

## Unidad 3

---



# Administración y aseguramiento de la información

## Implantación de sistemas operativos



# Índice



## 3.1. Sistemas de archivos

- 3.1.1. Sistemas de archivos más importantes
- 3.1.2. Rutas y nombres de archivos. Estructura jerárquica

## 3.2. Gestión de enlaces

- 3.2.1. Enlaces simbólicos
- 3.2.2. Enlaces físicos

## 3.3. Estructura de directorios de sistemas operativos libres y propietarios

- 3.3.1. Windows
- 3.3.2. Linux

## 3.4. Identificación del software instalado mediante comandos y herramientas gráficas

- 3.4.1. Windows
- 3.4.2. Linux

## 3.5. Gestión de la información del sistema

## 3.6. Montaje y desmontaje de dispositivos en sistemas operativos

- 3.6.1. Linux
- 3.6.2. Montar un volumen en una o más carpetas en Windows

## 3.7. Administración de particiones

- 3.7.1. Windows
- 3.7.2. Linux

## 3.8. Permisos locales de acceso a ficheros y directorios

- 3.8.1. Windows
- 3.8.2. Linux

## 3.9. Niveles RAID

## 3.10. Tipos de copias de seguridad

- 3.10.1. Copia de seguridad en entorno Windows
- 3.10.2. Copia de seguridad en entorno Linux



# Introducción

Uno de los principales objetivos a lo largo de la historia de la informática ha sido que haya un traspaso de información.

Esta información se puede transmitir entre distintos usuarios, entre un usuario y una computadora o simplemente entre computadoras que no necesitan de usuarios.

Pero bien, la información como elemento no tangible es susceptible de cambios y erro-

res y precisamente ese es uno de los principales puntos a tratar en esta asignatura.

La información se intercambia en distintos procesos y es importante que se mantenga su integridad y su confidencialidad.

Para poder conseguir estos objetivos se han desarrollado todas las herramientas como backups o particiones que desarrollaremos a lo largo de esta unidad.

## Al finalizar esta unidad

- + Comprenderemos cómo funcionan los principales sistemas de archivos y sus principales características.
- + Sabremos cómo funcionan en los distintos sistemas tanto los enlaces físicos como los simbólicos.
- + Diferenciaremos entre las distintas estructuras de directorios dependiendo del sistema de archivos usado.
- + Seremos capaces de identificar el software que hay instalado en cada sistema mediante distintos medios.
- + Gestionaremos la información del sistema mediante permisos, copias de seguridad y particiones en el almacenamiento.
- + Sabremos cómo desmontar y montar dispositivos en los distintos sistemas operativos.
- + Conoceremos los niveles raid existentes y sus principales características



# 3.1.

## Sistemas de archivos

Se llama sistema de archivos al sistema de almacenamiento usado en un dispositivo de almacenamiento que tiene la tarea de estructurar y organizar la escritura, búsqueda, lectura, edición y eliminación de la información de una manera concreta.

Este sistema tiene la funcionalidad de ayudar al usuario a que los archivos estén correctamente identificados y a facilitar un rápido acceso a memoria. Además, los sistemas de archivos otorgan de ciertas cualidades a los archivos, como son:

- > Nomenclaturas.
- > Atributos de archivo.
- > Controles de acceso.

Otra característica fundamental de los sistemas de archivos es que actúan de intermediarios entre el propio sistema operativo y los dispositivos que se conecten al equipo, tanto internos como externos.

Siempre que queramos instalar un sistema de archivos deberemos de **formatear** previamente el medio donde queramos que vaya el sistema.

Hoy en día lo normal es que el medio vaya con un sistema de archivos preinstalado, como es el caso de un USB, o que vengan formateados como suele ser el caso de un disco duro interno.

### 3.1.1. Sistemas de archivos más importantes

Dependiendo del sistema operativo utilizado también cambia el sistema de archivos que se usará, aunque existen multitud de sistemas de archivos, algunos para el mismo sistema operativo.

Con el paso del tiempo, las diferencias entre estos han sido más extensas debido a la evolución de la informática.

Lo que sí que tienen en común todos los sistemas de archivos es que comparten una **estructura de árbol** con la que se organizan los directorios y archivos, siempre partiendo del directorio raíz.

Vamos a ver ahora los principales sistemas de archivos y los más extendidos:





### FAT (File Allocation Table o tabla de asignación de archivos)

Se trata de un sistema de archivos creado en 1980 y que cuenta con varias versiones publicadas debido a la evolución que ha sufrido a lo largo de los años, las cuales son: FAT12, FAT16 y FAT32.

Este sistema de archivos surgió con la intención de que se administrara un volumen de datos pequeño, ya que pueden tener archivos con un tamaño básico de 4 GB y una partición con un máximo de 8TB. Esto es debido a que su versión más reciente, que es FAT32 sigue siendo antigua, datándose en 1997.

Incluso con todos estos inconvenientes suele ser un formato muy usado en dispositivos de almacenamiento extraíbles como un USB o incluso en algunos dispositivos especiales como routers o radios de coche, porque su principal ventaja es que tienen una alta compatibilidad.

### exFAT (Extended File Allocation Table o tabla de asignación de archivos extendida)

Se trata de una evolución del sistema de archivos FAT que se incluyó en 2006. Como su antecesor, fue diseñado para dispositivos portátiles y extraíbles con capacidad de almacenamiento individual.

Aunque funciona perfectamente con datos pequeños, exFAT puede trabajar con mayores rangos de memoria que exceden por mucho los 4GB que podía usar FAT32. A partir de Windows 7, exFAT es compatible con estos sistemas sin necesidad de una instalación concreta y de otros controladores.

### NTFS (New Technology File System)

Es el sistema de archivos por excelencia de los sistemas operativos Microsoft Windows y fue introducido por primera vez en 1993. Una de sus principales ventajas es que se pueden comprimir los medios de almacenamiento y tenemos una mayor seguridad de los datos como puede ser el cifrado de archivos.

Una de sus principales características y la más importante es que se pueden definir derechos de accesos y recursos compartidos de manera detallada e integral.

### HFS+ (Hierarchical File System)

Se trata de un sistema de archivos lanzado en 1998 que es una evolución de HFS para Apple. Otro modo de llamarlo podría ser Mac OS Extended (HFS+).

La principal diferencia con HFS, es que este es mucho más rápido y gestiona de manera más eficiente las lecturas y escrituras de información. Además, también puede administrar hasta 4000 millones de bloques de archivos o carpetas.

Si queremos usarlo en Linux, se podría con la instalación de ciertos paquetes específicos, mientras que Windows necesita como tal un software adicional.

### APFS (Apple File System)

APFS es un sistema solamente lanzado por Apple para sus dispositivos en 2017 que cumple con todos los requisitos de las unidades SSD modernas.

Este es un sistema de 64 bits, por lo que soporta sin problemas el cifrado de archivos y de información. Si tenemos el SO instalado en un SSD, el propio sistema que era HFS+ se transformará de manera automática en APFS, lo que pasa es que puede que haya algún problema de compatibilidad en algún caso específico.

### ext4

Este sistema fue introducido como sustituto de ext3 en 2008 y sirve de estándar para casi todas las distribuciones de Linux. Su principal ventaja frente a sus predecesores es, que la gestión de archivos grandes está mucho más optimizada y se evita que se fragmente la información en la medida de lo posible. Esto es posible gracias a la función extents.

Además, ext4 permite que las particiones se vayan ampliando y reduciendo conforme a la necesidad que haya, aunque esté en funcionamiento. Por último, ext4 admite unos volúmenes de 1 exabyte como máximo, lo que equivale a un millón de terabytes.

## 3.1.2. Rutas y nombres de archivos. Estructura jerárquica

Para que pueda haber una correcta gestión de los sistemas de archivos es necesaria una organización estructurada que recibe el nombre de sistema de directorios.

El usuario final encontrará la información almacenada en ficheros que a su vez estarán almacenados en directorios que pueden estar dentro de otros directorios hasta llegar al directorio raíz. Esto es la llamada estructura jerárquica del sistema de directorios.

Poniendo en común la estructura jerárquica, entre sistemas tienen diferencias, como que, por ejemplo, puede haber un fichero en varios directorios, pero no en todos los sistemas de archivos.

Una diferencia por ejemplo muy característica es que, en Linux, todos los dispositivos que se encuentran conectados al equipo comparten una única jerarquía de directorios, mientras que, en Windows, en cada unidad de almacenamiento hay una estructura distinta.



# 3.2.

## Gestión de enlaces

Tenemos varios tipos de enlaces que se gestionan de distintos modos en los sistemas operativos. Los tipos de enlaces son:

### > Enlaces simbólicos:

- » Es un fichero nuevo que tiene el nombre del fichero de destino dentro.
- » En Windows son accesos directos y en UNIX son "soft links".

### > Enlaces físicos:

- » Es un único archivo que cuenta con un contador de enlaces.
- » Son los "hard links" de UNIX.

### 3.2.1. Enlaces simbólicos

#### Creación de enlaces simbólicos en Windows

El proceso para la creación de un enlace simbólico mediante línea de comandos en Windows, por ejemplo, del escritorio a la carpeta de descargas es el siguiente:

1. Lo primero que haremos será abrir un símbolo del sistema como administrador, que ya vimos en las unidades anteriores como realizarlo.
2. Una vez en el símbolo del sistema tendremos que lanzar el siguiente comando:

```
mklink/D ruta_destino ruta_origen
```

```
C:\Windows\system32>Mklink/D C:\users\miguel\Downloads\prueba C:\Users\Miguel\Desktop  
vínculo simbólico creado para C:\users\miguel\Downloads\prueba <=> C:\Users\Miguel\Desktop  
C:\Windows\system32>
```

Imagen 1. Enlace simbólico en Windows por CMD.

3. Ahora si nos vamos a descargas podemos comprobar que tenemos una carpeta llamada "prueba", que tiene dentro los archivos del escritorio.

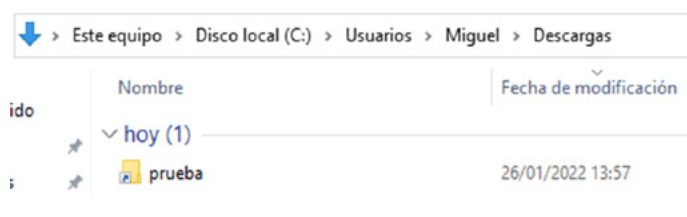


Imagen 2. Enlace simbólico en Windows.



## Creación de enlaces simbólicos en Linux

El proceso para la creación de un enlace simbólico en Linux, por ejemplo, de un archivo de prueba en el directorio home del usuario al escritorio, se hace del siguiente modo:

1. Lo primero que haremos será abrir un terminal y entrar en modo administrador, es decir, loguear como root.
2. Ahora debemos dirigirnos a la ruta de la que queremos crear el enlace.
3. Ahora lanzaremos como administrador el siguiente comando:

```
ln -s archivo_origen enlace_simbólico:
```

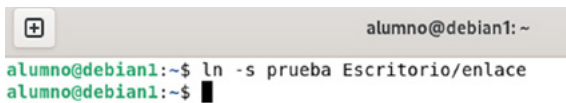


Imagen 3. Enlace simbólico en Linux 1.

4. Ahora nos dirigimos al escritorio y comprobamos si se ha creado dicho enlace con el comando **ls -l** qué, además, nos dirá que es un enlace:

```
alumno@debian1:~/Escritorio$ ls -l
total 0
lrwxrwxrwx 1 alumno alumno 6 ago 31 14:57 enlace -> prueba
alumno@debian1:~/Escritorio$
```

Imagen 4. Enlace simbólico en Linux 2.

## 3.2.2. Enlaces físicos

Las principales características de los enlaces físicos son las siguientes:

- > Se contabilizan a través de un contador de enlaces.
- > Si creamos un enlace estaremos incrementando el contador de enlaces.
- > Si borramos un fichero estaremos decrementando el contador de enlaces.
- > Solo se borran todos los datos del fichero si el contador de enlaces queda en 0 tras un decremento.
- > No existen en Windows.

## Creación de un enlace físico en Linux

Para crear un enlace físico en Linux, lo primero que debemos hacer es lo siguiente:

1. Abrir un terminal y loguearnos como root.
2. Ahora lanzamos el siguiente comando:

```
ln archivo_origen enlace_físico
```

```
alumno@debian1:~$ ln prueba Escritorio/enlace2
alumno@debian1:~$
```

Imagen 5. Enlace físico en Linux 1.

3. Ahora vamos a nuestro escritorio y comprobamos al igual que antes que se ha creado el enlace:

```
alumno@debian1:~/Escritorio$ ls -l
total 0
lrwxrwxrwx 1 alumno alumno 6 ago 31 14:57 enlace -> prueba
-rw-r--r-- 2 alumno alumno 0 ago 31 14:56 enlace2
alumno@debian1:~/Escritorio$
```

Imagen 6. Enlace físico en Linux 2.



# 3.3.

## Estructura de directorios de sistemas operativos libres y propietarios

### 3.3.1. Windows

Los Sistemas Operativos de Windows usan el sistema de archivos NTFS o FAT como hemos podido ver con antelación.

Los volúmenes en Windows suelen identificarse con una letra mayúscula y los dos puntos, pero no siempre será así.

Siempre que se haga una instalación limpia de Windows y si no ha habido problema, debe de tener los siguientes directorios en su unidad principal:

- > **Windows.** Este directorio contiene en su mayoría los archivos del sistema. Contienen multitud de directorios, pero a la hora de la verdad deberemos de tener en cuenta solo los siguientes:
  - » **System32:** aquí se almacenan todos los archivos compartidos del sistema, por ejemplo, como vimos en la unidad anterior, el archivo de hosts para la resolución de nombres se encuentra aquí.
  - » **Boot:** contiene todos los ficheros relativos al arranque del sistema.
  - » **Help:** se almacena la ayuda del sistema.
  - » **PolicyDefinitions:** Aquí están almacenadas las plantillas para las futuras directivas de seguridad.
  - » **Fonts:** se guardan aquí todas las fuentes de escritura de la que dispone el sistema.
  - » **Temp:** almacena archivos temporales.
- > **Archivos de programa.** Este directorio recoge los directorios referentes a los programas que se encuentran instalados en el equipo. Cuando tengamos un equipo con 64bits, también tendremos el directorio Archivos de programa x86 para hacer referencia a los programas de 32 bits que se instalen en el equipo.
- > **PerfLogs.** Es el directorio donde almacenamos los archivos de registro y monitorización de toda la actividad en el equipo. Cada vez que se crea un nuevo usuario e inicia sesión por primera vez se creará una carpeta dentro de esta con el nombre del usuario donde se almacenarán todos los archivos de registro .log que con el uso se irán generando. Estos archivos los podrá eliminar o el administrador o cada usuario los suyos, pero no es una actividad recomendable.
- > **Usuarios.** Aquí al igual que antes, se crearán subcarpetas con los nombres de los usuarios en el sistema y en cada una irán las carpetas personales de cada usuario como ya vimos anteriormente. Además, hay dos carpetas más, una oculta, que es Default para el usuario creado por defecto por el sistema y otra que se llama Acceso público, para poner en común a todos los usuarios.





### 3.3.2. Linux

En Linux se usan los sistemas de archivos extendidos versión 4, es decir, ext4. Este sistema de archivos tiene la particularidad de que, gracias a él, **todo en Linux se puede expresar como un archivo**. Entonces, tenemos que hacer una diferenciación entre las distintas formas que puede tener un archivo.

Los archivos llevan un **identificador** que ayuda a que Linux diferencie los tipos de archivos:

- > **Archivo sencillo (-):** estos pueden contener texto, datos, o ser un ejecutable.
- > **Directorio (d):** es un archivo donde se almacenan o listan otros archivos distintos.
- > **Vínculo (l):** es un enlace simbólico.
- > **Dispositivo de bloque (b) o de carácter (c):** son usados para los dispositivos de entrada y salida que interactúan directamente con el sistema operativo y el hardware.
- > **Socket (s):** este archivo se encarga de facilitar la comunicación entre distintos procesos locales.
- > **Tubería (p):** este archivo nos ayuda a comunicar de manera unidireccional de los procesos.

Estos identificadores se muestran a la izquierda del fichero cuando lancemos el comando:

`ls -l`

```
alumno@debian1:~$ ls -l
total 32
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Descargas
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Documentos
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:58 Escritorio
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Imágenes
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Música
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Plantillas
-rw-r--r-- 2 alumno alumno 0 ago 31 14:56 prueba
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Público
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Vídeos
```

Diferente a lo que pasaba con Windows, donde los directorios colgaban de una unidad lógica, en Linux cuelgan de un directorio raíz. A partir de este directorio raíz o root se extiende el árbol de directorios donde destacan los siguientes:

- > **/bin.** Este directorio contiene todos los archivos ejecutables y los comandos básicos del sistema.
- > **/boot.** Aquí se contienen los archivos que son necesarios para arrancar el sistema operativo además de las imágenes del núcleo del sistema operativo.
- > **/dev.** Se incluyen aquí los dispositivos del sistema.
- > **/etc.** En este directorio se almacenan todos los archivos de configuración del sistema operativo además de los de los programas instalados.
- > **/home.** Se incluyen los directorios personales de los usuarios, para cada usuario se generará un directorio dentro de este. Se puede designar con el símbolo "~".
- > **/lib.** Almacena las librerías que son necesarias a la hora de la ejecución de programas y comandos del sistema.
- > **/media.** Se incluyen las unidades de almacenamiento físicas conectada al equipo.
- > **/mnt.** Es el directorio que se debe usar para el montaje de nuevos dispositivos.
- > **/root.** Es el directorio home del usuario superadministrador o root.
- > **/sbin.** Aquí se almacenan los programas y comandos que requieren de privilegios avanzados para su ejecución.
- > **/usr.** Este es el directorio más grande que hay ya que contienen toda la información relativa a los programas y aplicaciones no básicos del sistema.
- > **/var.** Contiene todo tipo de información variable como puede ser las colas de gestión de recursos del sistema.

Imagen 7. Salida comando ls -l.



# 3.4.

## Identificación del software instalado mediante comandos y herramientas gráficas

### 3.4.1. Windows

#### Mediante el CMD

El proceso sería el siguiente:

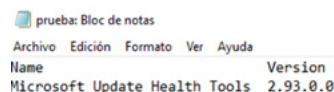
1. En primer lugar, lo que haremos será abrir un CMD o símbolo del sistema como Administrador.
2. Lo siguiente que vamos a hacer es en el CMD escribir el siguiente comando:

```
Wmic /output:ruta_destino\nombreambrivo.txt  
product get name, versión
```

```
C:\Windows\system32>wmic /output:C:\Users\miguel\desktop\prueba.txt product get name, version  
C:\Windows\system32>
```

Imagen 8. Comando wmic.

3. Ahora si nos vamos a la ruta donde hayamos depositado el archivo, que en nuestro caso es en el escritorio, veremos que aparece la lista descrita anteriormente:



Name	Version
Microsoft Update Health Tools	2.93.0.0

Imagen 9. Salida comando wmic.

#### Listar todos los programas instalados con PowerShell

Aunque PowerShell es supuestamente una herramienta para usuarios avanzados, Microsoft tienen la intención de dejarlo como el único intérprete de comandos y eliminar de una vez por todas el CMD porque el primero es más seguro, flexible y potente.

Para poder realizar la anterior tarea desde Powershell, tenemos que hacer lo siguiente:



1. Iniciamos Powershell en modo Administrador.
2. Se introduce el siguiente comando:

```
Get-WmiObject -Class Win32_Product |  
Select-Object -Property Name > ruta_destino\  
archivo.txt
```

```
PS C:\Windows\system32> Get-WmiObject -Class Win32_Product | Select-Object -Property Name > C:\Users\miguel\Desktop\prue  
ba2.txt  
PS C:\Windows\system32>
```

Imagen 10. Comando Get-WmiObject Powershell.

3. Ahora nos vamos al escritorio y si abrimos prueba2.txt nos debe dar el resultado igual que el anterior, pero con distinta estructura.

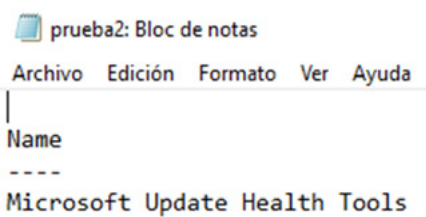


Imagen 11. Salida comando Get-wmiObject.

### 3.4.2. Linux

#### Ver de manera gráfica los paquetes instalados en Linux

Para poder ver que paquetes se encuentran instalados en Linux, tenemos una herramienta gráfica que nos ayuda a gestionarlos llamada **Synpatics**.

Si queremos ver lo que hemos dicho, debemos de ir a "Estado" y después seleccionamos la opción de "Instalado".

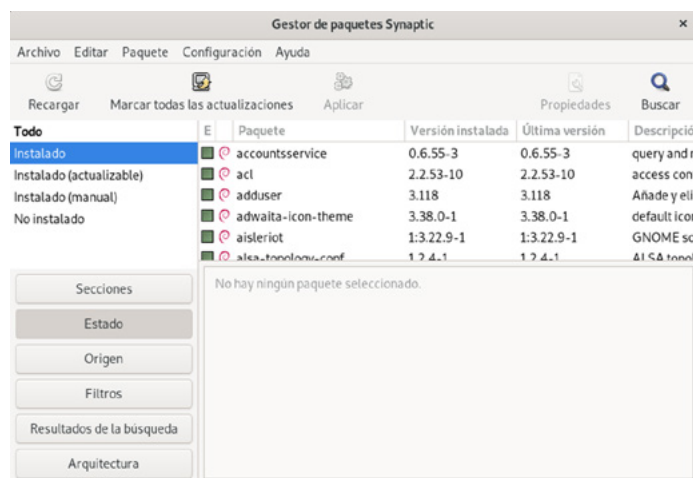


Imagen 12. Synaptics.

Debemos tener en cuenta que para poder gestionar estos paquetes es necesario conocer la contraseña del administrador.



## Ver los registros del sistema

Los sistemas basados en Linux registran de manera continua todas las acciones que se realizan en el equipo, que es lo conocido como logs del sistema. Los usuarios, además, tienen la posibilidad de consultar estos registros.

Si nos vamos a los primeros registros podremos ver los paquetes que se han instalado más recientes.

Esto se puede hacer de varios modos en Linux, consultando los registros de **dpkg** o **apt**. Además, como veremos más adelante, también existe la posibilidad de hacerlo directamente con estos comandos.

En esta ocasión vamos a mostrar como ver los registros de **dpkg** y **apt** **filtrados con el comando grep** para obtener un listado de los paquetes instalados.

En primer lugar, abriremos un terminal y nos loguearemos como root, para después lanzar el comando:

```
grep "install" /var/log/dpkg.log
```

Como podemos ver, es un listado muy largo el que se muestra, ahora vamos a ver el mismo comando, pero para apt:

```
grep "install" /var/log/apt/history.log
```

Observemos que la principal diferencia es que dpkg tiene una entrada en los logs del sistema mientras que apt tiene un directorio entero para sus registros, que dentro tienen lo siguiente:

```
root@debian1:/home/alumno# ls /var/log/apt/  
eipp.log.xz history.log history.log.1.gz term.log term.log.1.gz  
root@debian1:/home/alumno#
```

Imagen 13. ls /var/log/apt/



## Listar paquetes instalados con apt

El comando `apt-get` que se usa para instalar paquetes en Linux no conoce una opción como tal para listar todos los programas instalados, pero `apt` en solitario, sí.

Abrimos un terminal y lanzamos el siguiente comando para que paquetes han sido instalados:

`apt list --installed`

```
root@debian1:/home/alumno# apt list --installed
Listando... Hecho
accountsservice/unstable,now 22.07.5-1 amd64 [instalado, automático]
acl/unstable,now 2.3.1-1 amd64 [instalado, automático]
adduser/stable,now 3.118 all [instalado, actualizable a: 3.121]
adwaita-icon-theme/now 42-beta-1 all [instalado, actualizable a: 42.0-2]
aisleriot/now 1:3.22.9-1+b1 amd64 [instalado, actualizable a: 1:3.22.22-1]
alsa-topology-conf/unstable,now 1.2.5.1-2 all [instalado, automático]
alsa-ucm-conf/unstable,now 1.2.6.3-1 all [instalado, automático]
alsa-utils/unstable,now 1.2.6-1 amd64 [instalado, automático]
anacron/unstable,now 2.3-31+b1 amd64 [instalado, automático]
apache2-bin/now 2.4.52-1 amd64 [instalado, actualizable a: 2.4.53-2+b1]
apg/stable,unstable,now 2.2.3.dfsg.1-5+b2 amd64 [instalado, automático]
apparmor/unstable,now 3.0.4-2 amd64 [instalado, automático]
appstream/now 0.15.2-2 amd64 [instalado, actualizable a: 0.15.4-1]
apt-config-icons/now 0.15.2-2 all [instalado, actualizable a: 0.15.4-1]
apt-listchanges/stable,unstable,now 3.24 all [instalado]
apt-utils/now 2.4.0 amd64 [instalado, actualizable a: 2.5.0]
apt/now 2.4.0 amd64 [instalado, actualizable a: 2.5.0]
aspell-en/stable,unstable,now 2018.04.16-0-1 all [instalado, automático]
aspell-es/unstable,now 1.11-19 all [instalado, automático]
aspell/unstable,now 0.60.8-4 amd64 [instalado, automático]
```

Imagen 14. Salida comando apt-get installed.

De nuevo, podemos ver que la lista es muy amplia y extensa.

## Listar los paquetes instalados mediante dpkg

También se puede usar directamente el comando `dpkg` a la hora de querer listar o enumerar todos los paquetes que se han instalado en el sistema, en este caso Debian.

Lanzamos ahora el comando:

`dpkg-query -f`

```
Deseado=desconocido(U)/Instalar/eliminar/Purgar/retener(H)
| Estado=No/Inst/ficheros-Conf/desempaquetado/medio-conf/medio-inst(H)/espera-o
|/ Err?=(ninguno)/requiere-Reinst (Estado,Err: mayúsc.=malo)
||/ Nombre Versión Ar
+++-----+-----+-----+
ii accountsservice 22.07.5-1 am
ii acl 2.3.1-1 am
ii adduser 3.118 al
ii adwaita-icon-theme 42-beta-1 al
ii aisleriot 1:3.22.9-1+b1 am
ii alsa-topology-conf 1.2.5.1-2 al
ii alsa-ucm-conf 1.2.6.3-1 al
ii alsa-utils 1.2.6-1 am
ii anacron 2.3-31+b1 am
ii apache2-bin 2.4.52-1 am
ii apg 2.2.3.dfsg.1-5+b2 am
ii apparmor 3.0.4-2 am
ii appstream 0.15.2-2 am
ii apt 2.4.0 am
ii apt-config-icons 0.15.2-2 al
ii apt-listchanges 3.24 al
ii apt-utils 2.4.0 am
ii aspell 0.60.8-4 am
lines 1-23
```

Imagen 15. Salida dpkg -l.

Podemos ver que el modo de mostrarse es diferente al anterior y que se puede interactuar, de manera que se puede mover entre líneas, para salir de aquí podríamos lanzar `Ctrl+Z`.

El comando `dpkg -l` puede hacer exactamente la misma función.





## 3.5.

### Gestión de la información del sistema

La gestión de la información o (GI) se refiere a un conjunto de procesos mediante los cuales se procede al control del flujo de información, es decir, desde que se crea o se obtiene la información hasta que se contiene en un archivo o se elimina de manera permanente.

En estos procesos contemplamos la lectura, escritura, modificación y demás operaciones que como hemos visto en anteriores puntos y unidades se pueden realizar con la información.

La principal función de la GI es la de conservar la integridad de los datos.

## 3.6.

### Montaje y desmontaje de dispositivos en sistemas operativos

#### 3.6.1. Linux

Cuando hablamos de **montar** un dispositivo nos referimos a crear un acceso directo a dicha unidad, lo que sería muy similar a un enlace simbólico, por otro lado, si lo que queremos es **desmontar** un dispositivo, sería similar a borrar el enlace.

Linux trae de serie el comando `/mount` para realizar esta tarea.

Cuando montemos un dispositivo, en ese mismo momento los datos pasarán a estar disponibles de manera permanente para poder acceder en el directorio de destino del montaje. Lógicamente, si desmontamos dicho dispositivo, el directorio ya no almacena dicha información.

#### Montar y desmontar dispositivos en Linux

Como hemos mencionado antes, para montar un dispositivo en Linux se usará el comando `mount`

Para este comando existen dos argumentos complementarios que tenemos que especificar para que el comando funcione, esos son:

- > El archivo o partición que queremos que sea montado. (Debe de llevar un sistema de archivos predeterminado para que se pueda montar).
- > El directorio donde va a ser montado.

El directorio donde se va a montar el nuevo sistema de archivos se denomina punto de montaje.

Además, tenemos una opción, que sería el comando `mount -t`, mediante la cual seguidamente se especifica el sistema de archivos que va a ser montado para que, si es diferente, se formatee y se vuelva a poner dicho sistema.

Pero esto no siempre es así, porque depende de la distribución de Linux que usemos. A veces, si no es el sistema de ficheros igual al del dispositivo, simplemente da error de montaje.

Para realizar un montaje extendido o en alguna ruta sin privilegios, deberemos de ser el usuario root.

Para desmontar una unidad en Linux usaremos el comando `umount punto_montaje`.



### 3.6.2. Montar un volumen en una o más carpetas en Windows

---

#### Mediante la interfaz gráfica de Windows

1. Nos dirigimos al administrador de discos, para lo cual, siendo administrador habrá que escribir en inicio o ejecutar, diskmgmt.msc.
2. Hacemos clic derecho sobre la partición o volumen que posee la carpeta donde se va a montar la unidad.
3. Seleccionamos la opción Cambiar la letra y rutas de acceso de unidad,
4. Damos clic en Agregar.
5. Seleccionamos ahora la opción Montar en la siguiente carpeta NTFS vacía.
6. Escribimos la ruta de una carpeta vacía que se encuentre en un volumen NTFS o la buscamos con la opción Examinar.

#### Quitar un punto de montaje

Si lo que queremos es desmontar el punto de montaje en Windows lo que haremos será lo siguiente:

1. Seleccionamos y hacemos clic derecho en la unidad que ha sido montada.
2. Seleccionamos la opción Cambiar letras y rutas de acceso de unidad, como antes.
3. Ahora seleccionamos la carpeta en la lista de las que sale.
4. Hacemos clic en quitar.



# 3.7.

## Administración de particiones

Para poder gestionar de manera eficiente los medios de almacenamiento tenemos las particiones, todo con la intención de que se almacene mejor la información y sea de mejor accesibilidad para los usuarios.

### 3.7.1. Windows

#### Crear particiones en Windows

Para gestionar particiones en Windows tenemos dos modos distintos, el primero que vamos a ver es mediante la **interfaz gráfica**:

1. Lo primero que vamos a hacer es abrir al igual que antes, diskmgmt.msc.
2. Podemos ver que una vez abierto se nos presentan los volúmenes que tenemos en el sistema con sus distintas particiones.

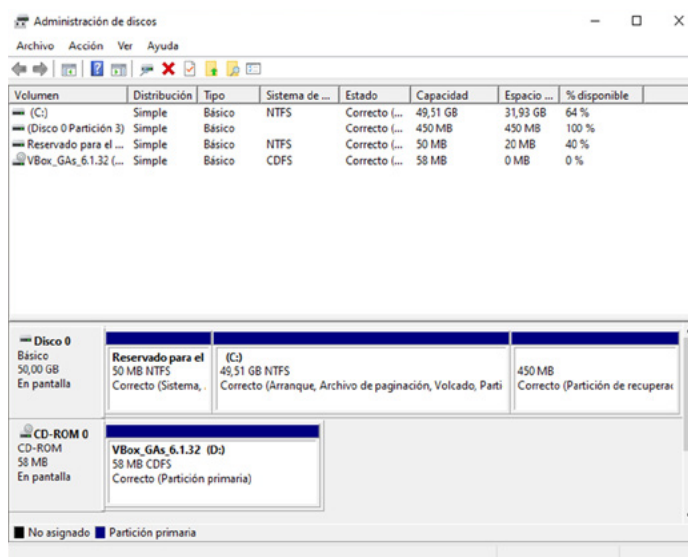


Imagen 16. diskmgmt.msc.

3. Entonces, lo siguiente que vamos a hacer va a ser crear una partición, y para eso, en la unidad C: hacemos clic derecho y hacemos clic en 'Reducir volumen'.

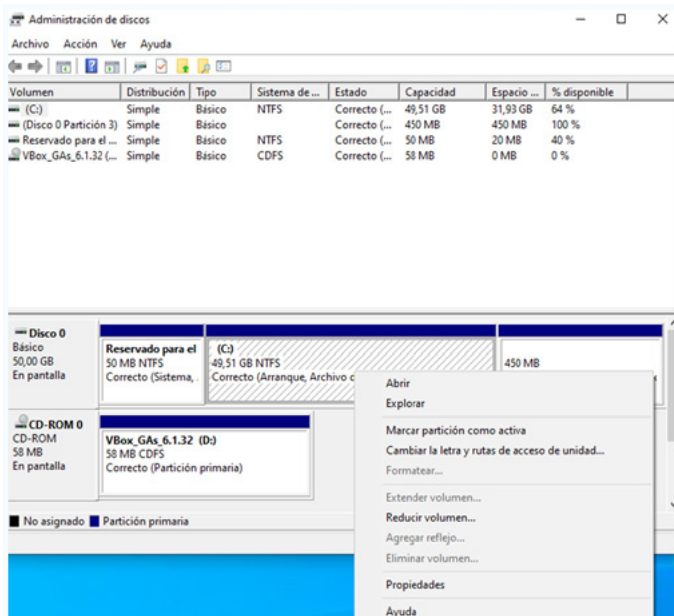


Imagen 17. Crear particiones en Windows 1.

4. Ahora nos muestra un tamaño con cuánto queremos reducir y como se quedará, en nuestro ejemplo reduciríamos 20000 MB.

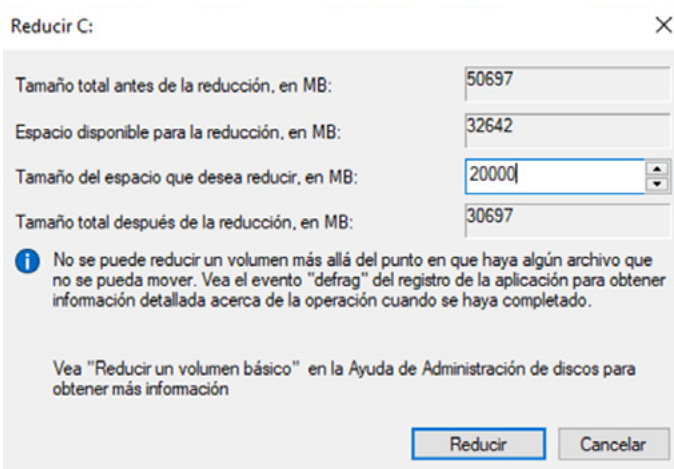


Imagen 18. Crear particiones en Windows 2.

5. Ahora vemos que se ha quedado un espacio 'Sin asignar'.
6. Hacemos clic derecho sobre ese espacio y seleccionamos 'Nuevo volumen simple'.

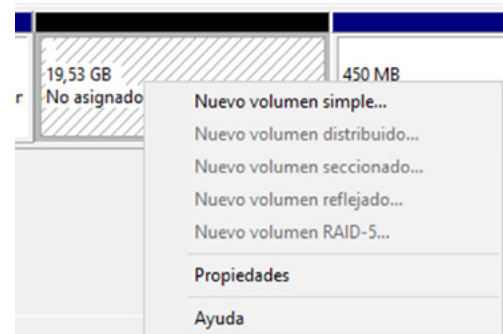


Imagen 19. Crear particiones en Windows 3.

7. Podemos dejarlo todo por defecto y simplemente darle a siguiente hasta que termine el proceso.
8. Ahora, si vemos, ya tendríamos una nueva partición primaria en forma de volumen simple.

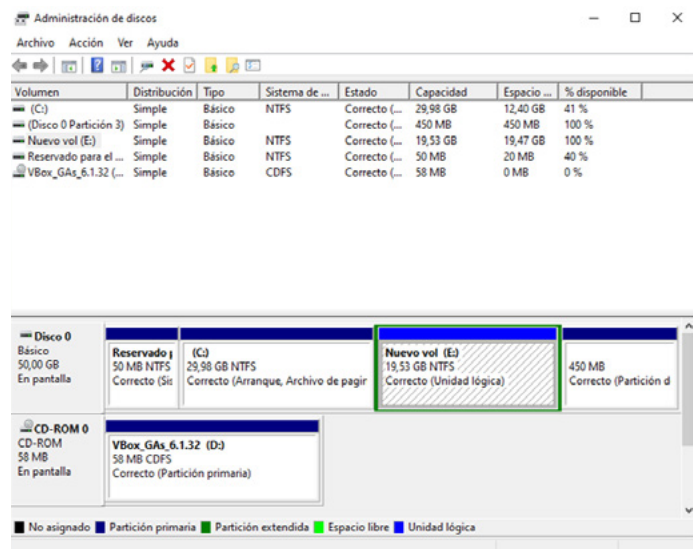


Imagen 20. Crear particiones en Windows 4.



### 3.7.2. Linux

Linux trae de manera general una herramienta para la creación y administración de particiones llamada fdisk.

Para trabajar con fdisk, debemos de hacer lo siguiente:

1. Lo primero que haremos será abrir un terminal y conectarnos como root.
2. Ahora vamos a comprobar los discos disponibles en el sistema, que se hace con el comando `sudo fdisk -l`

```
root@debian:/home/miguel# sudo fdisk -l
Disco /dev/sda: 8 GiB, 8589934592 bytes, 16777216 sectores
Modelo de disco: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xefa4e708

Disposit.  Inicio Comienzo      Final Sectores Tamaño Id Tipo
/dev/sda1  *          2048 14776319 14774272    7G 83 Linux
/dev/sda2          14778366 16775167  1996802    975M  5 Extendida
/dev/sda5          14778368 16775167  1996800    975M 82 Linux swap / Solaris
root@debian:/home/miguel#
```

Imagen 21. Particiones en Linux 1.

3. Como podemos ver tenemos un solo disco, /dev/sda, que tiene solamente 8 Gigabytes. Además, tenemos más abajo la estructura de particiones que automáticamente ha sido creada por Linux, divide en tres particiones: **la del sistema, la extendida y la de swap**.
4. Entramos en la consola de fdisk específica para el disco que tenemos con el comando `sudo fdisk /dev/sda1`, ya que es la partición principal.

```
root@debian:/home/miguel# sudo fdisk /dev/sda

Bienvenido a fdisk (util-linux 2.36.1).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

Orden (m para obtener ayuda):
```

Imagen 22. Particiones en Linux 2.

5. Ahora pulsamos la opción `m` y se nos mostrará la siguiente ayuda:





Ayuda:

```
DOS (MBR)
a  conmuta el indicador de iniciable
b  modifica la etiqueta de disco BSD anidada
c  conmuta el indicador de compatibilidad con DOS

General
d  borra una partición
F  lista el espacio libre no particionado
l  lista los tipos de particiones conocidos
n  añade una nueva partición
p  muestra la tabla de particiones
t  cambia el tipo de una partición
v  verifica la tabla de particiones
i  imprime información sobre una partición

Miscelánea
m  muestra este menú
u  cambia las unidades de visualización/entrada
x  funciones adicionales (sólo para usuarios avanzados)

Script
I  carga la estructura del disco de un fichero de script sfdisk
O  vuelca la estructura del disco a un fichero de script sfdisk

Guardar y Salir
w  escribe la tabla en el disco y sale
q  sale sin guardar los cambios

Crea una nueva etiqueta
g  crea una nueva tabla de particiones GPT vacía
G  crea una nueva tabla de particiones SGI (IRIX) vacía
o  crea una nueva tabla de particiones DOS vacía
s  crea una nueva tabla de particiones Sun vacía
```

Imagen 23. Particiones en Linux 3.

6. Como queremos crear una partición nueva, vamos a usar la opción **n**.

```
Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (0 primaria(s), 0 extendida(s), 4 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p):
```

Imagen 24. Particiones en Linux 4.

7. Tenemos dos opciones para elegir como podemos ver, la partición se puede crear tanto primaria como extendida. Vamos a crear una partición primaria, por lo que vamos a seleccionar la opción **p**.
8. Podremos ver que, si vamos dando a intro, se dejarán los valores por defecto, que es lo que queremos.

```
Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (0 primaria(s), 0 extendida(s), 4 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (1-4, valor predeterminado 1): 1
Primer sector (2048-14774271, valor predeterminado 2048):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (2048-14774271, valor predeterminado 14774271):

Crea una nueva partición 1 de tipo 'Linux' y de tamaño 7 GiB.

Orden (m para obtener ayuda): █
```

Imagen 25. Particiones en Linux 5.

9. Ahora ejecutamos el comando `sudo partprobe` para que el kernel de Linux relea la estructura de la tabla de particiones.



# 3.8.

## Permisos locales de acceso a ficheros y directorios

### 3.8.1. Windows

Cuando hablamos de permisos en Windows tenemos dos tipos de permisos que editar y/u otorgar a los usuarios que se deseen.

Estos son los permisos para restringir el acceso a archivos y carpetas en común, que son los permisos NTFS y los permisos para registrar accesos a recursos de red.

Los permisos NTFS son en los que nos vamos a centrar y tienen las siguientes características:

- > Se aplican a todos los archivos y carpetas que se encuentran almacenados en un volumen concreto del sistema que contiene el sistema de archivos NTFS.
- > Tienen activada la herencia de los permisos de manera nativa, es decir, si una carpeta tiene unos permisos asignados, por defecto las subcarpetas se crearán con los mismos permisos.
- > Los permisos principales que se ofrecen en los sistemas Windows son:
  - » Lectura
  - » Escritura
  - » Lectura y ejecución
  - » Listar
  - » Modificación
  - » Control total

### Como otorgar permisos NTFS en Windows

Para otorgar estos permisos vamos a usar el siguiente ejemplo, el usuario principal del equipo creará una carpeta en C: que se llamará Probando y dentro de la misma un archivo de texto.

Lo primero que haremos será dar solo permiso de listar en la carpeta para ver su contenido y lo segundo será dar permiso de modificación para poder ver el archivo y además borrarlo.

1. Creamos la carpeta y el archivo
2. Ahora hacemos clic derecho en la carpeta y damos a Propiedades, donde nos aparecerán las propiedades de la carpeta.
3. Nos vamos a la pestaña de Seguridad y podemos ver los permisos que tiene actualmente.

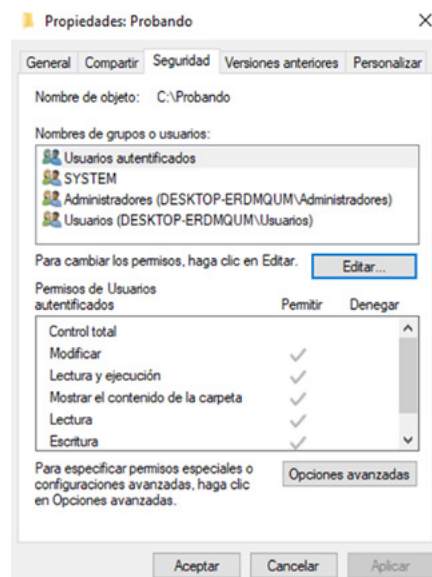


Imagen 26. Permisos en Windows 1.

4. Damos a Editar y nos aparecerá otra pestaña muy parecida, donde seleccionaremos la opción de Agregar.
5. Una vez aquí, tendremos que seleccionar el usuario que queremos, no de una lista, sino poniendo el nombre y buscando coincidencias.
6. Vemos que una vez que lo hemos puesto, aparecerá para otorgar los permisos que como hemos dicho serán solo de listar de momento.

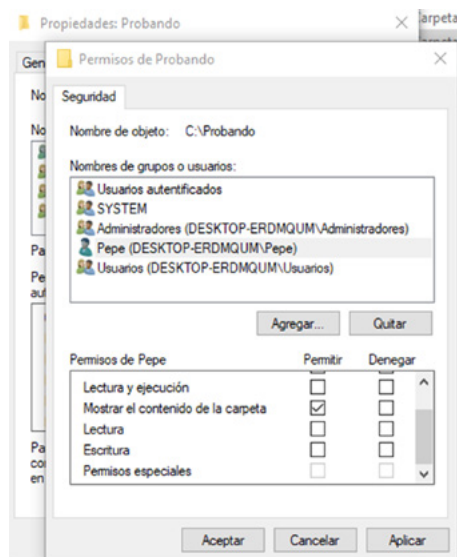


Imagen 27. Permisos en Windows 2.

7. Ahora si iniciamos sesión con pepe, veremos que se puede ver el contenido de la carpeta.

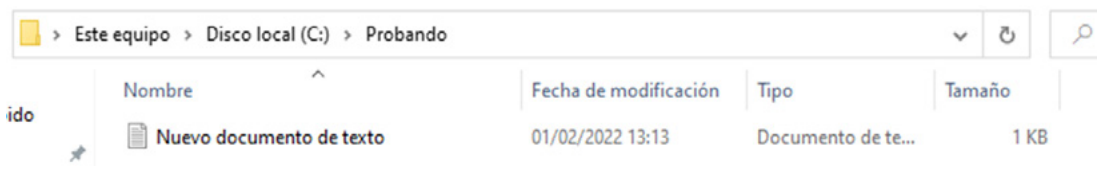


Imagen 28. Permisos en Windows 3.

8. Pero, por otro lado, si queremos borrar el documento, nos dirá que no podemos.
9. Lo que tenemos que hacer es otorgar los permisos de igual modo, pero cambiando los permisos y dando a escritura y se marcarán todos los demás de manera automática para que pueda borrar el archivo.



### 3.8.2. Linux

En Linux tenemos tres niveles de permisos de manera principal, que son los siguientes:

- > **Permisos del propietario**
- > **Permisos del grupo**
- > **Permisos del resto de usuarios**

Los permisos del propietario son los permisos que tiene el creador del archivo o directorio en primera instancia cuando se crean, estos también pueden cambiar ya sea porque cambiamos el propietario o porque se cambien los permisos de este.

En Linux es común que los usuarios pertenezcan a distintos grupos de trabajo ya que por defecto se añade un nuevo usuario a ciertos grupos. Estos permisos se aplican a todos los usuarios de un grupo predefinido para el archivo. Si no se pone nada, el grupo que se pone por defecto es el del usuario propietario.

Los permisos del resto de usuarios son los permisos que afectan al resto de usuarios que no forman parte del grupo en cuestión antes mencionado ni son el propietario.

Para ver los permisos por ejemplo de los directorios del home de un usuario lanzamos el comando `ls -l`.

```
alumno@debian1:~$ ls -l
total 32
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Descargas
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Documentos
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:58 Escritorio
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Imágenes
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Música
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Plantillas
-rw-r--r-- 2 alumno alumno 0 ago 31 14:56 prueba
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Público
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Vídeos
alumno@debian1:~$
```

Imagen 29. ls -l

Como podemos ver en la imagen de más arriba los permisos se dividen en grupos de tres en el mismo orden que se ha establecido al principio: **propietarios, grupo, otros**.

Además, vemos que hay hasta tres letras, r, w y x:

- > **R:** es el permiso de lectura que permite que se lea el contenido del archivo en cuestión. No es suficiente para que se pueda ver el contenido de un directorio.
- > **W:** es el permiso de escritura que nos permite hacer cambios dentro de un directorio o de un archivo en concreto.
- > **X:** es el permiso de ejecutar que nos permite ejecutar programas o script de Linux. Además, es necesario que se acompañe junto al permiso de lectura si se quiere ver el contenido de un directorio.

También podemos ver que salen dos nombres justo después, el primero hace referencia al usuario propietario y el segundo hace referencia al grupo propietario.



## Cambiar usuario y grupo propietario en Linux

Ahora, para cambiar permisos sin tener que tocar estos permisos, vamos a cambiar tanto el propietario como el grupo propietario de un archivo.

Lo que vamos a hacer es, por ejemplo, de la imagen de arriba, cambiar al archivo prueba para que su propietario y grupo sea el de todos los demás en los dos contextos.

1. Abrimos una terminal y nos logueamos como root.
2. Nos dirigimos a la ruta /home del usuario, aunque estaremos por defecto.
3. Ahora, para cambiar el propietario usamos el comando `chown nuevo_propietario archivo`. Y después usaremos `ls -l` para ver si se ha realizado el cambio.

```
root@debian1:/home/alumno# chown root prueba
root@debian1:/home/alumno# ls -l
total 32
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Descargas
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Documentos
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:58 Escritorio
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Imágenes
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Música
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Plantillas
-rw-r--r-- 2 root alumno 0 ago 31 14:56 prueba
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Público
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Vídeos
root@debian1:/home/alumno#
```

Imagen 30. Cambiar propietario en Linux.

4. Si queremos cambiar el grupo lo que haremos será ejecutar el comando

`chgrp grupo archivo`

```
root@debian1:/home/alumno# chgrp root prueba
root@debian1:/home/alumno# ls -l
total 32
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Descargas
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Documentos
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:58 Escritorio
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Imágenes
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Música
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Plantillas
-rw-r--r-- 2 root root 0 ago 31 14:56 prueba
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Público
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Vídeos
root@debian1:/home/alumno# █
```

Imagen 31. Cambiar grupo propietario en Linux.





## Cambiar permisos de usuario en Linux

Para cambiar los permisos de usuario en Linux, usaremos el comando `chmod`, pero tenemos dos vertientes, una usando letras y otra con dotación numérica.

Con letras, tenemos claro que los permisos que cambiaremos son los de los tres grupos de arriba nombrados.

Para esto necesitaremos saber que se identifican del siguiente modo: u, dueño; g, grupo y o, otros.

Ahora que ya tenemos en cuenta esto, vamos a cambiar los permisos del archivo prueba de modo que el grupo y otros usuarios también puedan modificar.

1. Abrimos un terminal y logueamos como root.
2. Lanzamos el siguiente comando

```
chmod u/g/o+-r/w/x archivo/directorio
```

```
root@debian1:/home/alumno# chmod go+w prueba
root@debian1:/home/alumno# ls -l
total 32
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Descargas
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Documentos
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:58 Escritorio
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Imágenes
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Música
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Plantillas
-rw-rw-rw- 2 root root 0 ago 31 14:56 prueba
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Público
drwxr-xr-x 2 alumno alumno 4096 ago 31 14:48 Vídeos
root@debian1:/home/alumno# █
```

Imagen 32. Cambiar permisos en Linux por letras.

3. Como podemos ver, se pone todo de manera consecutiva, por ejemplo, hemos puesto go para hacer referencia al grupo y a otros usuarios y +w para que quede claro que se añade el permiso de escritura.
4. Cuando hemos lanzado la opción `ls -l`, vemos que los permisos han sido cambiados.
5. Si, por otro lado, quisiéramos quitar de nuevo estos permisos, el comando sería

```
chmod go-w prueba
```

Ahora vamos a ver como se cambian los permisos en Linux con el método numérico. Para esto debemos de saber lo primero que cada permiso se asocia con un número:

> R → 4                      > W → 2                      > X → 1

El comando total queda del siguiente modo `chmod n1n2n3 archivo`, donde n1 son los permisos para el propietario, n2 para el grupo y n3 para los otros usuarios.

Cada uno de los números lo sacaremos de la suma de cada uno de los números de los permisos, por eso, el máximo permiso que se puede otorgar es el 777.

El ejemplo anterior en esta numeración sería:

```
chmod 666 prueba
```

# 3.9.

## Niveles RAID

**RAID**, que significa Redundant Array of Independent Disk se trata de un sistema de almacenamiento en el que se agrupan varias unidades de disco duro interconectadas de manera que se consiga optimizar al máximo lo siguiente:

- > **Redundancia:** una tolerancia muy alta a fallos.
- > **Rendimiento:** una velocidad de transferencia muy alta.
- > **Coste:** su implementación conlleva un coste muy bajo.

Tenemos varios tipos de RAID que se pueden formar y que se elegirán conforme a los recursos y las necesidades de cada caso en concreto.

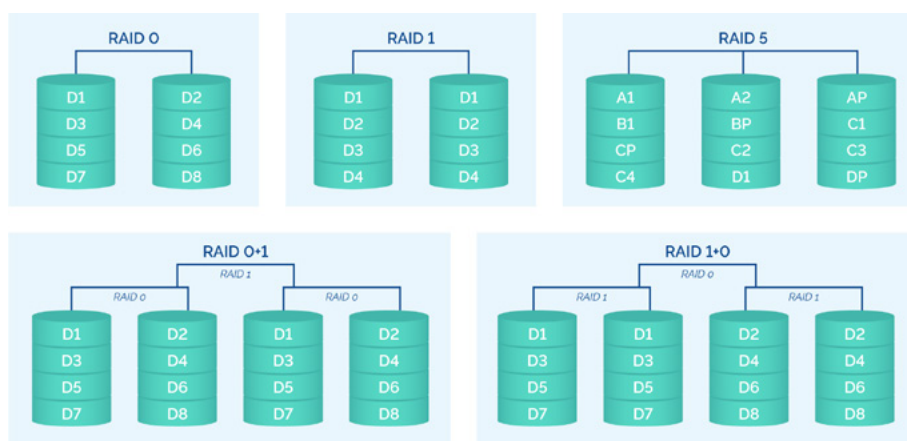


Imagen 33. Niveles RAID.

Tenemos los siguiente niveles RAID

- > **RAID 0.** También recibe el nombre de stripping y se forma usando varios discos, en general dos y por bloques se va distribuyendo la información en ambos.

Características RAID 0	
Factor	Descripción
Redundancia	Ninguna
Rendimiento	A más discos, más velocidad
Coste	Asequible
Uso habitual	Para audio, vídeo e imagen



- > **RAID 1.** También se le puede llamar mirroring y consiste en agrupar parejas de discos y de este modo que uno de cada pareja sea copia exacta del otro para conservar los datos.

Características RAID 1	
Factor	Descripción
Redundancia	Existe
Rendimiento	Bueno
Coste	Alto
Uso habitual	Entornos con trabajo ininterrumpido

- > **RAID 5.** En esta variante se distribuyen en bloques los datos de manera que se mejora la redundancia y se reduce el coste. Esto hace que posiblemente sea la mejor opción de todas en relación calidad/precio.

Características RAID 5	
Factor	Descripción
Redundancia	Existe
Rendimiento	Bueno
Coste	Económico
Uso habitual	En servidores

- > **RAID 0+1.** Se trata de una fusión entre el stripping y el mirroring, consiguiendo un espejo de divisiones, es decir, son dos RAID 0 duplicados. Este modelo es a la hora de la verdad el que más se usa en los entornos empresariales.
- > **Raid 1+0.** A la inversa del anterior, se hacen dos RAID 1 y después se unifican distribuyendo la información en forma de RAID 0. Si lo que queremos es una alta disponibilidad, este es el más adecuado.



## 3.10. Tipos de copias de seguridad

Hay varios casos diferentes de copia de seguridad y de diversos tipos, pero se pueden destacar los más usados, que son estos tres:

- > **Copia de seguridad completa.** Se hace una copia de seguridad de todos los datos que se quieren respaldar sin diferencia.

Es el tipo de copia más usado y se llama en otras ocasiones copia base porque puede servir de base a los otros tipos de copias.

Como es tan simple, a la hora de la verdad es de las copias más rápidas.

- > **Copia de seguridad incremental.** Para poder realizar esta copia tenemos que disponer de un archivo de copia de referencia. Lo primero que hará la copia incremental será partir de la copia base y se irán anotando los diferentes archivos que cambian con respecto al archivo que hemos nombrado al principio.

Aunque como su tamaño es menor que el de una copia completa, tarda menos, el tener que disponer de la copia base y de todos los archivos incrementales para poder recuperar la información hace que se produzca un retardo en cuanto al proceso de restauración se refiere.

- > **Copia de seguridad diferencial.** En este tipo de copia, el archivo de referencia siempre va a ser la copia base, por lo que simplemente se realizará la copia de los archivos que han ido cambiando a lo largo del tiempo.

### 3.10.1. Copia de seguridad en entorno Windows

Windows trae de manera genérica y por defecto una serie de herramientas que nos permiten realizar y gestionar las copias de Seguridad. Estas son las herramientas:

- > **Copia de seguridad de archivos:** estas copias de seguridad afectan a todos los usuarios y datos del equipo. Nos permite que se haga una selección de archivos si queremos y si no, el propio sistema se encarga de dicha selección. Estas copias se realizan por defecto de manera periódica, pero se puede personalizar totalmente.

Una vez que se ha creado la primera copia, el mismo sistema hará un reconocimiento de archivos nuevos o modificaciones que se debieran de añadir al repetir la copia.

- > **Copia de seguridad de imagen del sistema:** este tipo de copia es a nivel de sistema completo, crea una imagen del sistema que se puede volcar en otro equipo o en el mismo en caso de errores.

Es importante tener en cuenta que, si seleccionamos esta opción, a la hora de realizar el volcado en otro equipo, debe de tener unas características similares y que destruirá todo lo que se albergue anteriormente.

- > **Versiones anteriores de archivos:** desde las propiedades de los archivos y/o carpetas, Windows nos da la posibilidad de ver hasta dos días anteriores en una hora determinada, el contenido de este elemento y poder restaurarlo. Eso sí, en caso de restauración, el elemento actual se pierde automáticamente.

- > **Restaurar el sistema:** por último, estos sistemas te permiten crear puntos de restauración del sistema de forma que se creen en distintos puntos distribuidos en el tiempo.

Si tenemos estos puntos creados, el propio sistema nos dará la opción de restaurar toda la información a ese momento específico que elijamos.



### 3.10.2. Copia de seguridad en entorno Linux

---

Linux no trae de manera nativa herramientas que nos permitan gestionar las copias de seguridad como tal.

La opción disponible es realizar copias de seguridad nosotros mismos mediante como comandos como tar, dd, o cp, pero en realidad es una réplica o compresión de información y no un backup ciertamente.

Lo que si se encuentra en Linux es una serie de aplicaciones que funcionan muy bien cuando queremos realizar copias de seguridad, las más famosas son:

- > **Aplicaciones sin interfaz gráfica**
  - » Rsync
  - » Obnam
  - » Rdiff-backup
- > **Aplicaciones que realizan copias de la imagen al completo del sistema**
  - » DARGui
  - » UrBackup
- > **Aplicaciones para clonación de discos y recuperación de información**
  - » Partimage
  - » Clonezilla
- > **Aplicaciones para copias de seguridad en Cloud**
  - » SparkleShare
  - » Rclone





 [www.universae.com](http://www.universae.com)

