresora.

- Equipo o impresora LPD/LPR: es un protocolo de red para la presentación de trabajos de impresión a una impresora remota.

- Windows Printer vía SAMBA, deberá indicar el servidor samba y si es preciso



autentificarse para  
utilizarla.

Si detecta la impresora en local, los pasos son los siguientes:

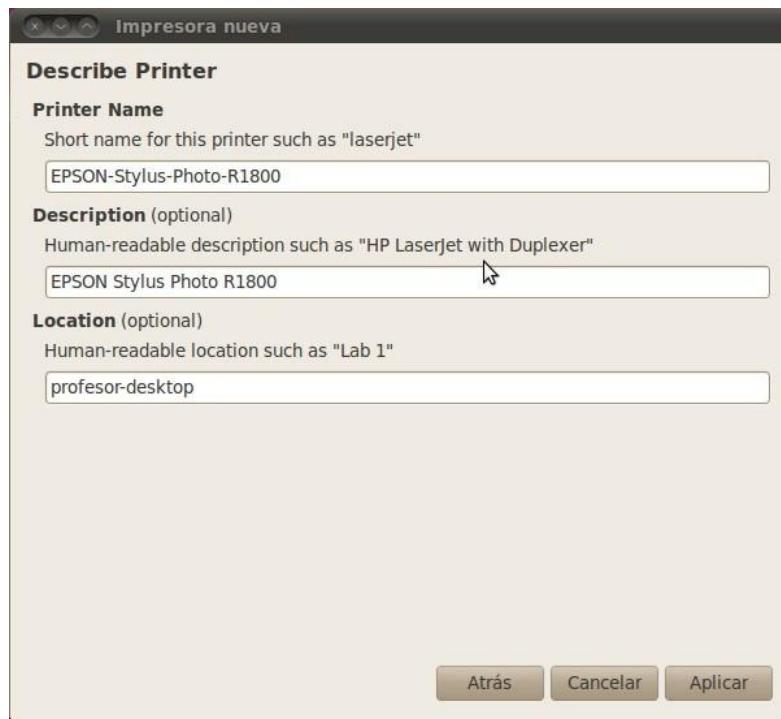
- Agregar impresora:



- Buscando controladores



- Y seleccionamos las opciones de la impresora, como el nombre que se la asigna, la descripción y el nombre del equipo en la que está instalada.



En Linux, existe un gestor de impresión que suele estar instalado en todas las distribuciones Linux, que permite la gestión de la impresión a través de un navegador Web. Este se llama CUPS (Common Unix Printing System).

CUPS es un sistema de impresión común para Unix, desarrollado para promover una solución de impresión estándar para todos los sistemas Unix y proporciona las tareas básicas de gestión de impresión y de colas de impresión para una gran cantidad de modelos de impresoras. Está basado en el Internet Printing Protocol (IPP), que es un protocolo de red que proporciona un conjunto estándar de servicios de impresión en red.

Para utilizar CUPS basta con imprimir en nuestro navegador <http://localhost:631>, y aparece una ventana como esta, donde se puede seleccionar múltiples opciones de impresión.

The screenshot shows a web browser window with the URL <http://localhost:631/>. The page title is "Inicio - CUPS 1.4.4". The main content area is titled "CUPS 1.4.4" and contains a brief introduction: "CUPS es el sistema de impresión de código abierto basado en estándares desarrollado por Apple Inc. para Mac OS® X y otros sistemas operativos tipo UNIX®." Below this, there are three columns of links:

CUPS para usuarios	CUPS para administradores	CUPS para desarrolladores
<a href="#">Descripción de CUPS</a>	<a href="#">Añadiendo impresoras y clases</a>	<a href="#">Introducción a la programación de CUPS</a>
<a href="#">Impresión desde la línea de comandos y opciones</a>	<a href="#">Gestionando políticas de funcionamiento</a>	<a href="#">La API de CUPS</a>
<a href="#">Qué hay de nuevo en CUPS 1.4</a>	<a href="#">Contabilidad básica de impresora</a>	<a href="#">Programación de filtros y programas de conexión</a>
<a href="#">Foro de usuarios</a>	<a href="#">Seguridad del servidor</a>	<a href="#">Las APIs HTTP e IPP</a>
	<a href="#">Usando autenticación Kerberos</a>	<a href="#">La API PPD</a>
	<a href="#">Usando impresoras de red</a>	<a href="#">La API Raster</a>
	<a href="#">Referencia de cupsd.conf</a>	<a href="#">Referencia del archivo de información del compilador de controladores PPD</a>
	<a href="#">Encontrar controladores de impresora</a>	<a href="#">Foro de desarrollo</a>

At the bottom of the page, a small note states: "CUPS y el logo de CUPS son marcas registradas de Apple Inc. Los derechos de copia de CUPS 2007-2010 son de Apple Inc. Todos los derechos reservados."

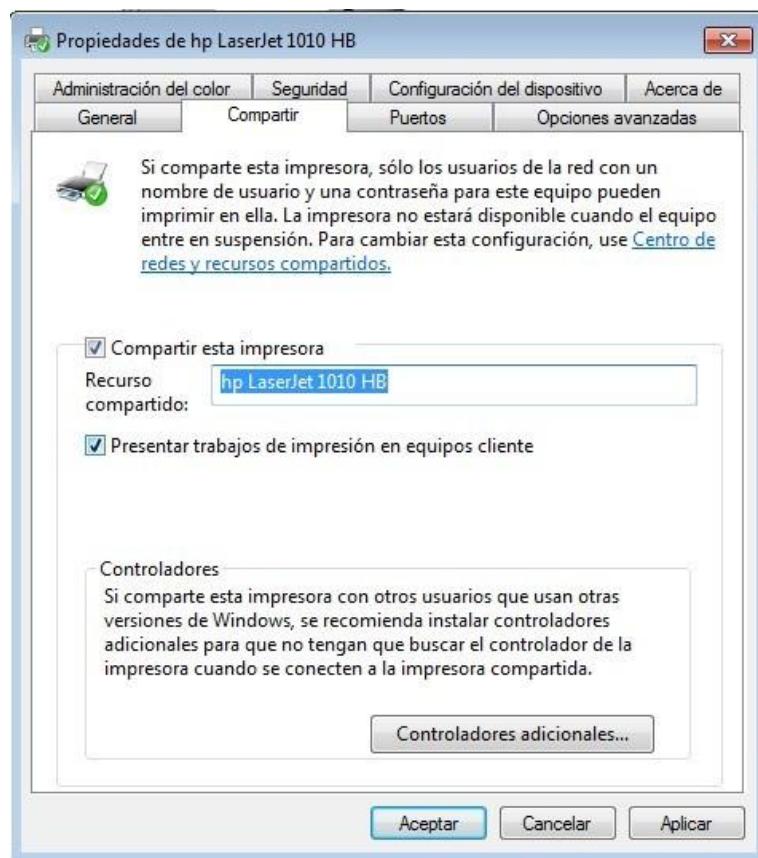
### 5.3. Conexión cliente servidor.

Otro modo de gestionar la impresión en red es conectar la impresora a un ordenador que realice la función de servidor de impresión. El modo de proceder es el siguiente:

- Instalar la impresora localmente en el servidor.
- Ofrecer al resto de ordenadores de nuestra red.
- Instalar en el resto de ordenadores (clientes) el software de la impresora.

De estos pasos es necesario configurar el servidor y el cliente.

Para configurar el servidor en Windows, propiedades de la impresora, compartir, compartir esta impresora.

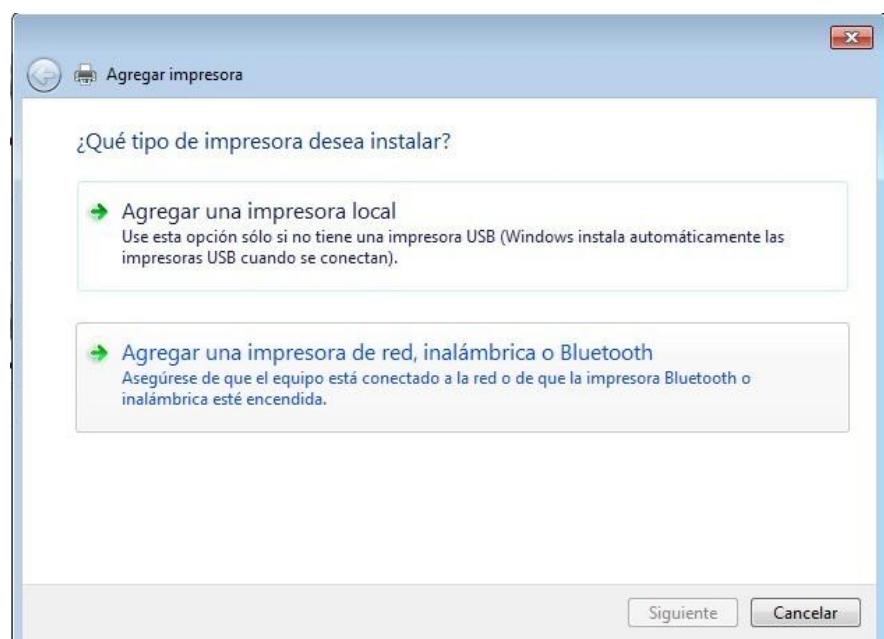


Para el cliente en Windows, agregar la impresora se escoge la segunda opción, y se escoge en la siguiente ventana, buscar una impresora, conectarse a una impresora mediante el equipo (o IP) y el nombre de la impresora o mediante su dirección URL.

Para configurar la impresora en Linux, para que sea visible para los ordenadores de nuestra red, debemos editar el archivo de configuración /etc/cups/cupsd.conf añadiendo la línea Allow From X.X.X.\*,(en caso de una red de clase C) debajo de cada una en la que diga Allow From 127.0.0.1.

Además, tendremos que hacer que el servicio cups “escuche” a toda la red: buscaremos la línea que diga Listen y pondremos Listen \*:631 (si la línea no aparece, se añade).

Para instalar la impresora en un cliente Linux, igual que en local, pero en la parte de impresoras en red.



**5.4. Colas de impresión.**

Como las impresoras son dispositivos lentos en comparación con la transmisión de datos, los trabajos que le llegan a una determinada impresora se un servidor de impresión se suele almacenar en el disco duro de ese servidor, en lo que se denomina la cola de impresión.

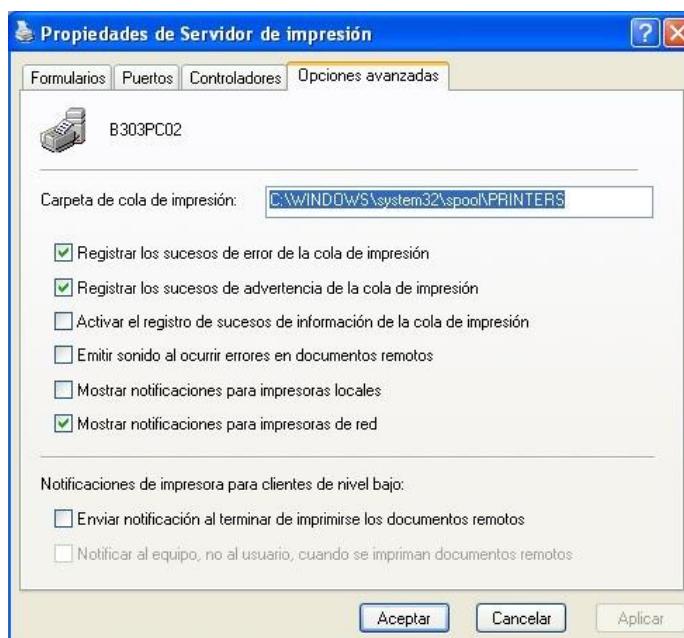
Para mostrar la cola de impresión, en Windows, desde la ventana de las impresoras, doble clic en la impresora, o desde la barra de herramientas doble clic en el icono de la impresora.

Desde la cola de impresión, podemos actuar sobre ellas. Las operaciones son:

- Pausa: retiene la impresión del trabajo hasta que se escoja reanudar.
- Reanudar: Se utiliza después de pausa.
- Reiniciar: se emplea si, mediada la impresión, se ha producido un error y el trabajo sigue en la cola.
- Cancelar: Se usa si ya no deseamos que ese trabajo se imprima.



El servidor de impresión almacena los trabajos en una carpeta. En Windows XP para ver el archivo donde se almacena teclearemos, Inicio ⇒ Impresoras y faxes ⇒ Menú Archivo ⇒ propiedades del servidor y opciones avanzadas.



En Linux, desde la ventana de las impresoras, botón derecho, ver la cola de impresión.

Estado de impresión de documentos (todos los trabajos)							
Archivo	Trabajo	Ver	Documento	Impresora	Tamaño	Hora de envío	Estado
10	profesor	Nuevo Documento ...			291k	hace un minuto	Pendiente
9	profesor	Nuevo Documento ...			291k	hace un minuto	Pendiente
8	profesor	Nuevo Documento ...			291k	hace un minuto	Pendiente
7	profesor	Nuevo Documento ...			291k	hace un minuto	Pendiente

Desde la cola de impresión, podemos actuar sobre ellas. Las operaciones son similares a las de Windows, estas son Cancelar, Eliminar, Retener, Liberar Reimprimir, Obtener, Mover a (Seleccionar otra impresora), autentificar y ver atributos.

En el terminal se puede ver la cola de impresión con el comando lpstat.

### 5.5. Servicios de impresión.

En todos los sistemas operativos, la impresión es ejecutada por un servicio (para evitar problemas de varios procesos imprimiendo). En Windows el servicio encargado de imprimir es cola de impresión (o spooler en inglés).

El servicio de Linux encargado de imprimir es el servicio cups, que arranca el demonio cupsd.

Para reiniciar el servidor es

```
sudo /etc/init.d/cups
```

```
restart
```

## 6. Instalación de software.

Para que nuestro sistema informático realice las funcionalidades deseadas por el usuario, será necesario instalar las aplicaciones de usuario necesarias. Dependiendo de las aplicaciones instaladas, nuestro equipo cumplirá con la funcionalidad deseada.

### 6.1. Instalación en Windows.

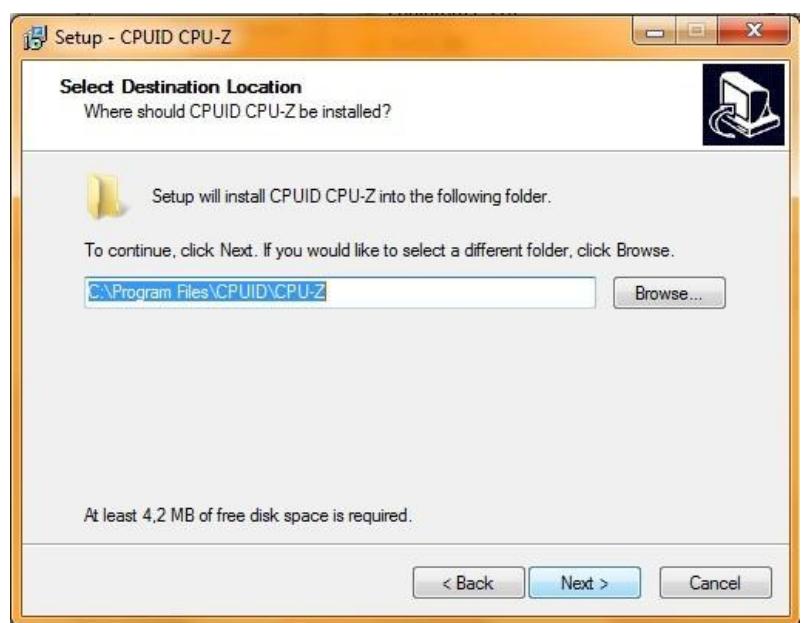
Para instalar una aplicación en Windows, la mayoría de las aplicaciones disponen de un archivo llamado Instalar, Install o Setup que realiza el proceso de instalación.

Son programas ejecutables que suelen realizar las siguientes tareas:

Una ventana de la licencia, donde tendremos que aceptar las condiciones si el programa es gratuito, o introducir la licencia si es de pago.

La copia de archivos al sistema de ficheros, que suelen ser varios archivos de configuración de la aplicación. Se suelen pedir la ubicación donde se van a instalar los archivos, lo normal y estándar es la carpeta C:\Archivos de programa (Program files), donde se generará una carpeta para cada aplicación.

Al copiar suele indicar cuánto espacio de disco duro es necesario para instalar la aplicación.

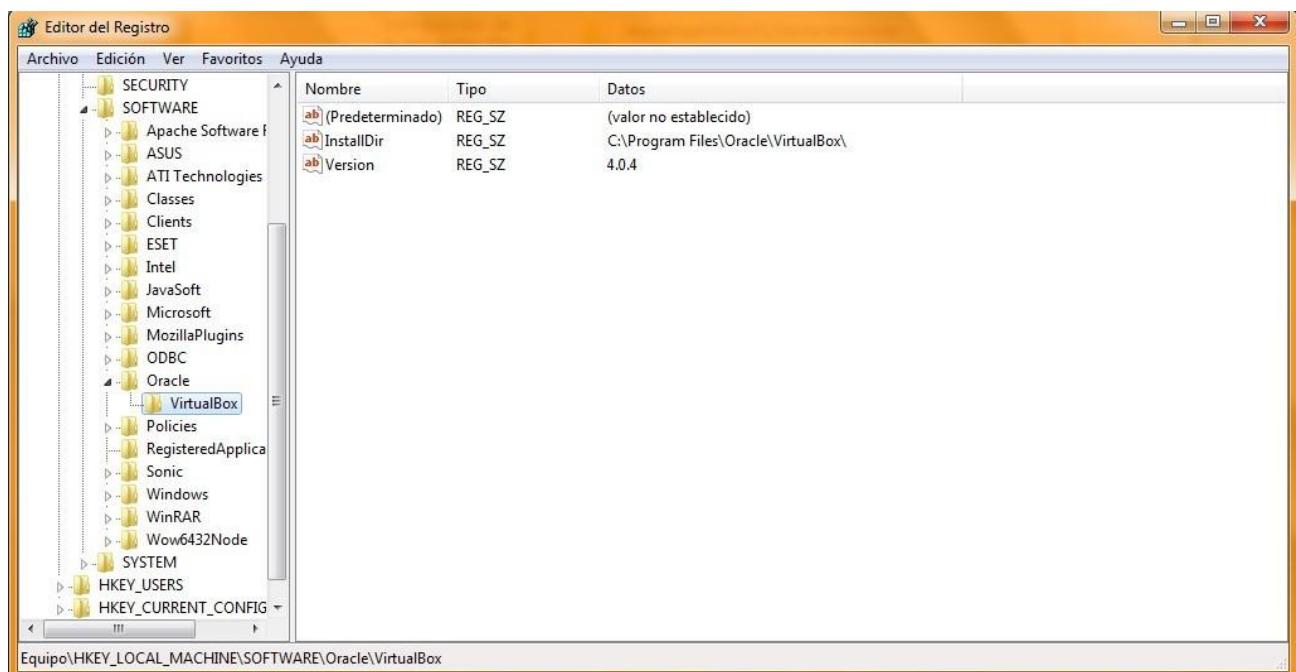


Introducir valores en el registro del sistema. Existe una base de datos donde almacena multitud de datos del sistema. Esta base de datos es denominada registro de Windows. Almacena las configuraciones y opciones del sistema operativo Microsoft Windows en sus versiones de 32 bits, 64 bits.

El registro de Windows contiene información y configuraciones de todo el hardware, software, usuarios, y preferencias del PC. 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.

Para acceder al registro de Windows, se accede a través del archivo regedit.exe.

Las aplicaciones al instalarse, suelen dejar claves, valores, etc. en el registro, concretamente en la entrada HKEY\_LOCAL\_MACHINE\SOFTWARE.



Crea los enlaces directos, tanto en el escritorio como en el menú de inicio del sistema, que posibilita el acceder a la aplicación sin buscar el archivo ejecutable de la aplicación.

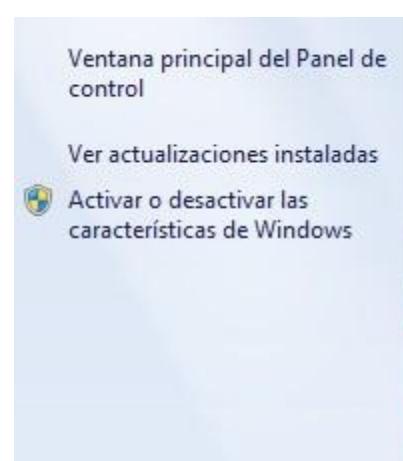
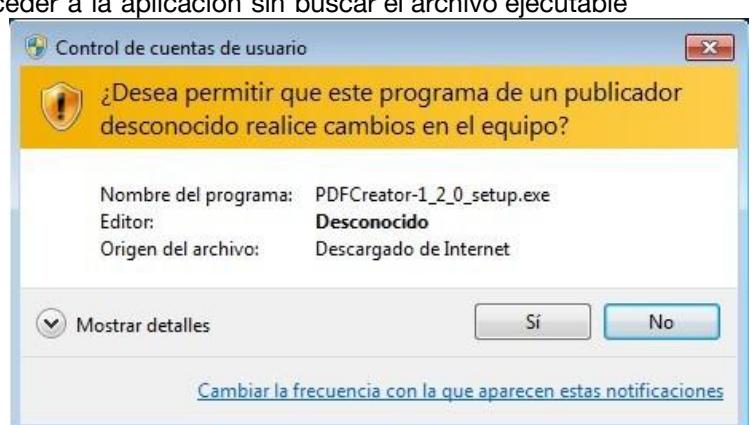
No todos los instaladores, ni todas las aplicaciones cumplen estos pasos, pero suelen ser lo normal casi todas.

A partir de Windows 7, al instalar software, pide una confirmación para permitir que el programa realice cambios en el equipo. Esto es una medida de seguridad para que todo el software que se instale en el equipo, el usuario tenga constancia de ello.

Para gestionar software en Windows, existe en el panel de control, una opción que en Windows 7 es Programas y características.

Las opciones que permite Windows 7 son:

Ver actualizaciones instaladas: Muestra los parches instaladas después de la instalación del sistema.



Desinstalar o cambiar un p

Para desinstalar un programa, selec

Organizar ▾

Nombre

- 7-Zip 16.04 (x64)
- Adobe Acrobat Reader DC - Español
- Adobe Flash Player 10 ActiveX
- Adobe Flash Player 10 Plugin

Obtener nuevos programas en línea: Se utiliza para descargar nuevo software del sistema operativo desde Internet.

Ver software adquirido: Se utiliza para comprar y descargar otro tipo de software desde Internet. Esta opción complementa a la anterior.

Activar o desactivar las características de Windows: Desde aquí se instala y desinstala software del propio sistema operativo.

Vista del software instalado: en esta aparece el software instalado que no forma parte del sistema operativo. Desde aquí podremos desinstalarlo, modificarlo o repararlo.

## 6.2. Instalación de software en Linux.

En Linux existen varios modos de añadir, eliminar o actualizar aplicaciones del sistema. Ten en cuenta que no todas las aplicaciones disponibles para Linux están disponibles por defecto para ser instaladas. Será necesario habilitar la posibilidad de instalar ciertas aplicaciones de forma manual.

Los principales modos de instalar aplicaciones en Linux son:

- Archivos Run. Los archivos con extensión .run suelen ser los asistentes para la instalación en Linux.
- Archivos binarios. Los archivos con extensión .bin son los programas ejecutables en Linux.
- Paquetes. Los ficheros con extensión .deb (para distribuciones derivadas de debian) o con extensión .rpm (para distribuciones derivadas de RedHat) son paquetes de aplicaciones ya preparados para instalarse de una forma sencilla en tu sistema.
- Desde programa de gestión de software como Synaptic, Adept (es la versión de Synaptic para KDE), centro de software de Ubuntu, etc.
- Los programas avanzados apt-get o apt. Estos son programas más avanzados que se ejecutan en modo terminal. Son muy potentes y te permiten también añadir y quitar aplicaciones del sistema entre otras cosas. (Apt es más completo que apt-get, recuerda las librerías descargadas y las desinstala si están en desuso). Para ver la ayuda de cualquier programa que se ejecute en modo terminal: (man nombre\_del\_programa). Ejemplo: man apt
- Desde hace unos años, Canonical (la empresa que desarrolla Ubuntu), ha puesto de moda los paquetes snaps. Los paquetes Snap se obtienen desde Snapcraft, la tienda de aplicaciones universales para Linux que mantienen Canonical, y para instalarlo, no hay que saber cómo se instala algo desde tu distro específica, solo hay que hacer un click, y las dependencias del paquete van incluidas en el paquete. Muchas aplicaciones tienen dos 2 formas instalar. Ejemplos de aplicaciones de este tipo spotify, discord, notepad++, etc

En resumen, existen 2 formas, utilizando Internet o una instalación off-line, y dentro de off- line, utilizando paquetes preconfigurados o utilizando código fuente. Y desde internet 2 formas, instalando el paquete y sus dependencias, o desde los paquetes snap.

### 6.2.1. Desde Internet.

Para la instalación desde Internet, será necesario disponer de conexión a Internet y solo se puede tener activo un programa a la vez. Por ejemplo, si tenemos abierto el programa Synaptic no podremos usar apt.

Para que funcione correctamente este método, es necesario configurar los repositorios.

Un Repositorio es un sitio donde se almacenan y mantienen los paquetes de aplicaciones y librerías o dependencias, o sea, es una biblioteca de software.

Cada vez que instalamos una aplicación, el sistema busca en los repositorios los paquetes necesarios para instalarla, que pueden ser de distintos tipos: imágenes, librerías, código fuente, documentación, traducciones, ... y el programa en sí.

El software se almacena en distintos repositorios separados, según sean sus características o procedencia, de este modo se puede dar acceso o restringirlo a un determinado software.

Por regla general, los repositorios son servidores ftp o http, aunque también pueden ser locales y estar disponibles en un dispositivo físico (DVD-Rom). Por todo el mundo hay repartidos servidores espejo, para no saturar los servidores principales y poder utilizar aquellos que estén más cerca de nuestra ubicación para una descarga más rápida.

En concreto, en Ubuntu tenemos repositorios separados por origen (uno oficial y otro mantenido por la comunidad) y por tipo (libres, privativos o con licencias restringidas), además de los correspondientes a las actualizaciones (updates, security, backports)

Los repositorios pueden añadirse, eliminarse o cambiar de varias formas:

- editando manualmente el archivo /etc/apt/sources.list.
- desde el comando add-apt-repository ppa: UbicacionRepositorio. (será necesario conocer el nombre completo del repositorio).
- mediante una herramienta gráfica que proporciona Ubuntu. **Software y actualizaciones**

```
#deb cdrom:[Ubuntu 10.10 _Maverick Meerkat - Release i386 (20101007)]/ maverick 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/ maverick main restricted
deb-src http://es.archive.ubuntu.com/ubuntu/ maverick main restricted

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

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team. Also, please note that software in universe WILL NOT receive any
## review or updates from the Ubuntu security team.
deb http://es.archive.ubuntu.com/ubuntu/ maverick universe
deb-src http://es.archive.ubuntu.com/ubuntu/ maverick universe
deb http://es.archive.ubuntu.com/ubuntu/ maverick-updates universe
deb-src http://es.archive.ubuntu.com/ubuntu/ maverick-updates universe

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## whether this is OK to do.
```

En la versión 18.10, se debe seleccionar software y actualizaciones, y desde este entorno podemos configurar, los servidores desde que se descarga el software, que versiones se descargan en las actualizaciones, las autenticaciones son desde un sitio autentificado, etc.

son:



los requisitos de la licencia de Ubuntu. 100% fiables.

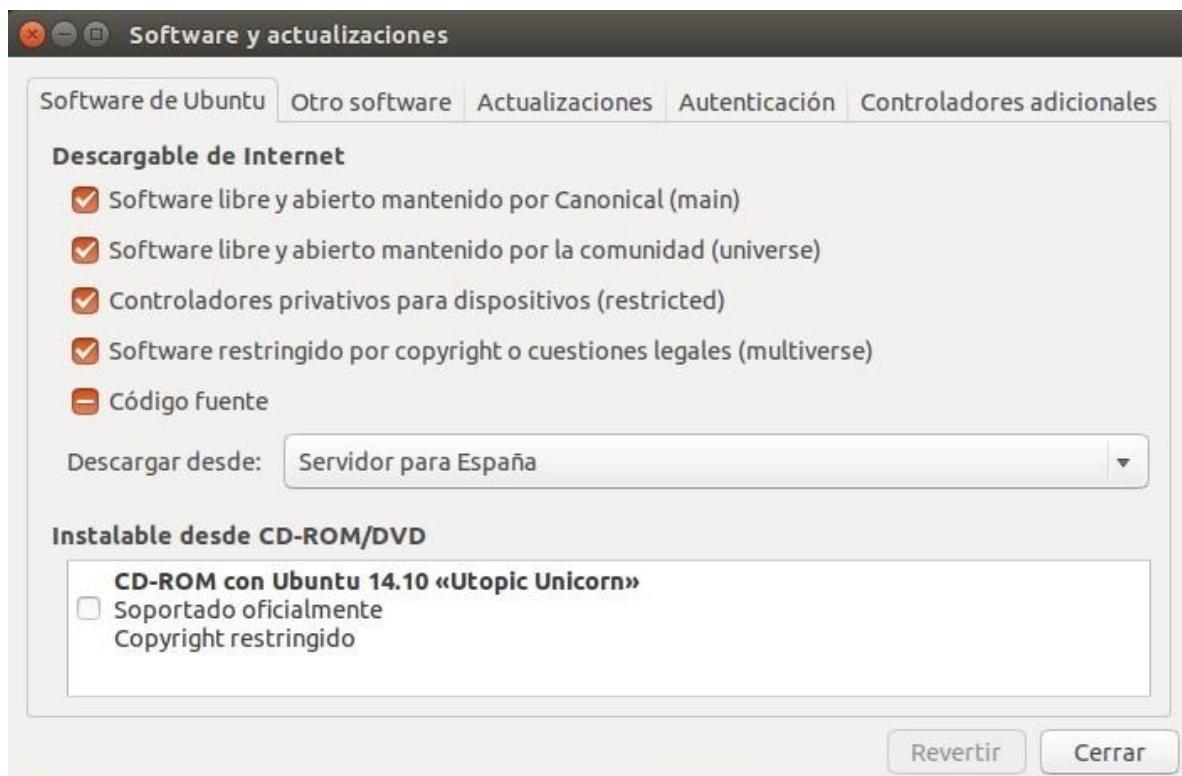
2. Estos paquetes (universe) pueden tener o no una licencia restringida, no recibe apoyo por parte del equipo de Ubuntu, pero sí por parte de la comunidad. Es interesante tener esto activado ya que nos permite instalar un gran número de programas.

3. Paquetes (restricted) sin incluir Licencia libre. Son paquetes que pueden ser necesarios como los controladores propietarios de tarjetas gráficas, pero a los que lo desarrolladores de Ubuntu no tienen acceso.

4. Paquetes (multiverse) sin soporte alguno. No cumplen ninguno de los requisitos. Pongamos por caso Medibuntu (Medibuntu (Multimedia, Entertainment & Distractions in Ubuntu, es decir, Multimedia, entretenimiento y distracciones en Ubuntu) es un repositorio de paquetes de software que no pueden ser incluidos en la distribución Ubuntu por motivos como problemas legales de copyright, licenciamiento o restricciones de patentes).

5. Desde aquí podemos seleccionar el Servidor que mejor nos resulte. Algunas veces puede ser que el servidor que tenemos asignado nos de error en algún paquete. Esto puede ser causado por que el servidor esté en ese momento actualizándose o que no tenga aún ese paquete. Podemos intentar bajarnos las actualizaciones desde otro, tenemos muchos donde elegir.

6. También podemos, si lo tenemos preparado, añadir paquetes desde un CD/DVD.



### 6.2.2. Centro de software de Ubuntu.

El programa El centro de software de Ubuntu es la forma más sencilla en Ubuntu de instalar o quitar programas. También es la más limitada.

Puedes encontrar el programa en Menú Aplicaciones ⇒ Centro de software de Ubuntu



Para instalar aplicaciones selecciona una de las categorías que se muestran en la pantalla principal del programa de la izquierda. De este modo se actualizará la ventana

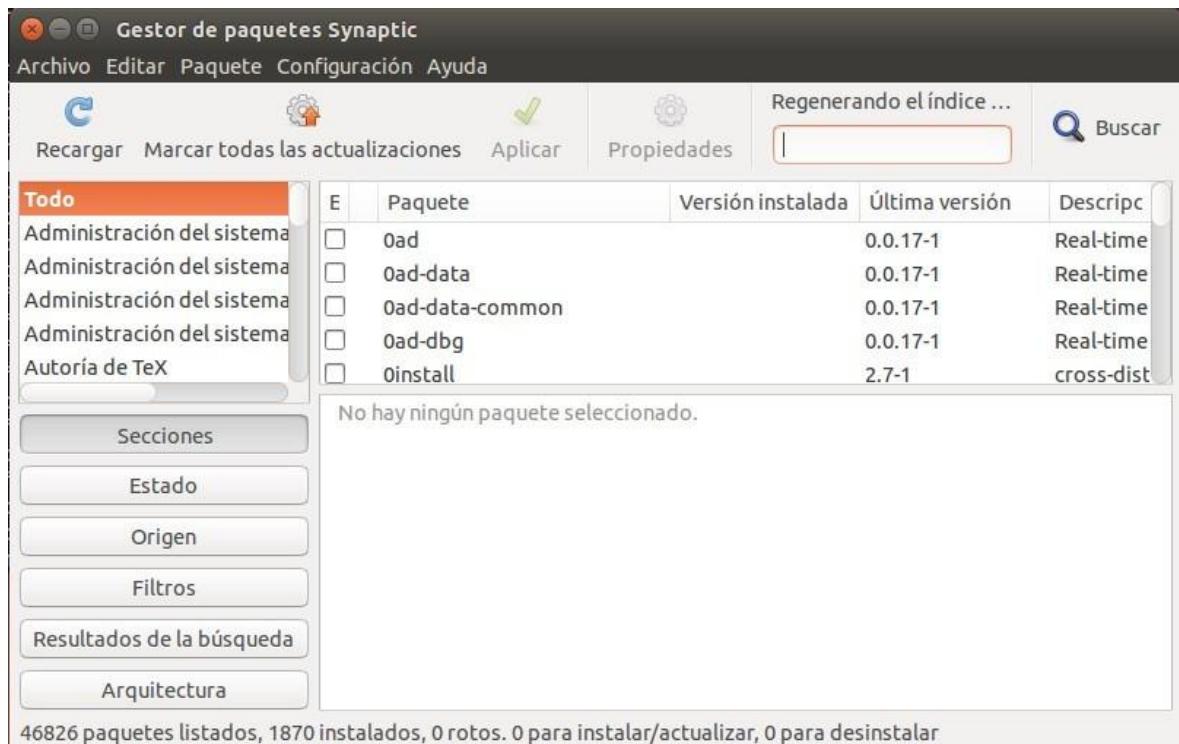
mostrando en ella los programas disponibles de esa categoría. Ahora solo tienes que buscar el programa que quieras instalar y pulsar dos veces sobre él. La ventana mostrará una descripción del mismo y te dará la opción de instalarlo pulsando el botón Instalar.

Si no sabes en qué sección se encuentra el programa que estás buscando. Introduce el nombre de la aplicación que deseas instalar en la caja de búsqueda de la parte superior derecha. A medida que vayas escribiendo el nombre del programa se irá reduciendo la lista de candidatos posibles, hasta que des con el que andabas buscando.



### 6.2.3. Synaptic.

Synaptic es una herramienta, que, en las últimas versiones de Ubuntu con esta instalada, que permite instalar, actualizar, desinstalar y verificar aplicaciones de tu sistema. El entorno es gráfico, como en El centro de software de Ubuntu, pero mucho más potente. Con Synaptic tienes los controles completos de los paquetes (aplicaciones) instalados en tu sistema.



Para instalar un paquete, seleccionar una categoría, pulsar con el botón derecho en el paquete deseado y seleccionar “marcar para instalar” o hacer *doble-click* en el nombre del paquete.

Se marcan todos los paquetes que deseas instalar en el sistema y pulsa Aplicar para proceder a su instalación. Synaptic descargará ahora los paquetes necesarios desde los repositorios en internet o desde el CD de instalación.

También puedes usar el botón Buscar para encontrar los paquetes que deseas instalar. Pulsando en el botón de buscar, podremos realizar búsquedas de programas mediante su

nombre o descripción. Una vez localizado el programa que deseamos instalar hacemos doble clic sobre él para instalarlo. Si queremos eliminar un programa bastará con pulsar con el botón derecho encima de éste y seleccionar eliminar o eliminar por completo.

El sistema de instalación de software en Ubuntu es muy potente y versátil. Dentro de los repositorios las aplicaciones están organizadas en "paquetes". Cada paquete tiene otros de los que depende para su correcto funcionamiento. Synaptic se encarga de resolver estas dependencias e instalar los paquetes necesarios por ti. Pero no sólo eso. En los paquetes de aplicaciones también se indican otros paquetes que, aunque no son necesarios para que funcione la aplicación que queremos instalar si son de utilidad. Estos son los "*paquetes recomendados*".

Podemos configurar Synaptic para que considere esos paquetes "recomendados" como si fuesen dependencias y así los instalará también automáticamente.

Para ello desde el entorno Synaptic ⇒ Configuración ⇒ Preferencias, en la Pestaña General marca la casilla "Considerar los paquetes recomendados como si fuesen dependencias".

Las distribuciones con KDE (como Kubuntu) disponen del equivalente de Synaptic, llamado administrador experto Adept. Se puede encontrar en el menú KDE ⇒ Sistema ⇒ Administrador experto. El funcionamiento es muy similar al de Synaptic.

Para distribuciones basadas en RedHat, el administrador de paquetes más usado actualmente se llama yum que puede ir con una herramienta más potente llamada Yast.

#### 6.2.4. Modo texto apt y apt-get.

Aunque podemos instalar programas de forma gráfica como hemos visto en los puntos anteriores, siempre podemos recurrir a la terminal para instalar cualquier programa.

A muchos usuarios noveles esta opción les puede parecer un poco más complicada y algo crítica. Nada más lejos de la realidad; cuando uno se acostumbra es mucho más cómodo, fácil y rápido.

Existen dos formas de instalar programas en modo texto: con apt y con apt-get. En algunas distribuciones la herramienta apt no está instalada, y es necesario instalarla (sudo apt-get install apt). También existe otra herramienta llamada aptitude.

Ambos programas son muy similares, salvo en un detalle: apt recuerda las dependencias que se han aplicado en la instalación de un paquete. Esto significa que si se instala o actualiza una aplicación con apt y luego se quiere desinstalar, apt borrará el programa junto con todas sus dependencias (excepto si son usadas por otros paquetes). Si se instala con apt-get o con entorno gráfico Synaptic, la desinstalación borrará sólo el paquete especificado, pero no las dependencias.

Desde terminal (Aplicaciones ⇒ Accesorios ⇒ Terminal) se pueden realizar muchas tareas, entre ellas.

Instalar paquetes:

```
$ sudo apt install
```

<paquetes> Desinstalar

paquetes:

\$ sudo apt remove <paquetes>

Desinstalar paquetes (incluyendo archivos de configuración):

```
$ sudo apt purge <paquetes>
```

Actualizar la lista de paquetes disponibles:

```
$ sudo apt update
```

Actualizar el sistema con las actualizaciones de paquetes disponibles:

```
$ sudo apt upgrade
```

Obtener una lista de opciones del comando:

```
$ sudo apt help
```

### 6.3. Desde paquetes preconfigurados.

En caso de que se quiera instalar algún programa en un ordenador sin Internet, es necesario descargar los paquetes en cuestión desde un ordenador que sí disponga de una conexión.

Existen 2 tipos de paquetes, los .deb es la extensión del formato de paquetes de software de Debian y derivadas y los .rpm para paquetes pensados en distribuciones RedHat.

Se puede descargar un paquete desde otro ordenador (o sistema operativo), o mediante el siguiente comando en ese equipo:

```
$ apt download <paquete>
```

De esta manera se descargará el paquete dentro del directorio donde nos encontramos (puede ser en la carpeta del usuario, el escritorio, etc.), cabe mencionar que si descargamos el paquete dentro del directorio personal no es necesario ejecutar el comando con permisos de *root*, ya que tenemos permiso de escritura dentro de dicho directorio.

Una vez descargado el paquete se procede a instalar. Para paquetes .deb se utiliza dpkg. El programa dpkg es la base del sistema de gestión de paquetes de Debian GNU/Linux.

```
$ sudo dpkg -i nombre_paquete
```

La opción -i busca las dependencias y la opción -r para eliminar un paquete.

En los sistemas basados en RedHat, la instalación se realiza con el comando rpm.

Se puede instalar paquetes rpm en distribuciones basadas en Debian (y viceversa). Para instalar estos paquetes es preciso convertirlos antes al formato .deb. Para ello se usa la aplicación alien (se debe de instalar previamente), la cual se puede instalar con el programa dpkg. La aplicación alien se utiliza de la terminal y ejecutamos la siguiente instrucción:

```
$ sudo alien <paquete>.rpm
```

De esta forma el programa crea un archivo con el nombre del paquete, pero con extensión .deb, que se podrá instalar siguiendo la explicación Paquetes Deb.

También se puede instalar paquetes desde el entorno, sólo tienes que hacer doble click sobre el fichero en el navegador Nautilus y automáticamente se lanzará la aplicación gdebi (si está instalada), que se ocupará de instalar el paquete y buscar las dependencias de otros paquetes que pudiera necesitar para su correcta instalación.

### **6.3.1. Paquetes Autopackage (de extensión .package).**

El proyecto Autopackage nace con la idea de facilitar la instalación de aplicaciones en Linux independientemente de la distribución y escritorio que usen. Por ello muchos proyectos lo utilizan, como por ejemplo Inkscape.

Instalar por primera vez un archivo de extensión .package es muy sencillo. Basta con seguir las siguientes indicaciones (en la página del proyecto también se indica cómo).

Una vez descargado el archivo, debemos darle permisos de ejecución, hacer doble clic sobre el archivo y en el aviso en el que pregunta *¿Quiere ejecutar o ver su contenido?* debemos hacer clic en Ejecutar. Una vez hecho esto, empezará el instalador del programa Autopackage y del contenido del paquete.

Al instalarse el programa Autopackage, el próximo archivo de este tipo que deseas instalar bastará con hacer doble clic sobre él sin necesitar hacer nada de lo escrito anteriormente.

### **6.3.2. Archivos binarios (.bin).**

Los archivos con extensión .bin son archivos binarios. No contienen un conjunto de programas o librerías como los paquetes, sino que son el programa en sí mismo. Normalmente se suelen distribuir bajo este sistema programas comerciales, que pueden ser o no gratuitos, pero que normalmente no son libres.

Cuando descargamos un archivo de este tipo y lo guardamos en el sistema, no tendrá permiso para ejecutarse.

Lo primero que debemos hacer, por consiguiente, es dar a ese fichero el permiso para ejecutarse. Desplegamos el menú contextual del archivo y elegimos la opción *Propiedades*. Seleccionamos la pestaña *Permisos* y veremos que el archivo tiene permisos de lectura y escritura para el propietario, pero no para ejecución. Activamos la casilla para dar permisos de ejecución y cerramos la ventana.

### **6.3.3. Archivos Run (.run).**

Los archivos .run son asistentes, normalmente gráficos, que ayudan a la instalación. Para ejecutarlos basta con introducir en la terminal:

```
$ sh ./<archivo>.run
```

Normalmente, en el caso de que necesite permisos de superusuario (también llamado administrador o *root*) pedirá la contraseña; de no ser así basta con agregar la orden *sudo* antes del comando, con lo cual quedaría así:

```
$ sudo sh ./<archivo>.run
```

## **6.4. Desde código fuente.**

A veces se encuentran aplicaciones que no proporcionan paquetes de instalación, y hay que compilar a partir del código fuente. Para ello, lo primero que debemos hacer en Ubuntu es

utilizar un meta-paquete llamado build-essential (sino esta se instala).

En general, los pasos a seguir para compilar una aplicación son los siguientes:

1. Descargar el código fuente.
2. Descomprimir el código, generalmente está empaquetado con tar comprimido bajo gzip (\*.tar.gz) o bzip2 (\*.tar.bz2).
3. Entrar en la carpeta creada al descomprimir el código.
4. Ejecutar el script configure (sirve para comprobar las características del sistema que afectan a la compilación, configurando la compilación según estos valores, y crear el archivo makefile).
5. Ejecutar el comando make, encargado de la compilación.
6. Ejecutar el comando sudo make install, que instala la aplicación en el sistema, o mejor aún, instalar el paquete checkinstall, y ejecutar sudo checkinstall. Esta aplicación crea un paquete .deb de forma que no haya que compilarlo la próxima vez, aunque no incluye la lista de dependencias.