

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS FILE SYSTEMS

CFGS DAW
DPT INF

1 - Introducción	2
1.1 – Unidades de almacenamiento	2
1.1.1 Estructura lógica de un disco duro	2
Diferencia entre MBR y GPT	4
1.2 Conceptos básicos	6
1.2.2 Los directorios	8
1.2.2.1 Directorio raíz	9
1.2.3 Rutas de los archivos	9
2 - Sistemas de archivos	11
2.1 Características y funcionalidades	13
2.1.1 Fragmentación del disco	14
2.2 Otras características de los sistemas de archivos	16
2.2.3 Sistemas comprimidos	17
2.2.4 Sistemas virtuales	18
2.3 Principales sistemas de archivos	19
3 Gestión de sistemas de archivos	22
3.1 Montaje de los sistemas de archivos	23
3.1.1 Windows	23
3.1.2 Gnu/Linux	23
3.2 Gestión desde el entorno gráfico	23
3.2.1 Windows	23
3.2.2 Gnu/Linux	25
3.3 Gestión por medio de órdenes	25
3.3.1 Windows	25
3.3.2 Gnu/Linux	26
4 Estructura de directorios de los sistemas operativos	27
4.1 Directorios en Windows	27
4.2 Directorios en GNU/Linux	28
4.2.1 FHS: Filesystem Hierarchy Standard	29
4.2.2 Todo son archivos	29
4.2.3 Organización del sistema de archivos según FHS	29
5.- Permisos y atributos	31
5.1.- Permisos y atributos en Windows	31

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

5.1.1 Atributos	31
5.1.2 Configuración de permisos NTFS	32
5.1.3 Permisos estándar y especiales	33
5.1.4 Añadir permisos	35
5.1.5 Permisos heredados y explícitos	37
5.1.6 Permisos efectivos	38
5.2.- Permisos en Gnu/Linux	39
5.2.1 Introducción	39
5.2.2 Cambiar los permisos por comandos	42
5.2.3.- Permisos especiales en Gnu/Linux	44
5.2.4.- Máscaras de permisos	45
5.2.5.- Permisos ACL	46
6 Gestión de enlaces	48
6.1 Enlaces simbólicos	48
6.2 Enlaces rígidos	49

1 - Introducción

En un sistema informático la información se guarda dentro de soportes de almacenamiento (disco duro, CDs, USBs, etc) en forma de archivos o ficheros y la forma en que se guardan esos archivos en cada soporte de almacenamiento la determina el **sistema de archivos** de ese soporte que establece la estructura, nombre, forma de acceso, uso y protección que tendrá cada archivo almacenado.

1.1 – Unidades de almacenamiento

Son fundamentales en cualquier sistema informático porque es donde se guarda de forma permanente la información con la que trabaja el ordenador (programas y datos) ya que son **dispositivos no volátiles** (no se borra su contenido cuando dejan de recibir electricidad como ocurre con la memoria RAM).

1.1.1 Estructura lógica de un disco duro

La estructura lógica de un soporte se crea cada vez que se formatea.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Primer Sector

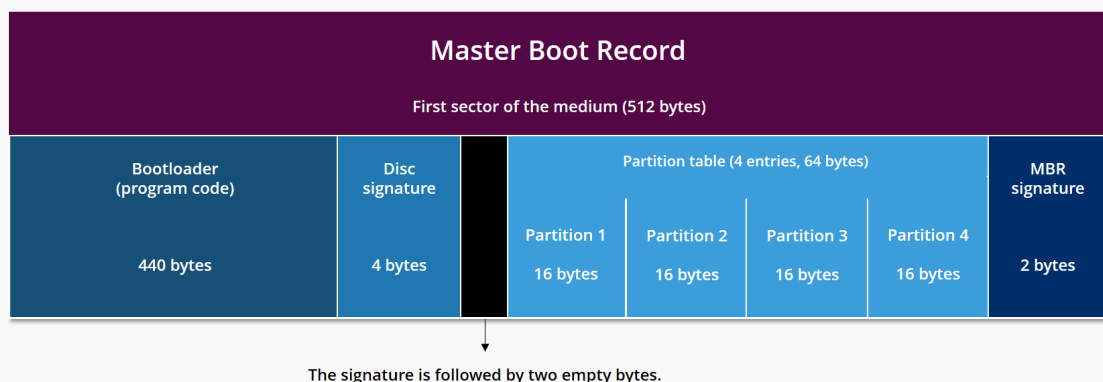
Último sector



Sector de arranque (MBR o GPT)

El **Master Boot Record** o **MBR** es el primer sector del disco duro, **pista 0, cilindro 0, sector 1**. Aquí se almacena la **tabla de particiones** de todo el disco duro marcando así el comienzo y el final de las mismas. También se almacena el **Bootloader**, en donde se recoge la partición activa en donde se instala el sistema o los sistemas operativos. En la actualidad se ha sustituido casi en todos los casos por el estilo de particiones **GPT**, que ahora veremos con más detalle.

Structure of a Master Boot Record (MBR)



COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Nombre del bootloader	Descripción
Bootmgr	Gestor de arranque de Microsoft desde Windows Vista / Windows Server 2008
NT-Loader (NTLDR)	Gestor de arranque de Microsoft hasta Windows XP / Windows Server 2003
barebox	Bootloader para sistemas integrados en impresoras, cámaras, coches, aviones, etc.
boot.efi	Bootloader EFI utilizado desde 2006 en dispositivos Mac
BootX	Antiguo bootloader de sistemas operativos Mac
Grand Unified Bootloader (GRUB)	Bootloader libre para sistemas operativos unixoides como Linux
ARM Core Bootloader	Bootloader para microcontroladores (usado, por ejemplo, en iPhones)
OpenBIOS	Gestor de arranque libre y portátil con licencia GNU-GPL

Diferencia entre MBR y GPT

Particiones

En la actualidad hay dos tipos de tablas de particiones disponibles para un HDD o SSD, las de tipo **MBR** o las de tipo **GPT** (Global Unique Identifier). El estilo de particiones GPT fue implementado para los sistemas EFI o Extensible Firmware Interface, que ha sustituido al antiguo sistema BIOS de los ordenadores. Entonces, mientras que BIOS utiliza MBR para gestionar el disco duro GPT está orientado a ser el sistema propio para UEFI.

Nota: En realidad la propia BIOS no necesita saber exactamente cómo está particionado un disco. Solo asegura que el **bootloader** del MBR se cargue en la memoria y se ejecute.

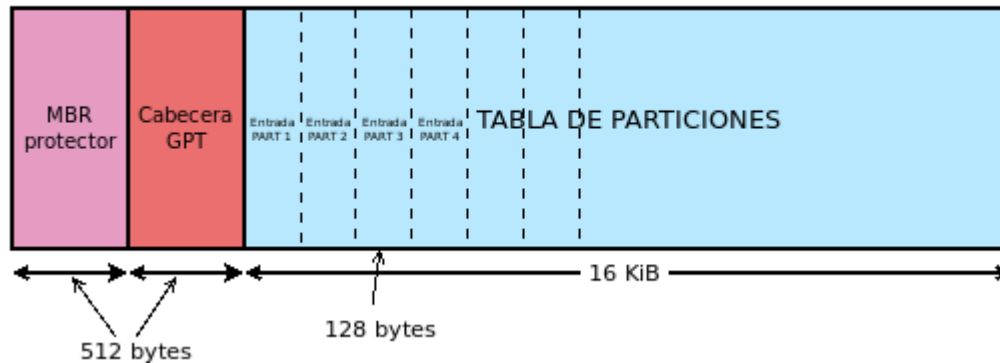
Sistema de arranque

Mientras que el MBR comienza con el código de arranque maestro (Master Boot Code), que contiene un binario ejecutable que identifica la partición activa e inicia el proceso de arranque, la GPT se basa en las capacidades extendidas del EFI para estos procesos. A pesar de que una entrada de MBR comienza el disco, con propósitos de protectividad y compatibilidad con el viejo esquema BIOS PC, la GPT propiamente dicha comienza con la cabecera de la tabla de particiones (el MBR protector de un disco GPT cumple la función de evitar que las herramientas que sólo son compatibles con los discos MBR reconozcan y sobrescriban los discos GPT).

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

TABLA DE PARTICIONES GPT



Lo mejor de todo es que este sistema asigna un GUID único a cada partición, es como si fuera una dirección MAC, y el asignador es tan largo que se podrían nombrar de forma única todas las particiones del mundo, lo que elimina prácticamente las limitaciones físicas de un disco duro en cuanto a particionado.

Administrador: Windows PowerShell

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

PS C:\Windows\system32> diskpart

Microsoft DiskPart versión 10.0.17134.1

Copyright (C) Microsoft Corporation.
En el equipo: W10VB

DISKPART> list disk

Núm Disco Estado Tamaño Disp Din Gpt
-----
Disco 0 En línea 50 GB 0 B *
```

En teoría, si quieres particionar algún medio de almacenamiento, todavía puedes elegir entre MBR o GPT. Sin embargo, la **tabla de particiones GUID** se ha convertido en el **estándar del hardware moderno**, ya que admite un número ilimitado de particiones y tampoco presenta inconvenientes (al menos ninguno que parezca relevante en un futuro cercano) con respecto a la capacidad de los medios extraíbles y los sectores de almacenamiento. En los **discos duros con más de dos terabytes de memoria**, la GPT es la única alternativa. El uso del antiguo estilo de partición **MBR** solo se

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

recomienda si se trabaja con hardware antiguo y versiones anteriores de Windows, así como en muchos sistemas operativos de **32 bits**, tanto antiguos como nuevos.

1.2 Conceptos básicos

La información se guarda en los dispositivos de almacenamiento mediante archivos, que son grupos de informaciones relacionadas. Cada sistema operativo utiliza el sistema de archivos para gestionar cómo se guardan los archivos y conocer información importantes de ellos como su nombre, el tamaño, el tipo de información que contienen, la fecha de grabación, el sitio físico donde se encuentra dentro del soporte, etc.

Cada sistema de archivos organiza la información de forma diferente pero en todos ellos existen 3 grandes tipos de archivos:

- archivos regulares o estándares: son los archivos normales que contienen información (programas, documentos, imágenes, etc)
- directorios o carpetas : son archivos especiales que contienen información de los archivos que hay en el directorio y se utilizan para organizar la información dentro del soporte
- archivos especiales: se usan para gestionar la entrada/salida de archivos regulares hacia o desde los periféricos .

En todos los sistemas de archivos la información se organiza de forma jerárquica en una estructura de árbol: cada dispositivo tiene un directorio raíz dentro del cual se encuentran archivos y otros subdirectorios, dentro de los cuales se encuentran más archivos y subdirectorios.



1.2.1 Nombre de un archivo

Un archivo es el objeto encargado de contener la información. Sus características más importantes son el nombre y su ruta (path), ya que ambos lo identifican de forma unívoca entre todos los demás archivos del sistema informático.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

El aspecto y formato del nombre depende del sistema de archivos que utilicemos. Las primeras computadoras sólo permitían unas pocas letras o dígitos en el nombre de un archivo, pero las computadoras modernas permiten nombres largos que contengan casi cualquier combinación de símbolos UNICODE.

Algunos sistemas informáticos permiten nombres de archivo que contengan espacios, otros no. La distinción entre mayúsculas y minúsculas en los nombres de archivo está determinada por los sistemas de archivos. Los sistemas de archivos Unix y GNU/Linux distinguen normalmente entre mayúsculas y minúsculas, y permiten a las aplicaciones a nivel de usuario crear archivos con nombres que difieren sólo en si los caracteres están en mayúsculas o minúsculas. Microsoft Windows normalmente no distingue entre mayúsculas y minúsculas.

Las reglas para nombrar los archivos las determina el sistema de archivos que tenga el soporte en que se guardará, por ejemplo si el sistema de archivos es una de las primeras versiones de FAT el nombre sólo podrá tener 8 caracteres más 3 para la extensión.

La extensión de un archivo va al final del mismo, detrás de un punto y son 3 caracteres (aunque actualmente pueden ser más o menos) que permiten a los sistemas operativos Windows conocer el tipo de información que contiene un archivo.

NombreDelArchivo.extensión

GNU/Linux no utiliza la extensión sino los metadatos para saber el tipo de información de un archivo. Los metadatos son atributos asociados a un archivo pero que no forman parte de él (ej: la fecha o el autor). Aún así solemos utilizar las extensiones porque a nosotros sí que nos aporta esa información.

Algunos ejemplos de extensiones comunes son:

- exe: archivo que contiene un programa ejecutable en Windows
- txt: archivo que contiene texto plano (sin formato)
- doc: archivo con un documento de Microsoft Word
- odt: archivo con un documento de LibreOffice Writer
- bmp, gif, jpg, jpeg, png, tif, ...: archivo con una imagen, con diferente formato
- wma, mp3, ...: archivo con sonido.
- mp4, avi, mpeg, ...: archivo de vídeo

Además de su nombre (y extensión) para cada archivo el sistema de archivos guarda una serie de atributos del mismo. Algunos de los más comunes son:

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

- Fecha: fecha en que el archivo fue creado o modificado
- Hora: hora en que el archivo fue creado o modificado
- Tamaño: tamaño en bytes del archivo
- Propietario y grupo propietario (en sistemas GNU/Linux): indica a qué usuario y a qué grupo pertenece ese archivo

En sistemas de archivos de Microsoft (FAT y NTFS) también son importantes los siguientes atributos:

- S: atributo del sistema (System). Indica que se trata de un archivo del sistema operativo
- H: atributo de oculto (Hidden). Indica que el archivo está oculto y no aparecerá al mostrar el contenido de su directorio
- R: atributo de sólo lectura (Read only). Indica que se puede abrir el archivo y ver su contenido pero no guardar modificaciones en el mismo
- A: atributo de archivo accedido. Se activa cuando se modifica un archivo y se utiliza para hacer copias de seguridad de sólo los archivos modificados

1.2.2 Los directorios

Todos los sistemas de archivos cuentan con un tipo especial de archivo llamado directorio que nos permite organizar todos los archivos que hay dentro de un determinado soporte. Un directorio puede contener archivos y otros subdirectorios.

De esta forma los archivos y directorios constituyen una estructura jerárquica en forma de árbol: cada soporte tiene un directorio principal llamado directorio raíz dentro del cual se encuentran archivos y otros subdirectorios que contienen más archivos y subdirectorios.

Como hemos dicho antes, un directorio es un caso especial de archivo y los datos que contiene es la información de cada archivo y subdirectorio que se encuentra dentro de él (su nombre y atributos y dónde se encuentra en el disco).

Se llama directorio padre de un directorio o archivo al directorio que contiene ese directorio o archivo. El directorio padre se representa con .. (dos puntos seguidos). El directorio donde se encuentra (llamado directorio actual) se representa con . (un punto).



COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

1.2.2.1 Directorio raíz

Es el primer directorio o carpeta en una jerarquía.
Contiene todos los subdirectorios de la jerarquía.

En sistemas tipo Unix, se identifica con el carácter */*. Todos los accesos al sistema de archivos, incluyendo los otros discos y particiones, se encuentran en la jerarquía dentro del directorio raíz.

En los sistemas operativos DOS y Windows cada partición tiene un directorio raíz individual (nombrado **letra:**, por ejemplo para una determinada partición C su directorio raíz se llama C:\) y no hay un directorio raíz común que los contenga todos ellos como en los sistemas Unix.

1.2.3 Rutas de los archivos

La ubicación de un archivo se indica con una cadena de texto llamada "**ruta**" (path en inglés). Esta ruta nos dice dónde podemos encontrar ese archivo, empezando desde el directorio raíz y terminando en el nombre del archivo. En realidad lo que identifica a un archivo no es sólo su nombre, sino su ruta entera. De hecho podemos tener diferentes archivos con el mismo nombre pero siempre deben estar en diferente directorio (y por tanto tendrán diferente ruta).

La nomenclatura para rutas varía ligeramente de sistema en sistema, pero mantienen por lo general una misma estructura. Una ruta viene dada por una sucesión de nombres de directorios y subdirectorios, ordenados jerárquicamente de izquierda a derecha y separados por algún carácter especial que suele ser una barra (*/*) o barra invertida (**) y termina con el nombre de un archivo o directorio presente en la última rama de directorios especificada.

Ejemplo de 'ruta' en un sistema Unix

Así, por ejemplo, en un sistema tipo Unix como GNU/Linux, la ruta para la canción llamada "canco.ogg" del usuario "pere" sería algo como:

`/home/pere/musica/canco.ogg`

donde:

- */* representa el directorio raíz donde está montado todo el sistema de archivos.
- *'home/pere/musica/'* es la ruta del archivo.
- *'canco.ogg'* es el nombre del archivo.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Ejemplo de 'ruta' en un sistema Windows

Un ejemplo análogo en un sistema de archivos de Windows se vería como:

C:\Usuarios\pere\Mis Documentos\Mi música\cancion.ogg

donde :

- 'C:' es la unidad de almacenamiento en la que se encuentra el archivo
- '\' representa el directorio raíz de la unidad C:
- 'Usuarios\pere\Mis Documentos\Mi música ' es la ruta del archivo.
- 'cancion' es el nombre del archivo.
- '.ogg' es la extensión del archivo, este elemento, parte del nombre, es especialmente relevante en los sistemas Windows, ya que sirve para identificar qué aplicación está asociada con el archivo en cuestión, es decir, con qué programa se puede editar o reproducir el archivo.

Rutas absolutas y relativas

El directorio en el que nos encontramos en un momento dado se llama **directorio activo**. Siempre hay 2 formas de identificar un archivo:

- utilizando su ruta absoluta: es la que comienza en el directorio raíz y llega hasta el archivo. Los ejemplos que hemos visto anteriormente usan rutas absolutas
- utilizando rutas relativas: en lugar de empezar la ruta por el directorio raíz empieza desde el directorio activo.

Por ejemplo en el caso anterior si el directorio activo es /home/pere la ruta en el archivo canco.ogg sería:

- ruta absoluta: /home/pere/musica/canco.ogg
- ruta relativa: musica/canco.ogg

Hay dos nombres de directorio especial que se encuentran en todos los directorios:

- . (punto): hace referencia a ese directorio
- .. (punto punto): hace referencia al directorio padre, o sea al directorio dentro del cual se encuentra este directorio.

En el ejemplo anterior si el directorio activo es /home/marc/documents la ruta relativa a canco.ogg sería:

..\..\pere\musica\canco.ogg

donde:

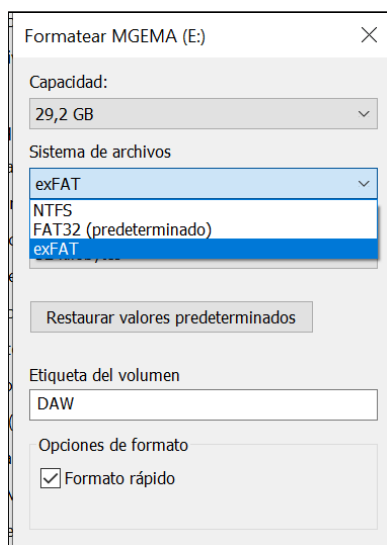
COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

- ..\ nos lleva al directorio padre del activo (documentos), o sea, a /home/marc
- ...\ nos lleva al directorio padre de “marc”, o sea, a /home
- pere\musica\ nos lleva al directorio musica dentro de pere dentro del anterior (/home)
- canco.ogg hace referencia al archivo

2 - Sistemas de archivos

En un sistema informático la información se guarda en forma de archivos o ficheros y la forma en que se guardan esos archivos en cada soporte de almacenamiento la determina el sistema de archivos del mismo. El sistema de archivos es quien establece la estructura, nombre, forma de acceso, uso y protección que tendrá cada archivo almacenado. El tipo de sistema de archivos de un dispositivo se determina al darle formato.



Además de alojar los datos de los archivos, el sistema de archivos también almacena y manipula información muy importante sobre los archivos y el propio sistema de archivos (fecha y hora, propietario, permisos de acceso, tamaño de los archivos, localización o localizaciones del almacenaje en el disco, etc). Esta información constituye lo que comúnmente denominamos **metadatos** que son atributos asociados a un archivo pero que no forman parte de él.

La mayoría de los sistemas operativos utilizan su propio sistema de archivos (NTFS, ext4, APFS,...).

Lo habitual es utilizar dispositivos de almacenamiento de datos que permiten el acceso a los datos como una cadena de bloques de un mismo tamaño, llamados sectores, normalmente de 512 bytes de longitud. El software del sistema de archivos es responsable de la organización de estos sectores en archivos y directorios y mantiene un registro de qué sectores pertenecen a qué archivos y cuáles no han sido utilizados.

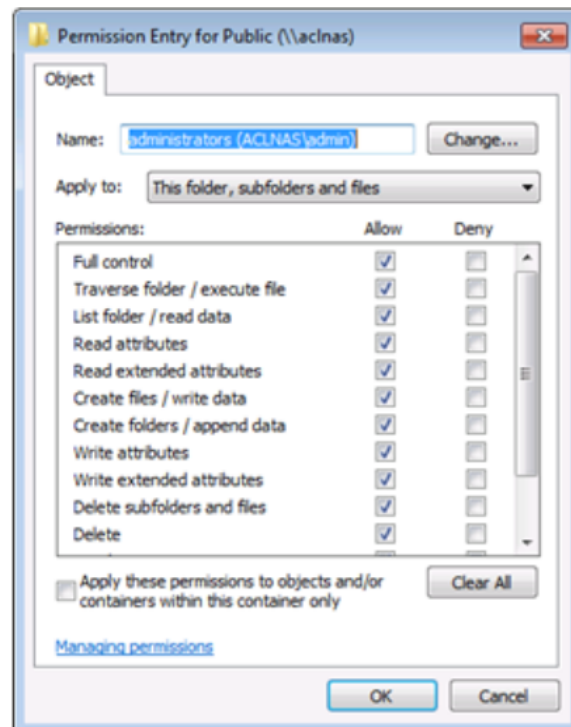
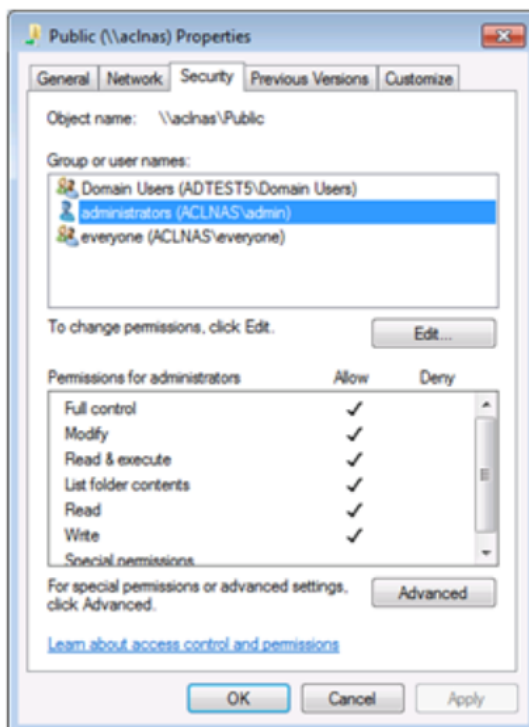
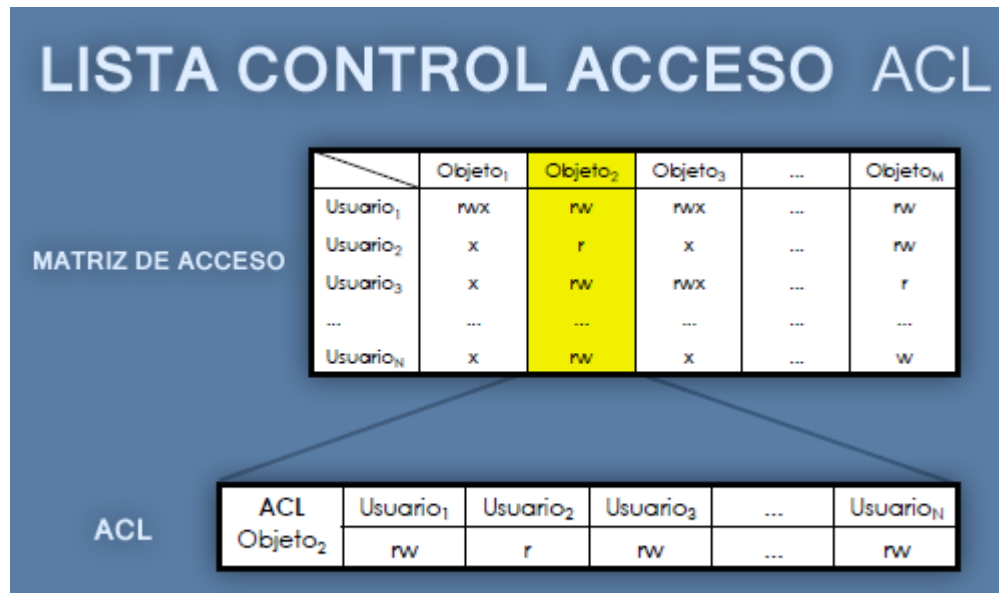
Los sistemas de archivos proveen métodos para crear, copiar y eliminar tanto archivos como directorios.

Respecto a la protección de los datos el acceso seguro a sistemas de archivos básicos puede estar basado en los esquemas de **lista de control de acceso** (ACLs) o en **capacidades**. En ambos casos se define una matriz de acceso que permite determinar para cada objeto (archivos, partes de archivos, programas, segmentos de memoria, ...)

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

y sujeto o proceso que accede al mismo qué acciones le está permitido llevar a cabo.



COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Nota: Las listas de control de acceso hace décadas que demostraron ser inseguras, por lo que los sistemas operativos experimentales utilizan el acceso por capacidades. Los sistemas operativos comerciales aún funcionan con listas de control de acceso.

2.1 Características y funcionalidades

Cuando elegimos un sistema de archivos para formatear un soporte debemos tener en cuenta sus características y las funcionalidades que proporciona. Las más importantes a tener en cuenta son:

- El tamaño máximo que puede tener un archivo.
- La máxima capacidad del sistema de archivos (es decir, el tamaño máximo de la partición).
- Cómo gestiona los permisos de archivos y directorios (es decir, quién puede acceder a cada archivo o directorio y qué puede hacer con él). Hay varias opciones:
 - Sin permisos (cualquier usuario puede hacer cualquier cosa con el archivo, ej: FAT)
 - Listas de control de acceso (ACLs). Cada archivo o directorio tiene asociada una lista de qué usuarios pueden hacer qué sobre ese archivo (ej: NTFS)
 - UGO (Usuario, Grupo, Otros). Se establecen los permisos para el propietario del archivo o directorio, el grupo propietario y el resto de usuarios (ej: los sistemas de archivo *ext* de GNU/Linux)
 - Capacidades granuladas. Son la mejor forma de establecer permisos aunque actualmente no está soportada por ninguno de los sistemas de archivos utilizados normalmente.
 - Atributos extendidos (sólo añadir al archivo pero no modificar, no modificar nunca, etcétera).
- Cómo evita la fragmentación.
- Si permite enlaces, tanto simbólicos como duros.
- Si tiene Journaling (recuperación rápida después de un apagado brusco).
- Si proporciona soporte para [archivos dispersos](#).
- Si proporciona soporte para cuotas de disco.
- Si proporciona soporte del crecimiento del sistema de archivos nativo.

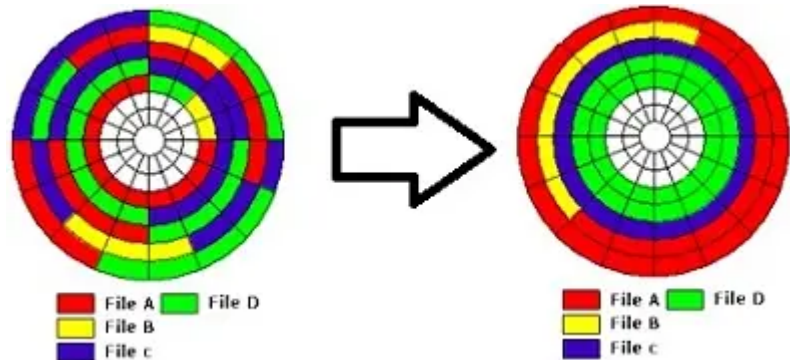
COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

2.1.1 Fragmentación del disco

Al ir escribiendo y borrando archivos continuamente en el disco duro, éstos tienden a no quedar en áreas contiguas. Así los diferentes clusters que forman el archivo quedan repartidos por todo el disco, y se dice entonces que el archivo está "fragmentado". Al tener los archivos esparcidos por el disco, se vuelve ineficiente el acceso a ellos.

La fragmentación se da muy comúnmente en el sistema operativo Windows aunque también afecta a otras plataformas pero en una escala mucho menor. También se produce fragmentación dentro de la memoria del computador (memoria RAM) cuando se asignan a los procesos los distintos bloques de memoria.



Desfragmentación

La desfragmentación es el proceso mediante el cual se acomodan los archivos de un disco de modo que cada uno quede en un área contigua y sin espacios sin usar entre ellos.

Este proceso consta de ordenar los pedazos de información distribuida a través de todo el disco para mejorar la velocidad de acceso y distribuir de mejor forma el espacio libre del dispositivo. Como este proceso consta en la reorganización de partes de archivos, requiere de suficiente memoria para realizar los movimientos de los pedazos de información. Al mover en forma física la información, la estructura lógica no sufre alteraciones.

Ciertos sistemas de archivos son más susceptibles a cambios por desfragmentación que otros. Por ejemplo, una partición tipo FAT se fragmenta más rápido que una partición tipo NTFS, ambos propios del sistema Windows.

A diferencia de los discos duros (cuyo cabezal se mueve físicamente a varias posiciones para acceder a los archivos), los **SSD tienen un acceso aleatorio**. Es decir, en lugar de tener que pasar de la posición 1 a la 10 recorriendo todas las posiciones, los SSD acceden directamente a la posición 10. Por ello, aunque un archivo esté dividido en varias partes más pequeñas, esto no afecta al rendimiento del ordenador, ya que los tiempos de acceso son los mismos que si estuviera todo seguido. Por tanto, a los SSD no les afecta la fragmentación y no hay necesidad de

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

desfragmentarlos.

2.1.2 Sistemas de archivos transaccionales: *Journaling*

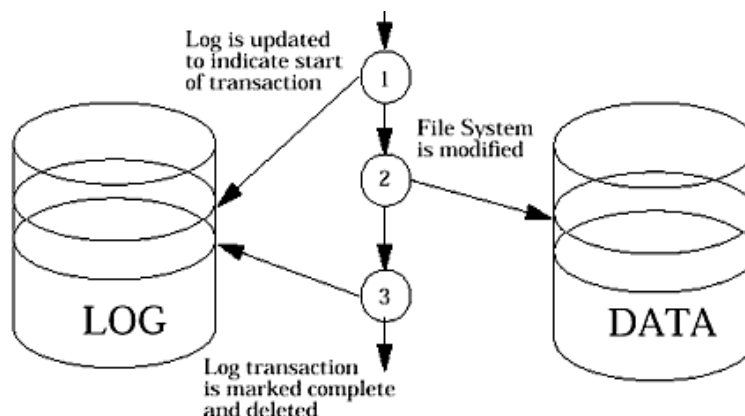
Los sistemas operativos y más concretamente los sistemas de archivo procuran que todo lo que se lee de un disco sea lo mismo que se escribió en él. Esto que parece obvio, cuando se implementa resulta que no lo es tanto.

Los sistemas de archivos no escriben directamente en el disco, para agilizar escriben en un buffer y luego el contenido se actualiza en el disco más tarde. Antiguamente no podías apagar un Linux desenchufando el equipo o dándole al botón porque los sistemas de archivos como EXT2 de Linux carecían de journaling.

Actualmente los sistemas de archivos además de journaling pueden gestionar particiones más grandes, reducen la fragmentación, tienen mejor rendimiento de entrada/salida y se recuperan mejor ante posibles accidentes.

Problema de un sistema de ficheros sin journaling.

Las escrituras en disco como se ha explicado se realizan en varios pasos. Si apagas el equipo en medio de algún paso los datos se quedan inconsistentes. Para detectar y reparar todos los problemas de integridad existentes habrá que correr el famoso **fsck** de Linux. Dado que el tiempo que tarda fsck es proporcional al tamaño de la partición y al número de ficheros y directorios, para los sistemas actuales es imprescindible utilizar un sistema de journaling.



Para realizar operaciones en un sistema de ficheros se utiliza un mecanismo parecido al que utilizan las bases de datos, las transacciones. Una transacción es un conjunto de operaciones que se realizan de forma unitaria (o se completan del todo o no se realiza

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

nada). Es algo parecido a lo que utiliza tu banco cuando quieres mover dinero entre dos de tus cuentas, el sistema retira una cantidad de dinero de una y la abona en la cuenta de destino. Si una vez retirado el dinero de la primera cuenta se produce un fallo eléctrico, la transacción no se habrá completado y el sistema restablecerá el saldo de las cuentas a los valores existentes antes de la transacción.

Cuando un sistema de ficheros con journaling arranca no revisa y verifica los metadatos del mismo, sino que tiene que revisar solamente aquellas transacciones que no se llevaron a cabo y las ejecuta. De esa manera se asegura que la información del disco y los metadatos están sincronizados y son veraces.

El journal en un sistema de ficheros consiste básicamente en una lista de transacciones, de esta manera las recuperaciones ante posibles accidentes son mucho más ágiles que en sistemas sin journaling.

Existen muchos sistemas de ficheros con journaling como por ejemplo EXT4, NTFS, ReiserFS, JFS, etc.

2.2 Otras características de los sistemas de archivos

2.2.1 Sistemas distribuidos

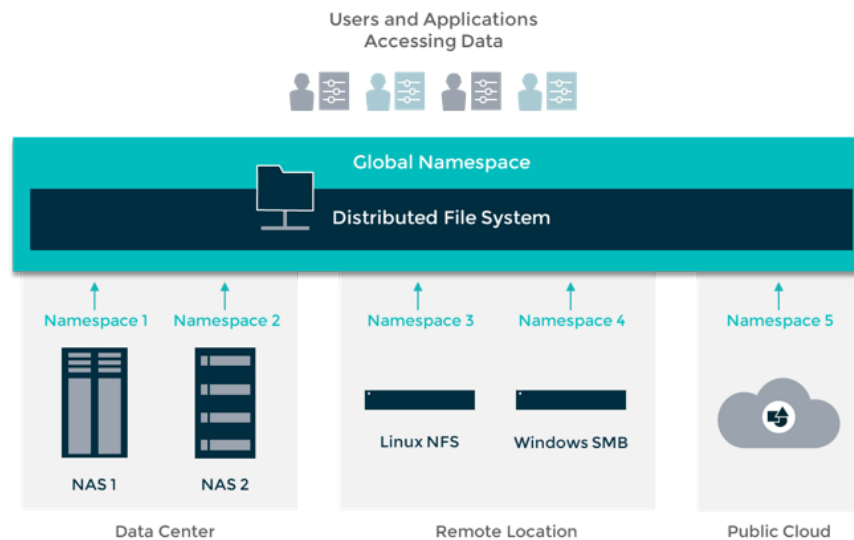
Un **sistema de archivos distribuido** o **sistema de archivos de red** es un sistema de archivos que sirve para compartir archivos, impresoras y otros recursos en una red de computadoras. Algunos ejemplos son **NFS (Network File System)** de Gnu/Linux o **SMB (Server Message Block)** de Microsoft.

Los sistemas de archivos distribuidos permiten a los programas almacenar y acceder a archivos remotos al igual que si fueran locales.

Un servidor de archivos es un proceso que se ejecuta en alguna máquina y permite implantar el servicio de archivo. Un sistema puede tener uno o varios servidores de archivos, cada uno de los cuales ofrece un servicio de archivos diferentes, pero los clientes no es necesario que conozcan cuántos servidores de archivos existen, su posición o función, ni siquiera deben saber que el servicio de archivos es distribuido sino que cuando tienen montado en su máquina el sistema de archivos remoto, pueden verlo como un sistema de archivos local.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF



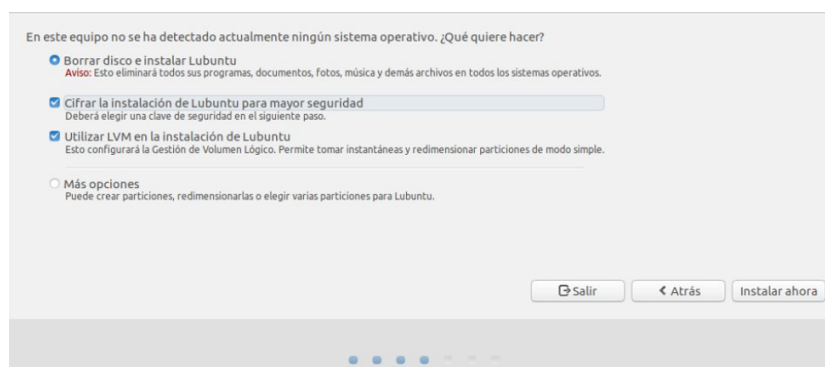
2.2.2 Sistemas cifrados

Un sistema de archivos cifrado permite almacenar información en el disco duro en formato cifrado para evitar que pueda accederse a la información incluso accediendo físicamente al disco.

Estos sistemas de archivos permiten que se encripten de manera transparente para el usuario todos o algunos de los archivos al guardarse en el disco y se desencripten cuando el usuario abra el archivo para trabajar con él.

Existen multitud de aplicaciones para encriptar particiones como LUKS o TrueCrypt y trabajan sobre sistemas de archivo ext4 o ext3.

En Windows la aplicación se llama EFS (Encrypt File System) y funciona sobre NTFS.



2.2.3 Sistemas comprimidos

Los sistemas de archivos modernos también permiten la compresión de los archivos

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

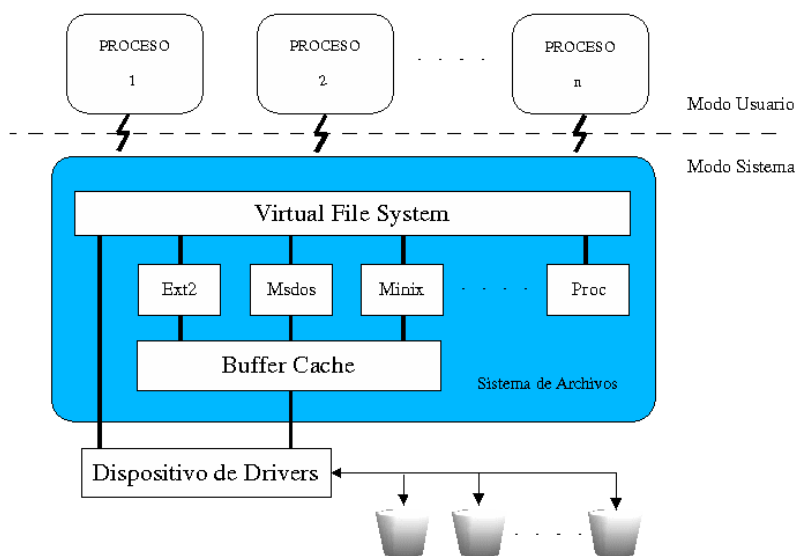
indicados de forma transparente para el usuario. De todas formas la capacidad cada vez mayor de los discos duros y los requerimientos de CPU y memoria para el proceso de compresión y descompresión hacen que esta característica no se utilice mucho. Sólo tiene sentido con archivos que no se utilizan habitualmente.

2.2.4 Sistemas virtuales

Un sistema de archivos virtual o **VFS**, es un tipo de software que es parte de la mayoría de los sistemas operativos.

El propósito de un VFS es permitir que las aplicaciones cliente tengan acceso a varios tipos de sistemas de archivos de un modo uniforme (por ejemplo para obviar las diferencias en los sistemas de archivos de Windows, de Mac y Unix), de modo que las aplicaciones pudieran tener acceso a archivos en los sistemas de archivos locales sin tener que saber a qué tipo de sistema de archivos están teniendo acceso.

El virtual **filesystem** de Linux con un solo directorio raíz, permite acceder a los archivos de los discos que estén montados, pero también permite acceder a dispositivos, y archivos virtuales que literalmente no existen en ningún disco, pero son modelados como archivos.



COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

2.3 Principales sistemas de archivos

Nombre	Uso	Sistema operativo (compatibilidad)	Características
FAT32	Medios de almacenamiento extraíbles	<ul style="list-style-type: none"> - Windows - Mac OS X/macOS - Linux (si se instalan los correspondientes controladores) 	<ul style="list-style-type: none"> - Alta compatibilidad - Compatible con muchos tipos de <i>hardware</i> - Sin funciones de cifrado ni compresión - No garantiza particularmente la seguridad de los datos - Ideal para particiones más pequeñas - Volumen máximo de datos: 4 GB

exFAT	Medios de almacenamiento extraíbles	<ul style="list-style-type: none"> - Windows - Mac OS X/macOS (compatible a partir de 10.6.4) - Linux (si se instalan los correspondientes controladores) 	<ul style="list-style-type: none"> - Aún no es un estándar generalizado - No permite gestionar derechos - No permite comprimir los datos - Ideal para memorias <i>flash</i> más pequeñas, a partir de 32 GB (memorias USB, tarjetas SD) - Tamaños y particiones ilimitados (según el estado actual de la tecnología) - Volumen máximo de datos: 512 TB
--------------	-------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

NTFS	Discos duros internos y externos	<ul style="list-style-type: none"> - Windows - Mac OS X/macOS (de forma integral instalando un <i>software</i> adicional) - Linux (instalando controladores) 	<ul style="list-style-type: none"> - Gestión de derechos - Mejora de la seguridad de los datos: protege contra la pérdida y la modificación de los datos; permite el cifrado - Permite comprimir los datos; alto rendimiento con medios de almacenamiento grandes - Se especializa en archivos extensos y en grandes capacidades de almacenamiento - Inadecuado para discos pequeños y particiones de menos de 400 MB (demasiada potencia) - Volumen máximo de datos: 256 TB
-------------	-------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

APFS	Unidades SSD	<ul style="list-style-type: none"> - macOS (el estándar desde la versión 10.13 High Sierra) - Versiones anteriores de Mac OS y Windows (instalando un <i>software</i> adicional) 	<ul style="list-style-type: none"> - Optimizado para unidades de estado sólido (SSD) y otros dispositivos de almacenamiento <i>flash</i> - También funciona en unidades mecánicas e híbridas - Permite el cifrado de datos - Optimiza la gestión del espacio de almacenamiento (función de espacio compartido) - Función de protección contra bloqueos, que protege contra daños en el sistema de archivos (por ejemplo, en caso de caída del sistema) - Compatible con Fusion Drive desde macOS 10.14 Mojave - Volumen máximo de datos: 8 exbibytes
-------------	--------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

HFS+	Discos duros internos y externos	Mac OS X/macOS	<ul style="list-style-type: none"> - Sistema de archivos maduro y probado - Especialmente indicado para discos mecánicos - No optimizado para tecnologías de almacenamiento modernas (SSD, <i>flash</i>) - Mejor compatibilidad con versiones anteriores que APFS - Vida útil limitada; probablemente deje de ser compatible con Apple a largo plazo - Perderá importancia progresivamente debido a la “conversión forzada” y parcialmente automatizada a APFS - Volumen máximo de datos: 8 exbibytes
-------------	-------------------------------------	----------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

ext4	Linux	<ul style="list-style-type: none"> - Linux - Windows (solo con <i>software</i> adicional) - Mac OS X/macOS (solo con <i>software</i> adicional) 	<p>En comparación con versiones anteriores de ext:</p> <ul style="list-style-type: none"> - Mejora del rendimiento - Mejora de la seguridad de los datos - Incorpora cifrado (desde Linux Kernel 4.1) - La nueva función <i>extents</i> aumenta la velocidad de procesamiento de archivos grandes y evita la fragmentación - Gestión de derechos - Volumen máximo de datos: 16 TB
-------------	-------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

¿Cuándo es más apropiado cada sistema de archivos?

Si tienes un USB en el que vas a llevar documentos o archivos multimedia pocos pesados lo mejor es formatear tu unidad con el sistema FAT32.

Si compartes archivos de más de 4 GB con equipos de sistemas operativos que tienen Windows, GNU/Linux y macOS, la mejor opción es formatear en formato exFAT.

Si solo utilizas Windows y necesitas pasar archivos pesados entre ordenadores, copias de seguridad de los archivos multimedia o ver un vídeo pesado tendrías que formatear en NTFS.

Si utilizas dispositivos de Apple y quieres pasar archivos pesados de un ordenador a otro o realizar copias de seguridad formatear en HFS+. Sabiendo que no lo vas a usar en equipos con Windows.

Y por último, si tienes pensado hacer copias de seguridad y compartir archivos entre sistemas con GNU/Linux, la mejor opción es formatear en Ext4. Recuerda que este formato no se puede usar en Windows o Mac

3 Gestión de sistemas de archivos

Los sistemas operativos incluyen herramientas para gestionar los sistemas de archivos soportados por ellos. Esta gestión incluye las operaciones más habituales como creación, eliminación, copia, etc de archivos y directorios, así como operaciones sobre todo el sistema de archivos como comprobación, desfragmentación, etc.

Un sistema de gestión de archivos es el software que proporciona a los usuarios y aplicaciones servicios para el uso, acceso y control de accesos, tanto de archivos como de directorios.

Algunos de los objetivos de la gestión del sistema de archivos es:

- Garantizar que la información del archivo sea válida.
- Optimizar el acceso a los archivos.
- Proveer soporte E/S a una gran variedad de dispositivos de almacenamiento.
- Entregar los datos que el usuario solicita.
- Minimizar o eliminar una potencial pérdida de datos.
- Proveer un conjunto estándar de rutinas E/S.
- Proveer soporte de E/S a múltiples usuarios.

Entre las principales funciones de la gestión de archivos, podríamos destacar las

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

siguientes:

- Identificar y localizar un archivo.
- Usar un directorio para describir la ubicación y atributos de un archivo.
- Controlar el acceso de varios usuarios a los archivos.
- Bloquear el uso de archivos.
- Ubicar archivos en bloques libres.
- Administrar el espacio libre.

3.1 Montaje de los sistemas de archivos

Para que el sistema operativo pueda utilizar un dispositivo y acceder a su sistema de archivos debe primero montarlo. Este proceso es normalmente automático y asocia ese dispositivo a una ruta (una letra en Windows o un directorio en Gnu/Linux).

3.1.1 Windows

Windows monta automáticamente todos los dispositivos que encuentre y que tengan un sistema de archivos soportado por el sistema, es decir, FAT o NTFS.

A cada partición se le asignará una ruta formada por una letra y 2 puntos. La primera partición que detecta es normalmente en la que está instalado Windows y le asigna la primera letra, C: (las letras A: y B: estaban reservadas para unidades de disquete). Al resto de dispositivos le va dando letras a continuación (D:, E:, ...). En el caso de dispositivos extraíbles como USB hace el mismo proceso y les asigna la siguiente letra disponible.

Podemos modificar la letra asignada a cada dispositivo en Windows desde el Administrador de discos.

3.1.2 Gnu/Linux

En Linux, UNIX y sistemas operativos similares, los sistemas de archivos en diferentes particiones y dispositivos extraíbles (CDs, DVDs o memorias USB, por ejemplo) se pueden montar en un punto determinado (el punto de montaje) en el árbol de directorios, y luego desprenderse de nuevo. Mientras un sistema de archivos está montado en un directorio, el contenido original del mismo no es accesible.

3.2 Gestión desde el entorno gráfico

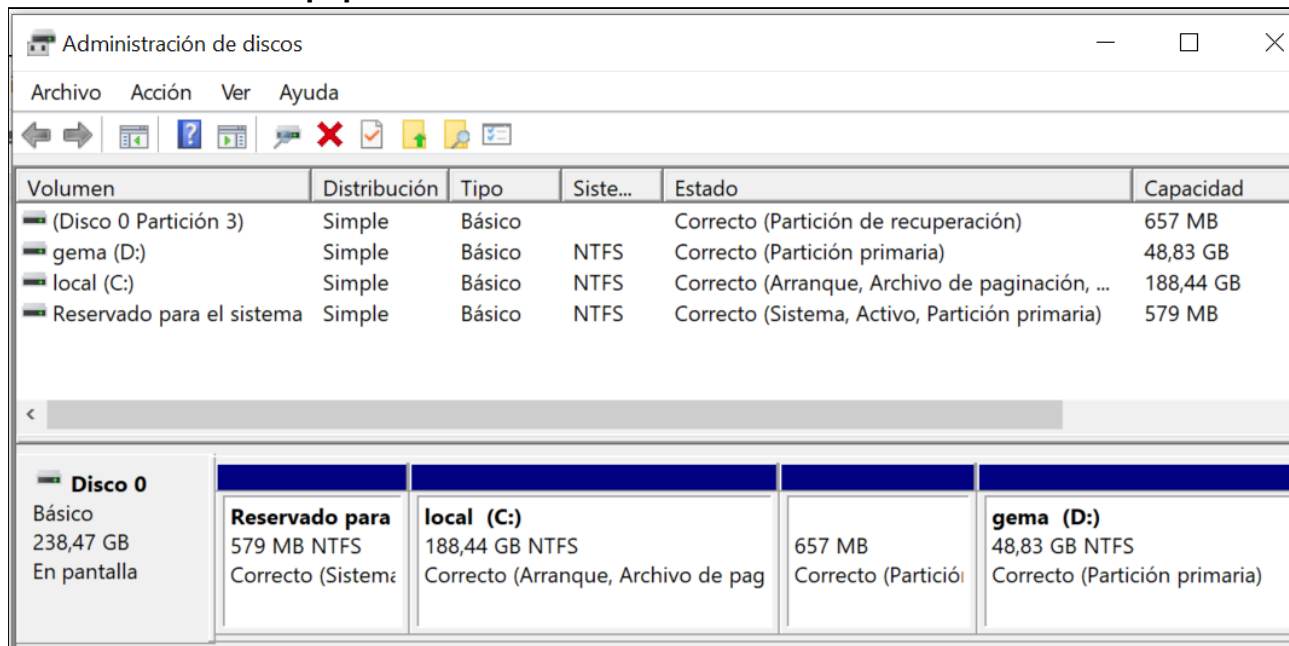
3.2.1 Windows

La principal herramienta gráfica en Windows para gestionar los sistemas de archivos es el **Administrador de disco (diskmgmt.msc)** que encontramos dentro del

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Administrador de equipo.

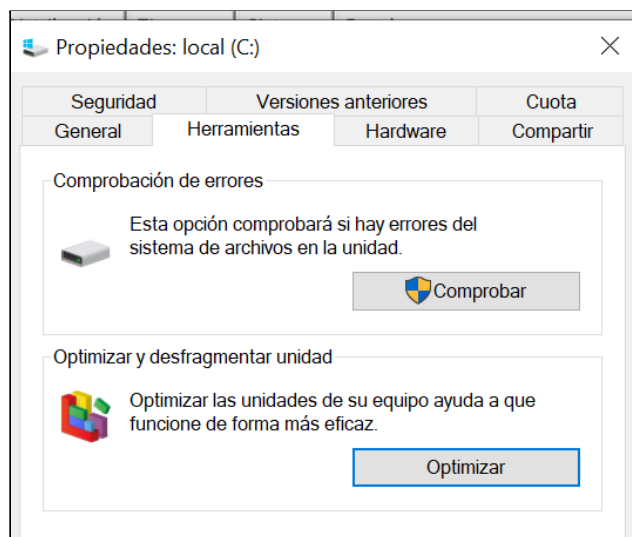


Desde aquí tenemos acceso a todos los discos detectados y podemos gestionar las particiones: crear, eliminar, redimensionar, formatear o cambiar la forma en que Windows monta la partición (podemos cambiar la letra o montarla sobre una carpeta al estilo de Gnu/Linux).

Las particiones con sistemas de archivos que no sean FAT o NTFS aparecen como particiones desconocidas.

También podemos trabajar con volúmenes dinámicos lo que nos permitirá extender particiones por distintos discos o configurar RAID 0,1 o 5 por software (Panel de control / Sistema y seguridad/ Espacios de almacenamiento)

Desde las Propiedades de una partición (menú contextual-> Propiedades) tenemos la pestaña de **Herramientas** para comprobar el sistema de archivos o desfragmentarlo.



También se puede compartir la unidad en la red y, si el sistema de archivos es NTFS, establecer cuotas de disco para distintos usuarios (pestaña Cuota en propiedades del

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

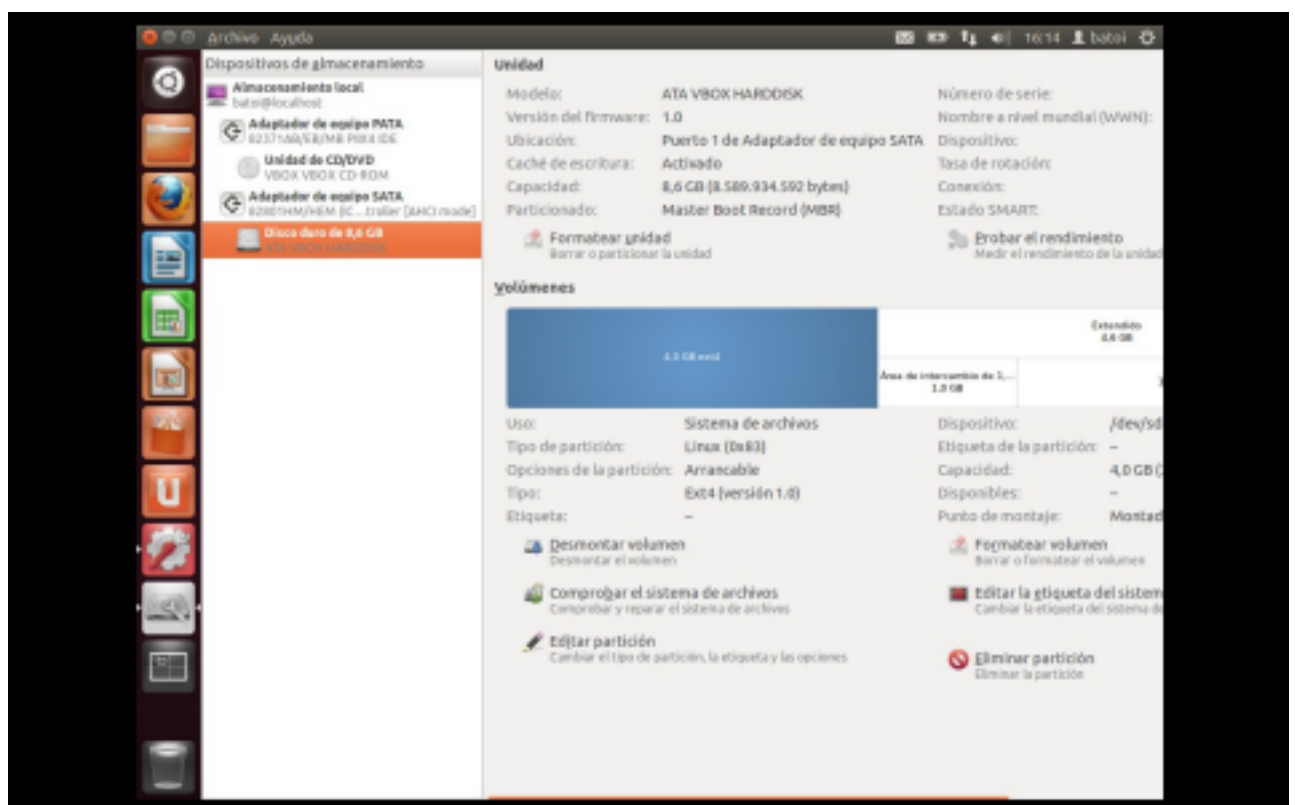
CFGS DAW
DPT INF

disco).

3.2.2 Gnu/Linux

Las diferentes distribuciones y los diferentes escritorios tienen herramientas gráficas propias para gestionar los sistemas de archivo, aunque son bastante similares.

En Ubuntu tenemos la Utilidad de disco que nos permite gestionar las particiones de forma similar a lo que hemos visto en Windows. También podemos instalar el paquete gparted.



3.3 Gestión por medio de órdenes

3.3.1 Windows

No es habitual gestionar los sistemas de archivo en Windows desde la consola pero tenemos varios comandos para hacerlo:

- **chkdsk**: permite comprobar un sistema de archivos FAT o NTFS
- **defrag**: para desfragmentar el sistema de archivos que le indicamos
- **format**: formatea una partición con sistema de archivos FAT o NTFS
- **convert**: convierte una partición FAT a NTFS

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Algunos comandos (como `chkdsk` o `defrag`) deben ejecutarse desde la consola de Administrador. También cuenta con el programa de texto **diskpart** para gestionar particiones (dentro del programa teclear *help* para obtener ayuda).

```
C:\> Administrador: Símbolo del sistema - diskpart
DISKPART> help

Microsoft DiskPart versión 10.0.19041.964

ACTIVE          - Marcar la partición seleccionada como partición activa.
ADD             - Agregar un reflejo de volumen.
ASSIGN          - Asignar una letra de unidad o punto de montaje al volumen
                 seleccionado.
ATTRIBUTES      - Manipular los atributos de volumen o disco.
ATTACH          - Expone un archivo de disco virtual.
AUTOMOUNT       - Habilitar y deshabilitar el montaje automático de los volúmenes básicos.
BREAK           - Separar un conjunto de reflejos.
CLEAN            - Borra la información de configuración, o toda la información del
                 disco.
COMPACT          - Intenta reducir el tamaño físico del archivo.
CONVERT          - Hacer conversiones entre formatos de disco diferentes.
CREATE           - Crear un volumen, una partición o un disco virtual.
DELETE           - Eliminar un objeto.
DETAIL           - Proporcionar detalles sobre un objeto.
DETACH           - Oculta un archivo de disco virtual.
EXIT             - Salir de DiskPart.
EXTEND           - Extender un volumen.
EXPAND           - Expande el tamaño máximo disponible en un disco virtual.
FILESYSTEMS      - Mostrar sistemas de archivos actuales y compatibles del volumen.
FORMAT           - Formatear el volumen o partición.
GPT              - Asignar atributos a la partición GPT seleccionada.
HELP             - Mostrar una lista de comandos.
IMPORT           - Importar un grupo de disco.
TNTACTIVE        - Marcar la partición seleccionada como inactiva
```

3.3.2 Gnu/Linux

Algunos de los comandos para gestionar el sistema de archivos son:

- **fsck**: permite comprobar un sistema de archivos
- **mkfs**: formatea una partición con sistema de archivos ext , FAT, NTFS, etc.
- **mount**: monta una partición en una carpeta para poder utilizarla
- **df**: muestra un resumen de todas las particiones montadas en el sistema
- **dd**: permite copiar directamente al o desde el disco duro

Algunos comandos deben ejecutarse con permisos de Administrador (al abrir el cmd o símbolo del sistema, botón derecho “Ejecutar como administrador”)

También cuenta con el programa de texto **fdisk** para gestionar particiones ms-dos o el más moderno **parted** que permite gestionar tanto particiones ms-dos como GPT.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

```

Terminal
Archivo Editar Ver Buscar Terminal Ayuda
(parted) help
align-check TIPO N                comprueba particiones N para alineaciones TIPO(min|opc)
comprobar NUMERO hace una comprobación simple del sistema de ficheros
cp [DESDE-DISPOSITIVO] DE-NUMERO A-NUMERO copia el sistema de ficheros a otra partición
help [ORDEN] muestra ayuda general, o ayuda sobre ORDEN
mklabel,mktable TIPO-ETIQUETA crea una nueva etiqueta de disco (tabla de particiones)
mkfs NUMERO TIPO-SA              hace un sistema de archivos TIPO-SA en la partición NUMERO
mkpart TIPO-PART [TIPO-SF] INICIO FIN crea una partición
mkpartfs TIPO-PART TIPO-SF INICIO FIN crear una partición con un sistema de ficheros
resizepart NUMBER END            resize partition NUMBER
move NUMERO INICIO FIN           mueve la partición NUMERO
name NUMERO NOMBRE               nombra la partición NUMERO como NOMBRE
print [devices|free|list,all|NUMERO] muestra la tabla de particiones, los dispositivos disponibles, el espacio libre
partición en particular

quit                             sale del programa
rescue INICIO FIN                recupera una partición perdida entre INICIO y FIN
resize NUMERO INICIO FIN         redimensiona la partición NUMERO y su sistema de archivos
rm NUMERO                       elimina la partición NUMERO
select DISPOSITIVO              elige el dispositivo que se editará
set NUMERO BANDERA ESTADO       cambia la BANDERA en la partición NUMERO
toggle [NUMERO [BANDERA]]       cambia el estado de BANDERA en la partición NUMERO
unit UNIDAD                     establece la unidad UNIDAD como predeterminada
version                          muestra el número de versión y la información de copyright de GNU Parted

(parted) print
Modelo: ATA HGST HTS725032A7 (scsi)
Disco /dev/sda: 320GB
Tamaño de sector (lógico/físico): 512B/4096B
Tabla de particiones. gpt

Numero Inicio Fin Tamaño Sistema de archivos Nombre Banderas
1 1049kB 420MB 419MB ntfs Basic data partition oculta, diag
2 420MB 735MB 315MB fat32 EFI system partition arranque
3 735MB 869MB 134MB Microsoft reserved partition msftres
4 869MB 74,1GB 73,3GB ntfs Basic data partition msftdata
7 74,1GB 76,1GB 2000MB linux-swap(v1)
8 76,1GB 108GB 31,4GB ext4 debian
9 108GB 148GB 40,0GB ext4 homedebian
5 148GB 294GB 147GB ntfs Basic data partition msftdata
6 294GB 320GB 25,9GB ntfs Basic data partition oculta, diag

```

4 Estructura de directorios de los sistemas operativos

Por defecto cuando instalamos un sistema operativo se crean una serie de directorios donde se almacenan los archivos del sistema, los programas instalados, etc.

4.1 Directorios en Windows

En los sistemas Microsoft el sistema asigna una letra para cada dispositivo de almacenamiento (partición, disco o unidad extraíble).

Normalmente tendremos una unidad llamada C: en la que tenemos nuestro sistema operativo, además, las recientes versiones de Windows crean una partición reservada para el sistema y otra de recuperación. Es aconsejable crear otras particiones para los datos e información que no tiene que ver directamente con el sistema.

podremos comprobar que tenemos las diferentes unidades o raíces:

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Respecto a los directorios o carpetas principales que tenemos en los sistemas Microsoft:

C:\Windows	Carpeta de Windows
C:\Windows\System32	Carpeta del sistema
C:\Program Files	Archivos de programa
C:\Program Files\Common Files	Archivos de programa, datos comunes
C:\ProgramData	<u>Datos de programas compartidos</u>
C:\Windows\Fonts	Fuentes
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Administrative Tools	Herramientas administrativas compartidas
C:\ProgramData\Microsoft\Windows\Start Menu	Programas en el menú de Inicio
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup	Programas que inician con Windows
C:\Users\Public	<u>Carpetas compartidas (Publicas)</u>
C:\ProgramData\Microsoft\Search\Data\Applications\	<u>Cache de búsquedas hechas en Windows</u>
C:\Users\NombreDeUsuario.	carpeta del perfil del usuario Contiene los directorios: Escritorio, Mi música, Mis documentos, Mis imágenes, Contactos, Favoritos.

4.2 Directorios en GNU/Linux

Un sistema Gnu/Linux reside bajo un árbol jerárquico de directorios muy parecido a la estructura del sistema de archivos de los sistemas Unix.

Originariamente, este árbol de directorios no era 100% estándar, podíamos encontrar diferencias entre unas distribuciones y otras. Esto hizo que diferentes gentes se agruparan y desarrollaran el proyecto FHS (Filesystem Hierarchy Standard) alrededor

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

de 1993, posteriormente se aplicaría también en sistemas Unix (año 1995).

4.2.1 FHS: Filesystem Hierarchy Standard

Se define como un estándar que detalla los nombres, ubicaciones, contenidos y permisos de los archivos y directorios, por tanto, podemos considerarlo como un conjunto de reglas que especifican una estructura y distribución común de los directorios y archivos de los sistemas GNU/Linux.

FHS es un documento guía pero cualquier persona, empresa y/u organización que cree una nueva distribución puede aplicarlo o no. Si éste se aplica a la distribución en cuestión, ésta será más compatible y comprensible para el resto de gente.

Este estándar es bastante flexible, y por tanto existen diferencias en la aplicación de las pautas o normas que marca entre las diferentes distribuciones GNU/Linux.

4.2.2 Todo son archivos

En los sistemas GNU/Linux (y también en Unix), todo son archivos. Tanto el software como el hardware son archivos. Por tanto una impresora, un DVD, el monitor, un directorio o un archivo de texto son archivos según la representación de estos sistemas operativos.

Todos hemos escuchado o escucharemos conceptos como montar o desmontar un CD, un USB, etc. Por ejemplo, el CD se monta como un directorio en el sistema de archivo. En este directorio encontraremos el contenido del CD si lo tenemos montado y nada de lo contrario.

Para comprobar qué tenemos montado en nuestro sistema podemos ejecutar, por ejemplo, el siguiente comando:

mount

Éste es un concepto básico y muy importante para conocer y entender el funcionamiento de los sistemas GNU/Linux.

4.2.3 Organización del sistema de archivos según FHS

El directorio raíz (/)

Todo surge a partir del directorio raíz. Éste es como el punto de inicio de cualquier otro directorio o archivo de nuestro sistema.

Es interesante y deseable que éste sea pequeño por temas de seguridad y funcionamiento (no deberemos crear directorios o archivos directamente en este directorio).

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF



bin	Binarios de usuario
boot	Ejecutables y archivos requeridos para el arranque
dev	Archivos de información de todos los volúmenes
etc	Archivos de configuración del sistema y de aplicaciones
home	Directorio personal con las carpetas de usuario
lib	Bibliotecas necesarias para la ejecución de binarios
media	Directorio de montaje de volúmenes extraíbles
opt	Ficheros de aplicaciones externas que no se integran en /usr
proc	Ficheros de información de procesos
root	Directorio personal de superusuario
sbin	Binarios de sistema
srv	Archivos relativos a servidores web, FTP, etc.
sys	Archivos virtuales con información de eventos del sistema
tmp	Directorio de ficheros temporales
usr	Archivos de programas y aplicaciones instaladas
var	Archivos de variables, logs, emails de los usuarios del sistema, etc.

Otros directorios

Existen otros directorios que no especifica el estándar FHS, pero que son importantes:

- Directorio /lost+found. "Perdidos y encontrados" Las herramientas y utilidades para restaurar y/o reparar el sistema de archivos almacenan datos en este directorio. Es un espacio temporal en el que se guardan los datos que se recuperan después de una caída del sistema. Fijamos que, normalmente por cada partición que creemos habrá un /lost+found en el nivel superior.

- Directorio /proc (visto en un punto anterior): es un sistema de archivos virtual. Se genera y actualiza dinámicamente, esto es, no se mantiene en el disco duro sino en la

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

memoria RAM. Es el sistema quien lo crea y destruye. Este directorio contiene información sobre los procesos, el núcleo e información relativa al sistema.

5.- Permisos y atributos

5.1.- Permisos y atributos en Windows

Un atributo es una característica de un archivo o carpeta que se aplica a todos los usuarios. Por ejemplo, el atributo Oculto hace que el archivo no aparezca cuando entremos en su directorio. Los atributos de los archivos y carpetas se utilizan desde las primeras versiones del sistema de archivos FAT.

El sistema de archivos NTFS, además de los atributos, incluye el concepto de permisos. Los permisos de un archivo o carpeta definen qué puede hacer cada usuario con ese archivo o carpeta (si puede o no leerlo, si puede cambiarlo si puede eliminarlo, etc). Los sistemas Microsoft a partir de Windows XP utilizan los sistemas de archivos NTFS por lo que podemos aplicar permisos a archivos o carpetas ubicados en particiones con formato NTFS.

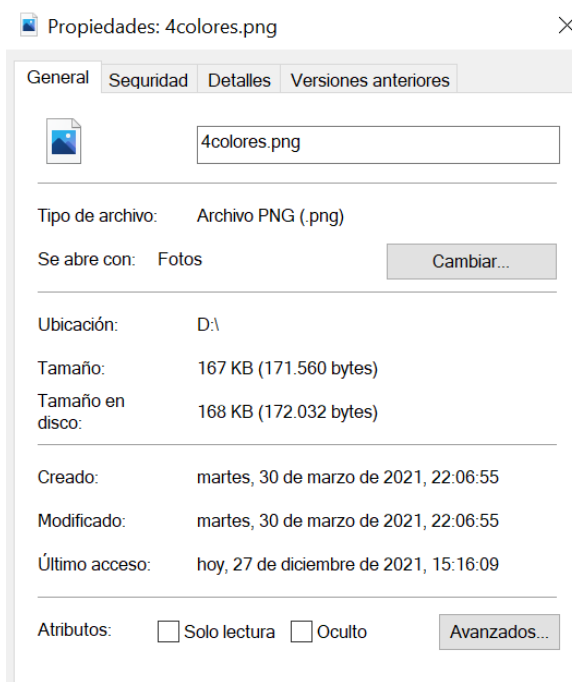
5.1.1 Atributos

Los atributos que puede tener un archivo o carpeta son:

- Oculto (H): indica que el archivo no se ve al entrar en su directorio
- Sólo lectura (R): indica que no podemos modificar el contenido del archivo

Para ver y establecer los atributos de un archivo o carpeta lo seleccionamos y con el botón derecho del ratón seleccionamos Propiedades.

Además de los 2 atributos que utilizamos habitualmente hay dos más que utiliza Windows que son el de Sistema (S) que significa que se trata de un archivo especial del sistema operativo y el de Archivo (A) que utiliza para realizar copias de seguridad.



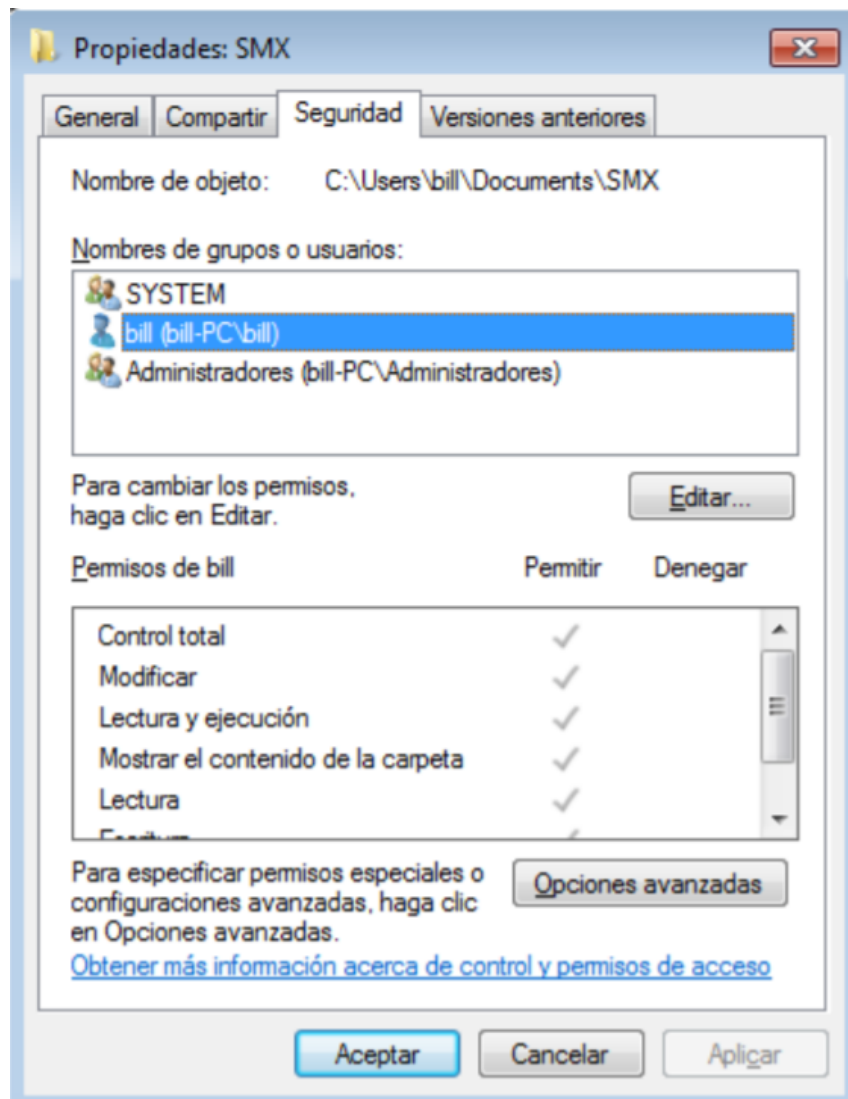
COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

5.1.2 Configuración de permisos NTFS

A diferencia de los atributos que se aplican a todos los usuarios los permisos se especifican para cada usuario o grupo concreto, lo que significa que sobre un mismo archivo un usuario tendrá unos permisos distintos al que tiene otro usuario.

Para ver y establecer los permisos de un archivo o carpeta lo seleccionamos y con el botón derecho del ratón seleccionamos Propiedades y vamos a la pestaña de Seguridad:



En la ventana aparece en primer lugar la ruta del recurso (archivo o directorio) del que estamos viendo sus permisos. En este caso la carpeta c:\Users\bill\Documents\SMX.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

A continuación tenemos a los usuarios y grupos con permisos sobre este recurso. En la imagen tienen permisos el grupo SYSTEM, el usuario *bill* y el grupo *Administradores*. Quienes su icono es una persona se trata de un usuario y los que son dos personas es un grupo.

Un grupo es un conjunto de usuarios (por ejemplo en el grupo Administradores están todos los usuarios que sean administradores del equipo y en el grupo Usuarios autenticados todos los usuarios que han entrado con un usuario y contraseña). Cuando asignamos permisos a un grupo todos los usuarios que pertenecen al grupo tendrán esos permisos. El grupo SYSTEM es un grupo especial que utilizan algunas utilidades de Windows para realizar su trabajo (por ejemplo el programa de copias de seguridad necesita permisos para acceder a los archivos de los que debe hacer copia).

En la parte inferior de la ventana encontramos qué permisos tiene el usuario o grupo seleccionado arriba. En este caso, podemos ver que el usuario *bill* tiene permiso de “Control Total” sobre esta carpeta. El permiso “Control total” incluye los permisos de Modificar, Lectura y ejecución, Lectura y Escritura por lo que aparecen todos marcados.

5.1.3 Permisos estándar y especiales

En Windows encontramos 2 tipos de permisos NTFS:

- Permisos NTFS **especiales**: controlan cada acción que un usuario puede o no realizar sobre una carpeta o archivo. Son muchos y es bastante compleja su administración
- Permisos NTFS **estándar**: son combinaciones de varios permisos NTFS especiales para hacer más sencillo su uso. Normalmente utilizaremos sólo los permisos estándar y si en algún momento necesitamos un mayor grado de control utilizamos los permisos especiales.

Los permisos estándar que usuarios y grupos pueden tener sobre archivos y carpetas son 6 aunque nosotros casi siempre utilizaremos sólo los 3 primeros:

- Control total: permite hacer cualquier cosa, incluyendo cambiar los permisos o la propiedad del archivo o carpeta. Este permiso incluye a todos los demás.
- Modificar: permite hacer cualquier cosa con el recurso, como leer, modificar o eliminar pero no permite cambiar los permisos. Este permiso incluye los permisos de Leer y ejecutar y Escritura.
- Leer y ejecutar: permite leer el recurso y ejecutarlo si se trata de un programa. Si es una carpeta podemos ver su contenido y entrar dentro. Este permiso incluye el permiso de Leer y, en el caso de carpetas, incluye el de Mostrar contenido.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

- Mostrar el contenido de la carpeta: se aplica a carpetas y permite ver su contenido y entrar dentro de la carpeta
- Leer: permite ver el contenido (si es una carpeta ver los archivos que contiene pero no entrar dentro) y ver sus atributos y permisos.
- Escribir: si es un archivo permite modificarlo y si es una carpeta permite copiar archivos dentro. Pero no eliminar nada ni leerlo (si es un archivo) o ver su contenido ni entrar dentro (si es una carpeta).

La mayoría de las veces utilizamos sólo los permisos de Control total, Modificar o Leer y ejecutar.

En la siguiente tabla podemos ver cada permiso estándar a qué permisos especiales corresponde. La primera tabla muestra los permisos para los archivos y la segunda para las carpetas (que es donde normalmente daremos los permisos):

	Control total	Cambiar	Leer y ejecutar	Mostrar el contenido de la carpeta	Leer	Escribir
Recorrer por carpeta/ejecutar archivo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
Mostrar carpeta/leer datos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Atributos de lectura	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Atributos extendidos de lectura	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Crear archivos/escribir datos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>
Crear carpetas/agregar datos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>
Atributos de escritura	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>
Atributos extendidos de escritura	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>
Eliminar subcarpetas y archivos	<input checked="" type="checkbox"/>					
Eliminar	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				
Permisos de lectura	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Cambiar permisos	<input checked="" type="checkbox"/>					
Tomar posesión	<input checked="" type="checkbox"/>					

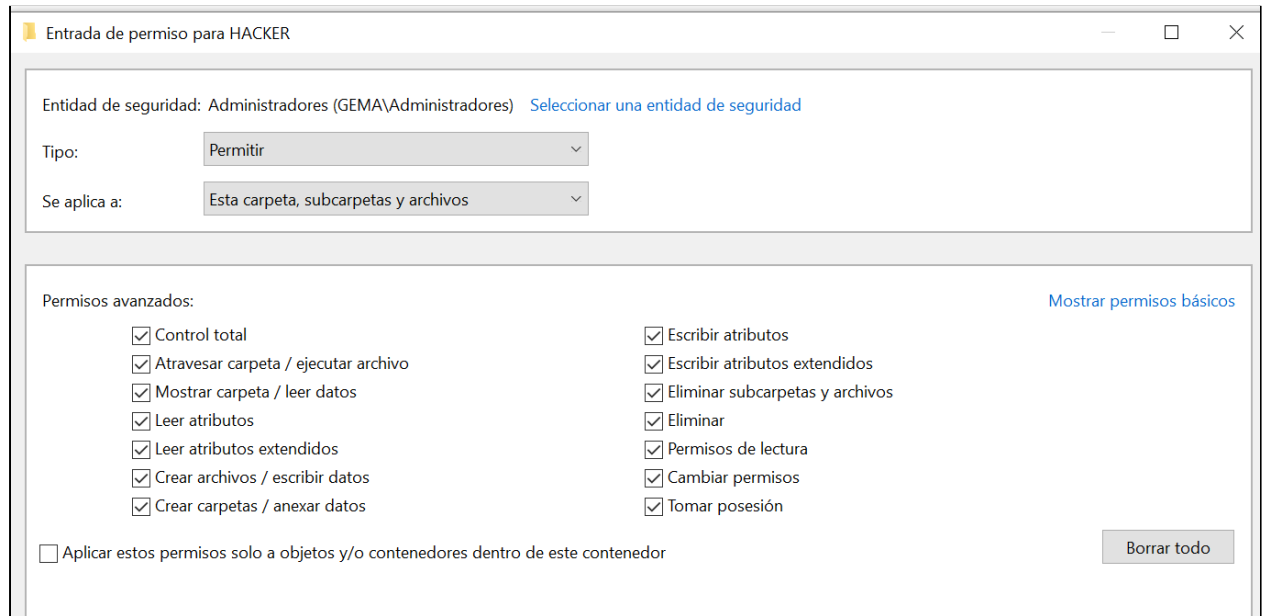
Definición de los permisos especiales

Los Permisos especiales se encuentran seleccionando un usuario/grupo concreto dentro de la

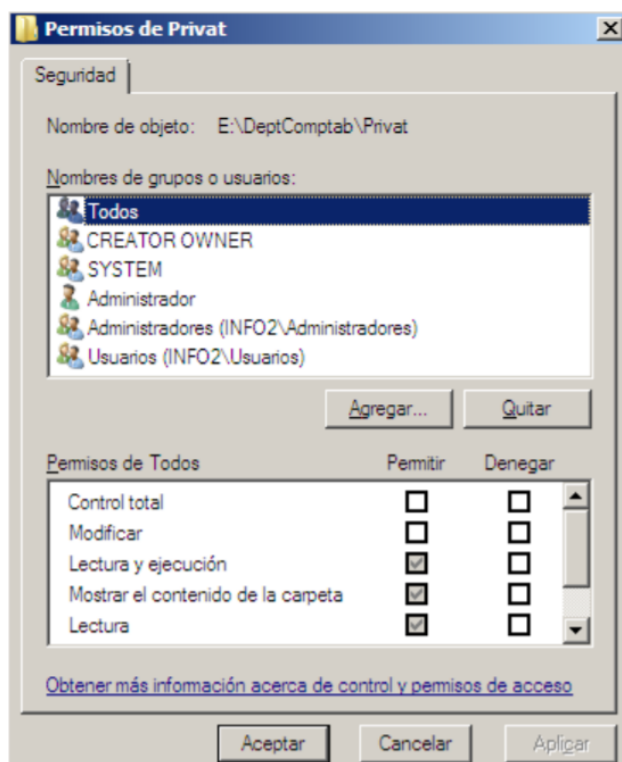
COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

pestaña seguridad-Opciones avanzadas-Editar .



Los permisos especiales de la imagen superior los podemos configurar por archivos y carpetas.



5.1.4 Añadir permisos

Para cambiar los permisos de un archivo o carpeta desde la pestaña de Seguridad utilizamos el botón de Editar:

Los permisos que aparecen en gris son permisos heredados de la carpeta padre y no podemos quitarlos (posteriormente veremos cómo quitarlos).

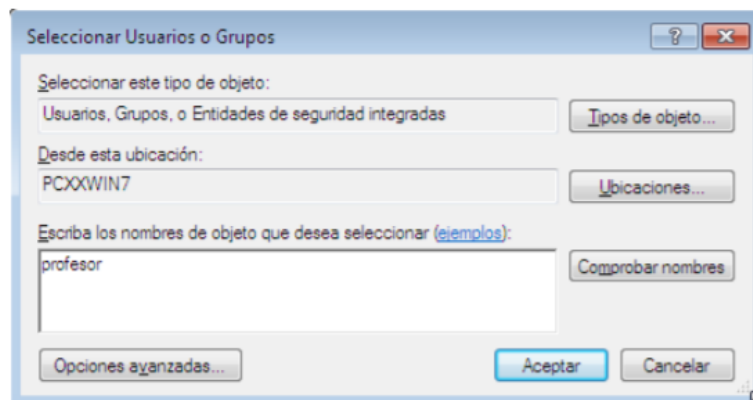
Para añadir un permiso a un usuario o grupo de los que ya tienen permisos sólo debemos marcarlo. El nuevo permiso aparece en negro y si queremos podemos quitarlo desmarcándolo.

Para dar permisos a un usuario o grupo que no está en la lista

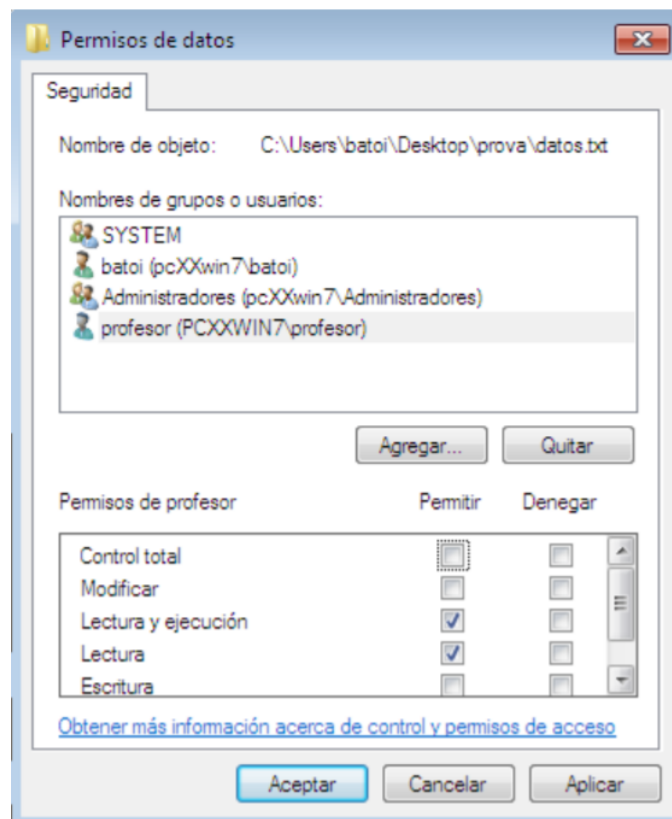
COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

pulsamos el botón de Añadir:



Escribimos el nombre del usuario o grupo y pulsamos el botón en Aceptar. Ya aparece en la lista de usuarios y podemos marcar los permisos que deseamos:



Si posteriormente queremos quitarle todos los permisos lo seleccionamos y pulsamos el botón de Quitar.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

5.1.5 Permisos heredados y explícitos

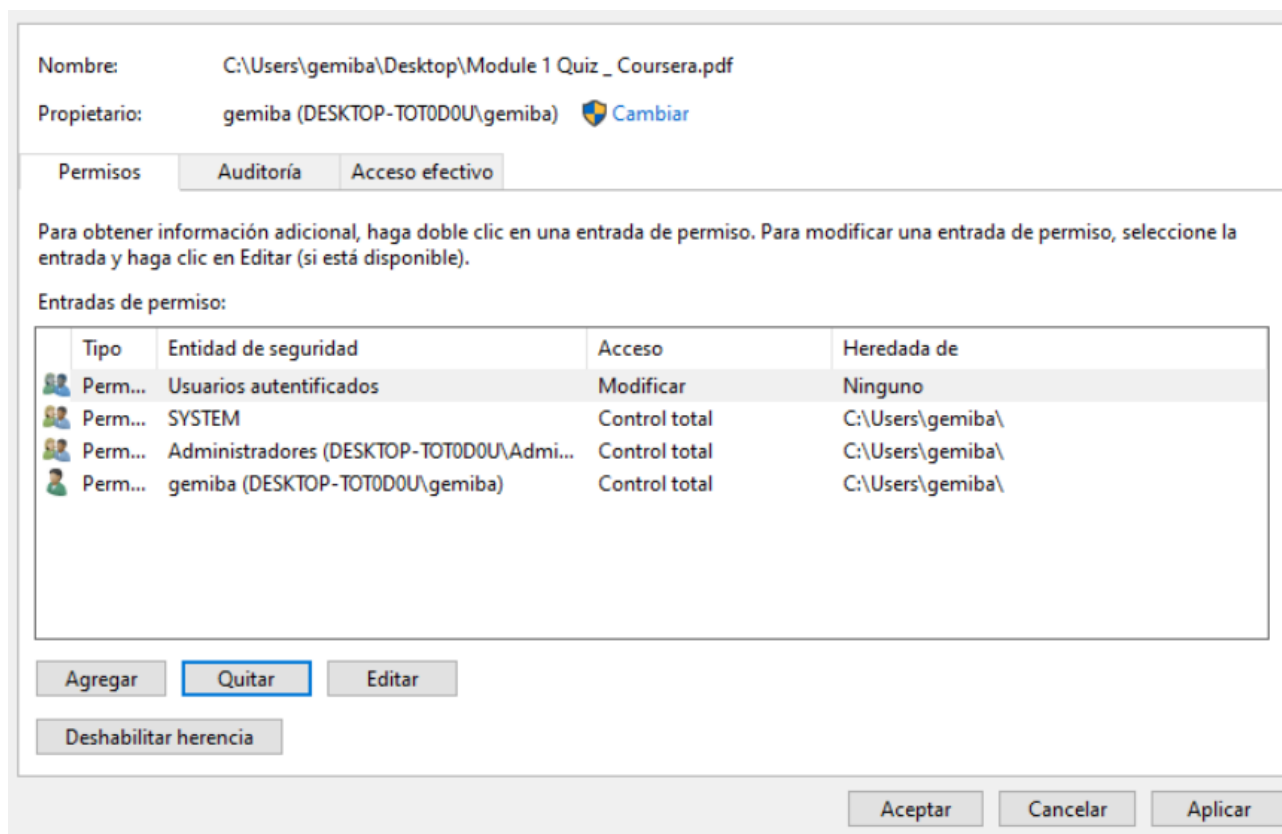
Al crear una carpeta o archivo éste hereda los permisos de la carpeta o unidad donde se crea. Estos permisos se llaman permisos heredados. La herencia es una funcionalidad muy práctica porque los permisos que definimos sobre una carpeta tendrán también todos los archivos y subcarpetas que contenga.

Además de los permisos cualquier usuario que tenga permiso de Control total sobre el recurso (por ejemplo su propietario) puede añadir permisos llamados explícitos.


Los permisos que aparecen en gris son permisos heredados y los que aparecen en negro son permisos explícitos.

En algunos casos queremos que un objeto no herede los permisos de su objeto padre. Para eliminar la herencia de un objeto desde la pestaña de Seguridad pulsamos el botón Opciones avanzadas.

Desde aquí podemos ver todos los permisos que tiene definidos el objeto y si son o no heredados:



Nombre: C:\Users\gemiba\Desktop\Module 1 Quiz_Coursera.pdf

Propietario: gemiba (DESKTOP-TOT0D0U\gemiba)  Cambiar

Permisos Auditoría Acceso efectivo

Para obtener información adicional, haga doble clic en una entrada de permiso. Para modificar una entrada de permiso, seleccione la entrada y haga clic en Editar (si está disponible).

Entradas de permiso:

Tipo	Entidad de seguridad	Acceso	Heredada de
Perm...	Usuarios autenticados	Modificar	Ninguno
Perm...	SYSTEM	Control total	C:\Users\gemiba\
Perm...	Administradores (DESKTOP-TOT0D0U\Admi...	Control total	C:\Users\gemiba\
Perm...	gemiba (DESKTOP-TOT0D0U\gemiba)	Control total	C:\Users\gemiba\

Agregar **Quitar** Editar

Deshabilitar herencia

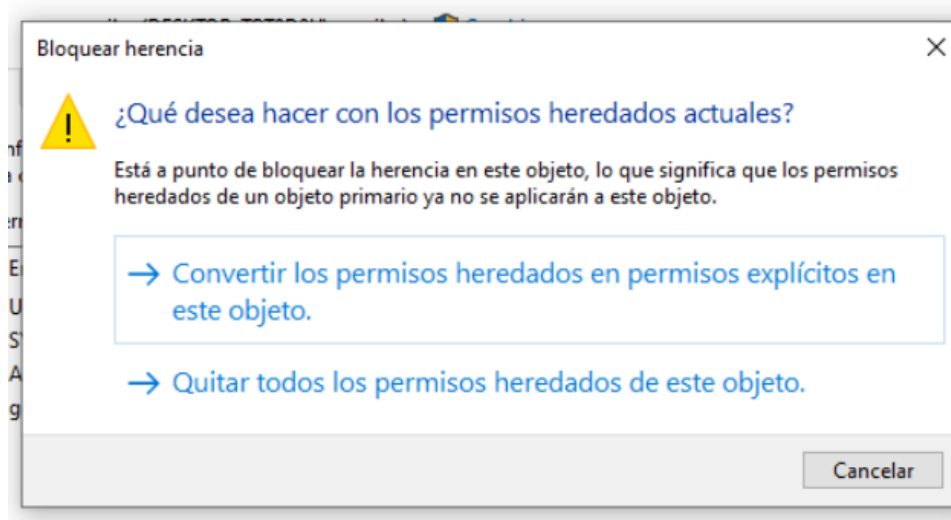
Aceptar Cancelar Aplicar

Como podemos ver los miembros del grupo “Usuarios autenticados” tienen permiso de Modificar y es un permiso explícito, el resto son permisos heredados.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Desde aquí podemos añadir, eliminar o editar los permisos a nivel de permisos especiales. Para hacer que este objeto no herede los permisos de su objeto padre clicamos en “Deshabilitar herencia”. Al hacerlo aparece la siguiente pantalla:



Las opciones que nos da son:

- Convertir los permisos heredados en permisos explícitos de modo que podemos cambiarlos si queremos. Es la opción que escogeremos habitualmente.
- Quitar todos los permisos heredados de este objeto

Al seleccionar la primera opción el objeto ya no tiene permisos heredados sino que todos son explícitos y podemos cambiarlos como queramos.

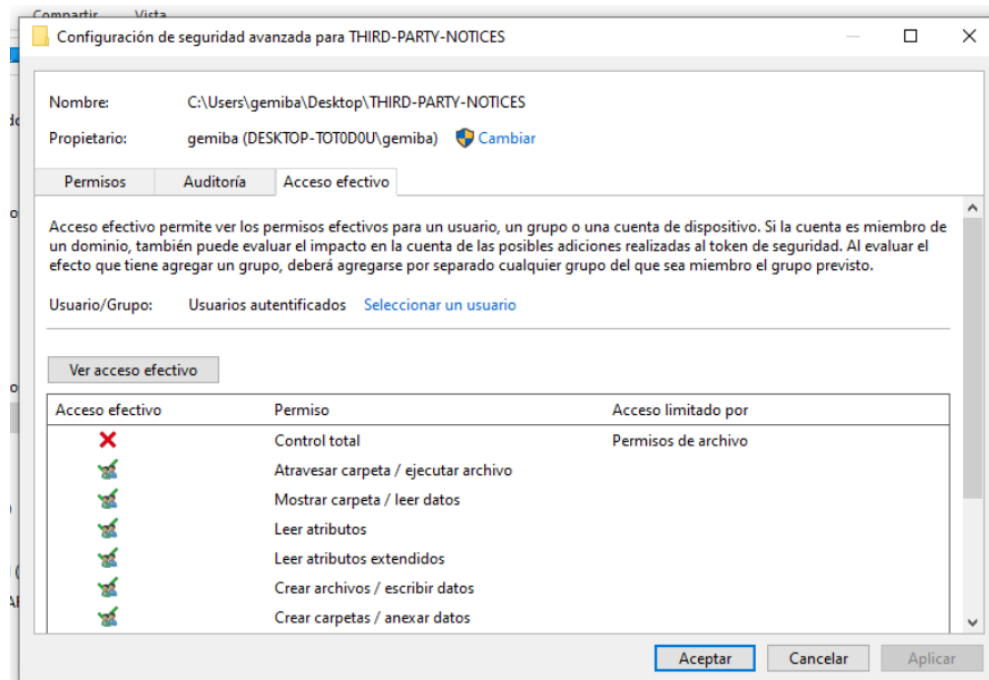
5.1.6 Permisos efectivos

Que un usuario tenga o no permiso sobre un archivo o carpeta depende de los permisos del usuario, de los permisos de todos los grupos a los que pertenece el usuario, de los permisos explícitos del objeto y de los permisos heredados del usuario mismo. Por eso a veces resulta difícil determinar qué permisos reales tenemos sobre algún recurso.

Windows incluye una funcionalidad que nos permite visualizar los permisos efectivos de cualquier usuario sobre una carpeta o archivo determinado. Desde el menú contextual -> Propiedades -> pestaña Seguridad -> botón Opciones avanzadas -> pestaña Permisos efectivos:

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF



5.2.- Permisos en Gnu/Linux

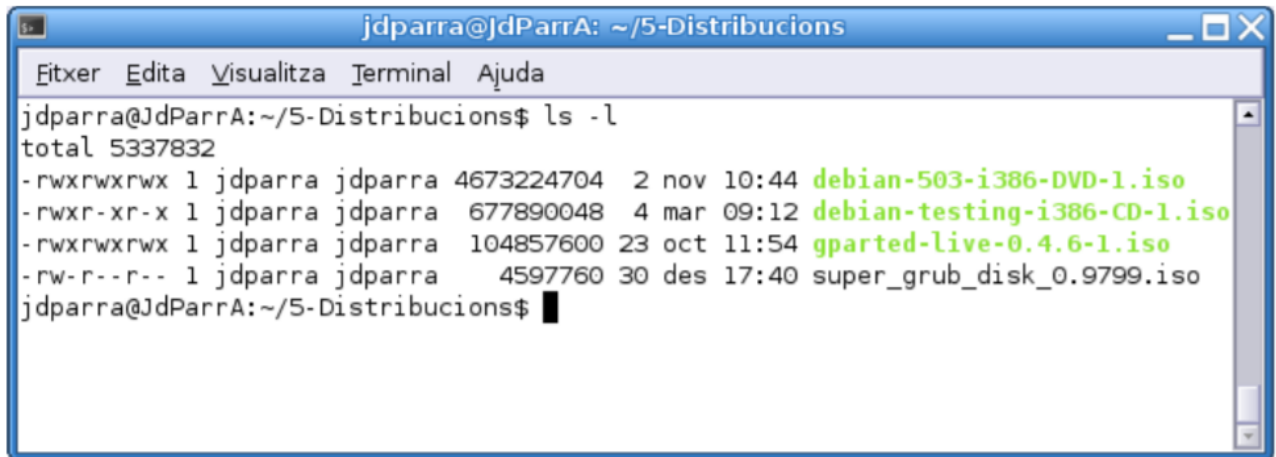
5.2.1 Introducción

En cualquier sistema operativo multiusuario necesitamos que los archivos que guardamos en nuestro disco puedan tener una serie de propiedades que nos permiten verlos, modificarlos o ejecutarlos para los usuarios que nosotros hemos definido.

Aunque existen varias alternativas para ello, GNU/Linux utiliza el sistema clásico de usuarios y grupos, que nos permite cualquier configuración posible. Lo que interesa es definir, para cada archivo o directorio, a qué usuario y grupo pertenece y qué permisos tiene para cada uno de ellos, así como para el resto de usuarios del sistema. Al ejecutar **ls -l** veremos cómo en cada archivo del directorio donde estamos aparece una línea similar a la siguiente:

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF



```
jdparra@JdParraA: ~/5-Distribucions
Fitxer  Edita  Visualitza  Terminal  Ajuda
jdparra@JdParraA:~/5-Distribucions$ ls -l
total 5337832
-rwxrwxrwx 1 jdparra jdparra 4673224704  2 nov 10:44 debian-503-i386-DVD-1.iso
-rwxr-xr-x 1 jdparra jdparra  677890048  4 mar 09:12 debian-testing-i386-CD-1.iso
-rwxrwxrwx 1 jdparra jdparra  104857600 23 oct 11:54 gparted-live-0.4.6-1.iso
-rw-r--r-- 1 jdparra jdparra   4597760 30 des 17:40 super_grub_disk_0.9799.iso
jdparra@JdParraA:~/5-Distribucions$
```

- Carácter 1: esta entrada nos indica si es un archivo o directorio. En caso de ser un archivo, aparece el carácter "-", mientras que para los directorios aparece una "d" ; "b" para archivos de bloques, "c" archivos de caracteres especiales;"l" archivo de enlace y "p" pipe o tubería .

- Caracteres 2, 3, 4: nos indican, respectivamente, los permisos de lectura, escritura y ejecución para el propietario del archivo. En el caso de no tener el permiso correspondiente activado, encontramos el carácter "-" y si no "r", "w" o "x", según le podamos leer (read), escribir (write) o ejecutar (execute).

En el caso de un directorio la "x" significa que podemos entrar en el directorio. En el tercer carácter, además, podemos encontrar una "s", que nos indica si el archivo es de tipo SetUserId, que significa que al ejecutarlo obtendrá los permisos del propietario del archivo (si sólo tiene el permiso "x" , cuando el programa se ejecuta lo hace con los permisos de quien lo haya lanzado).

- Caracteres 5, 6, 7: estos caracteres tienen exactamente el mismo significado que los anteriores, pero hacen referencia a los permisos concedidos a los usuarios del grupo al que pertenece el archivo. En el carácter 7 podríamos tener una "s" si el directorio es de tipo SetGroupId, lo que significa que los archivos y directorios que se creen dentro tendrán este grupo como grupo propietario.

- Caracteres 8, 9, 10: igual que en el caso anterior, pero para otros usuarios del sistema.

Después de estos 10 caracteres encontramos una cifra que nos indica el número de enlaces fuertes que tiene el fichero. Para los directorios, este número indica cuántas carpetas contiene además de los enlaces fuertes que tiene (cuando no hay ninguno, el número es 2, que hace referencia en las carpetas "."(la misma carpeta) y ".." (la

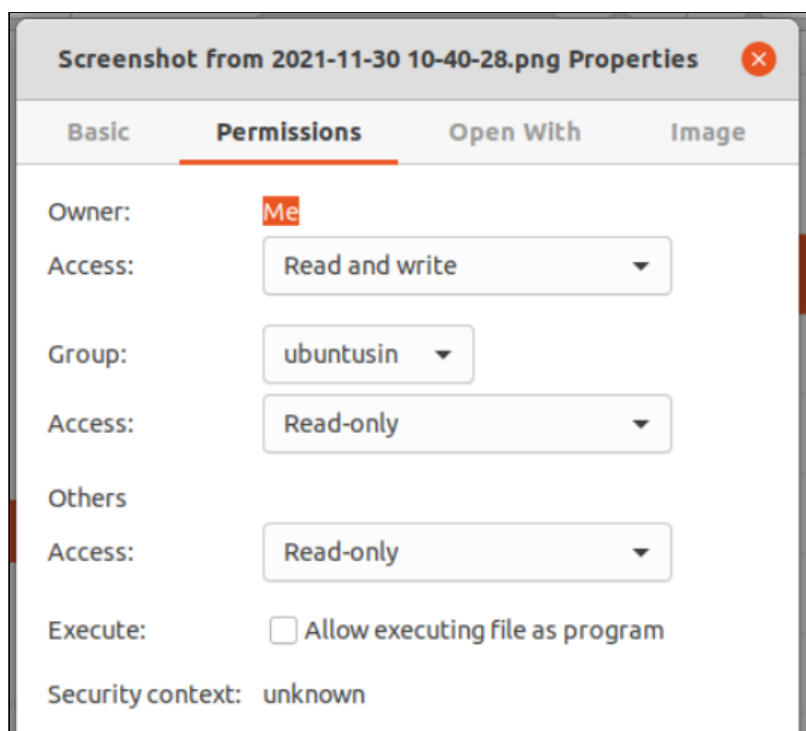
COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

carpeta padre). A continuación, vemos el propietario y el grupo propietario del archivo, seguido de la medida (en bytes) que ocupa y la fecha de la última modificación. Para todos los ficheros se guarda su fecha de creación, la del último acceso y la última modificación, que podemos manipular con la orden **touch**. Al final se encuentra el nombre del fichero, en el que se diferencian minúsculas de mayúsculas y podemos tener todo tipo de caracteres sin ningún problema.

Para cambiar los permisos de un archivo determinado podemos utilizar la orden **chmod**. Debemos tener en cuenta que solo el propietario del archivo (o root) puede cambiar estos, ya que de otra forma el mecanismo no tendría sentido.

Para ver y/o modificar los permisos podemos acceder pulsando sobre el botón derecho del ratón sobre el archivo o directorio e ir a la pestaña permisos:



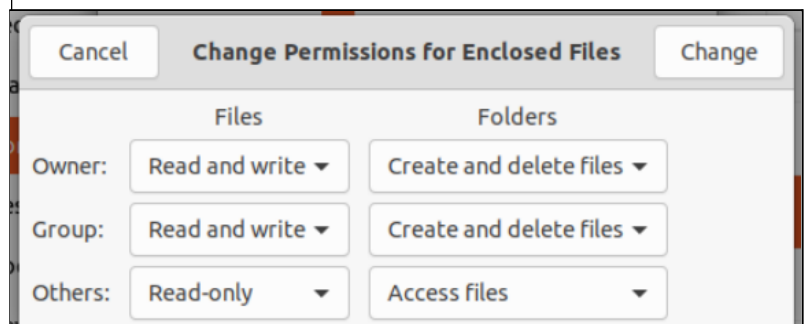
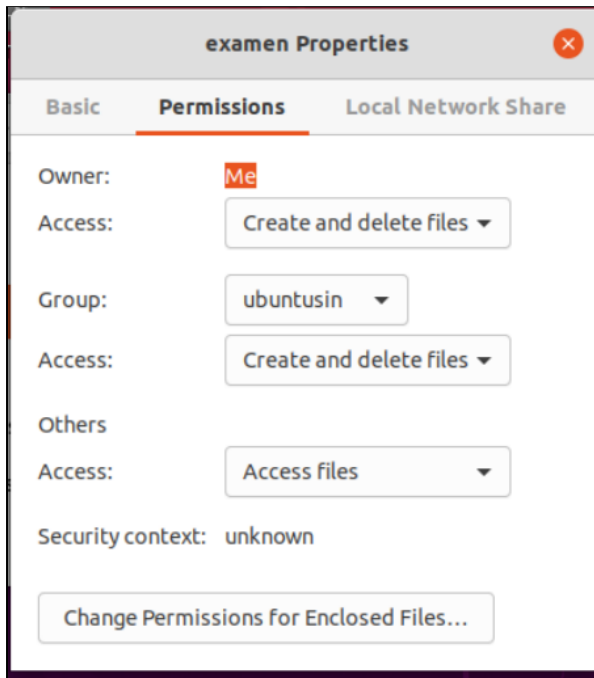
El significado de los permisos que se ven en la ventana anterior es, para los archivos:

- Ninguno : ---
- Sólo lectura: r--
- Lectura y escritura: rw-
- Si marcamos la opción de "Permite ejecutar este archivo como un programa" se añade el permiso x, pasando "Sólo lectura" a ser rx y "Lectura y escritura" en rwx

Si estamos sobre una carpeta en vez de un archivo la pantalla cambia:

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF



El significado para las carpetas es:

- Ninguno: ---
- Sólo listar archivos : r--
- Acceder a los archivos: rx
- Crear y suprimir archivos: rwx

La parte de acceso al archivo nos permite modificar los permisos de los archivos contenidos en la carpeta poniendo la opción que queramos (Ninguna, Sólo lectura o Lectura y escritura, más la opción de Ejecutar).

5.2.2 Cambiar los permisos por comandos

chmod

Cambia los permisos de los archivos o directorios pasados por parámetro. Funciona de dos formas: Permite añadir (+), eliminar (-) o asignar (=) permisos de lectura (r), escritura (w) o ejecución (x) para el propietario (u), el grupo (g), la resto de usuarios (o) o todos a la vez (a). Ejemplos:

- `chmod g-w alumnos.txt` : Elimina el permiso de escritura sobre el archivo alumnos.txt a los usuarios del grupo al que pertenece el archivo

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

- `chmod a+x alumnos.txt` : Añade permisos de ejecución sobre el archivo a todos los usuarios
- `chmod u=rw alumnos.txt` : Mueve al propietario permisos de lectura y ejecución sobre el fichero y quita el de ejecución (si lo tenía)

También permite establecer directamente qué permisos tendrá, en notación octal:

- 1 permisos de ejecución (`--x = 001 = 1`)
- 2 permisos de escritura (`-w- = 010 = 2`)
- 4 permisos de lectura (`r-- = 100 = 4`)

Por tanto, lectura y escritura será $4+2=6$ (puesto que `rw- = 110 = 6`), lectura y ejecución 5 ($4+1$) y todos los permisos será 7 ($4+2+1$). De esta forma para establecer los permisos indicamos un número octal para el usuario, otro para el grupo y otro para el resto de usuarios. Ejemplos:

- `chmod 664 alumnos.txt` : Los permisos sobre el archivo `alumnos.txt` serán de lectura y escritura (6) para el propietario, lectura y escritura (6) para los miembros del grupo propietario y de lectura (4) para la resto de usuarios
- `chmod 750 examen` : Asigna todos los permisos (lectura, escritura y ejecución -en el caso de un directorio, entrar dentro-) al propietario, permisos de lectura y ejecución/entrar en el grupo y ningún permiso en el resto sobre el archivo o directorio `examen`

Este comando permite la opción `-R` para ejecutarse recursivamente sobre el contenido de un directorio. Ejemplo:

- `chmod -R 750 examen` : Asigna todos los permisos (lectura, escritura y entrar dentro) al propietario, permisos de lectura y entrar dentro del grupo y ningún permiso en el resto sobre el directorio `examen` y todos los archivos y directorios contenidos dentro de `examen`

chown

Permite cambiar el propietario y el grupo propietario de los archivos o directorios pasados por parámetro. Ejemplos:

- `chown gemiba alumnos.txt` : El archivo `alumnos.txt` pasa a pertenecer al usuario `gemiba`
- `chown gemiba:examen alumnos.txt`: El archivo `alumnos.txt` pasa a pertenecer al usuario `gemiba` y al grupo `examen`

Al igual que lo anterior permite el uso de la opción `-R` para funcionar recursivamente.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

chgrp

Permite cambiar el grupo propietario de los archivos o directorios pasados por parámetro.

5.2.3.- Permisos especiales en Gnu/Linux

Setuid bit (s o S)

Aplicado sobre un archivo ejecutable, permite que el programa no se ejecute con los permisos del usuario que lo *ejecuta* sino *con los permisos del usuario propietario* (si el bit está activado para el usuario, o SetUid) o el grupo propietario (si está activado para el grupo, o SetGid) del archivo.

Un ejemplo de archivo “setuidat” es el archivo `/usr/bin/passwd` que debe modificar el archivo `/etc/shadow` para cambiar una contraseña. El archivo ejecutable `/bin/passwd` pertenece al usuario root y al grupo root, y tiene el bit SetUID activo. Las contraseñas se guardan en `/etc/shadow` que sólo tiene permisos de escritura para root. Cuando un usuario ejecuta el archivo `/bin/passwd` este archivo escribe su nueva contraseña en `/etc/shadow` y lo hace como si fuera root y no el usuario lo que lo ha ejecutado porque el usuario no podría escribir en el archivo `/etc/shadow`.

El permiso **s** está representado en su lugar por la **x** del propietario o grupo de propietarios, por lo que se muestra como **s** si el archivo o directorio tiene los seis permisos con **S** si solo tiene el permiso s.

NOTA: Debemos tener muy en cuenta en los archivos “setuidats” por los permisos que adquiere cualquier usuario que los ejecute.

Un directorio “setuidat” (en este caso sólo puede tener el permiso el grupo) hace que los archivos y subdirectorios que se crean dentro no tengan como grupo propietario el grupo principal del usuario que los crea sino que heredan el grupo propietario (y también el permiso s) de este directorio. Se utiliza por ejemplo en el directorio de un FTP anónimo.

Sticky bit (t o T)

El permiso **t** se representa en el sitio del **x** del resto de usuarios por lo que se muestra cómo **t** si el archivo o directorio tiene los permisos tixo con **T** si sólo tiene el permiso t.

Normalmente se utiliza en directorios e indica que los archivos que se crean dentro sólo se podrán borrar por su propietario (o por root). Se usa por ejemplo en `/tmp`.

En el caso de un archivo le indica al sistema que se guarde en la swap en lugar de en la RAM para ahorrar memoria pero hoy en día no tiene demasiado sentido ya que los equipos modernos tienen GB de RAM.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

Para establecer estos permisos se utilizan 3 bits frente a los 9 normales. Por tanto el número octal que indica los permisos a `chmod` tendrá 4 dígitos en lugar de 3 y el primero será el de los permisos especiales. Ese primer número octal representa el bit `s` del propietario, el bit `s` del grupo y el bit `t` del resto:

`sst rwx rwx rwx`

Por ejemplo:

`--- rwx rw- r--` se representaría como 764 (o 0764)

`rws rwS r--`

en realidad equivale a:

`ss- rwx rw- r--`

por lo que en octal sería 6764.

Como hemos comentado, no administrar correctamente estos permisos puede provocar problemas de seguridad por lo que no debemos utilizarlos a menos que sea estrictamente necesario.

5.2.4.- Máscaras de permisos

Por defecto al crear un archivo se crea con los permisos 644 y al crear un directorio con 755. Esto es porque el sistema tiene una máscara, que por defecto es 0022, que se resta a los permisos totales (666 para archivos y 777 para directorios) cuando creamos un nuevo recurso.

Para cambiar la máscara por defecto que tiene un usuario tenemos el comando **umask**. Sin parámetros muestra la máscara actual del usuario y le podemos pasar como parámetro la nueva máscara por defecto (en octal).

Por ejemplo si queremos que el resto de usuarios no tengan ningún permiso sobre nuestros archivos estableceremos nuestra máscara con:

`umask 027`

(al propietario no se le quita ningún permiso -0-, al grupo propietario se le quita el de escritura - 2- y en el resto los permisos de lectura, escritura y ejecución -4,2,1=7-).

Si queremos hacer permanente la nueva máscara añadiremos el orden al archivo `.bash_profile` o `.bash_rc` de nuestro directorio de inicio. Si queremos que se aplique a TODOS los usuarios la añadiremos a `/etc/profile` o `/etc/bash.bashrc`.

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

5.2.5.- Permisos ACL

Normalmente con los permisos UGO tenemos suficiente para asignar los permisos necesarios a archivos y directorios pero en ocasiones necesitamos poder asignar permisos diferentes a distintos grupos. En ese caso podemos utilizar ACLs en GNU/Linux al igual que se hace en Windows.

Lo único necesario es montar el sistema de archivos en el que queramos usarlos con la opción "acl" (en el archivo /etc/fstab o con el comando mount). Podemos comprobar si un sistema de archivos tiene activado ACL con el comando **tune2fs**, por ejemplo:

```
tune2fs -l /dev/sda1 | grep acl
```

La salida debe ser algo como:

```
Default mount options: user_xattr acl
```

Ahora ya podemos asignar los permisos ACL. Para ello podemos instalar alguna herramienta gráfica, como por ejemplo **eiciel**, o hacerlo desde la terminal.

Al mirar los permisos desde la terminal si un archivo o directorio tiene establecidos permisos ACL aparece junto a los permisos el símbolo + que nos dice que hay más permisos (los ACL):

```
juan@juan-TravelMate-B113 ~/prova $ ls -l
total 8
drwxrwxr-x+ 2 juan juan 4096 sep 18 00:03 provaacl
```

El comando para ver los permisos ACL de un archivo o directorio es **getfacl**:

Lo que nos dice es:

- # file: provaacl/ - el nombre del archivo o directorio (dice file aunque sea un directorio)
- # owner: juan - nombre del usuario propietario
- # group: juan - nombre del grupo propietario
- user::rwx - permisos del usuario propietario (en

```
juan@juan-TravelMate-B113 ~ $ getfacl provaacl/
# file: provaacl/
# owner: juan
# group: juan
user::rwx
group::r-x
group:alumnes:r-x
group:professors:rwx
mask::rwx
other::r-x
default:user::rwx
default:group::r-x
default:group:alumnes:r-x
default:group:professors:rwx
default:mask::rwx
default:other::r-x
```

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

este caso juan)

- group::rx - permisos del grupo propietario (en este caso el grupo juan)
- group:alumnos:rx - permisos del grupo alumnos
- group:profesores:rwx - permisos del grupo profesores
- mask::rwx - la máscara indica los permisos máximos que se pueden dar
- other::rx - permisos del resto de usuarios
- default:user::rwx - permisos del usuario propietario (en este caso juan) para los nuevos archivos y directorios que se crean aquí dentro
- default:group::rx - permisos del grupo propietario (en este caso juan) para los nuevos archivos y directorios que se crean aquí dentro
- default:group:alumnos:rx - permisos del grupo alumnos para los nuevos archivos y directorios que se crean aquí dentro
- default:group:profesores:rwx - permisos del grupo profesores para los nuevos archivos y directorios que se crean aquí dentro
- default:mask::rwx - máscara para los nuevos archivos y directorios que se crean dentro
- default:other::rx - permisos para el resto de usuarios para los nuevos archivos y directorios que se crean aquí dentro

El comando para asignar permisos ACL es **setfacl**. Los principales parámetros que tiene son:

- -b borra permisos
- -m añade permisos a los que ya tuviera
- -s quita los permisos que tuviera y pon los que le indican
- -d (junto a -mo -s) los permisos que ponemos serán para los nuevos archivos y directorios que se crean dentro (default) no para el archivo o- directorio indicado
- -R aplica el comando también a todos los archivos y directorios incluidos en ese directorio

Para indicar el permiso a añadir o borrar debemos decir si es para usuario o grupo (u | g), a qué usuario o grupo se aplicará (podemos poner el nombre o el UID | GID) y qué permiso dar (rwx o el octal). Ejemplos:

```
setfacl -b provaacl
```

Borra todos los permisos ACL de provaacl

```
setfacl -Rm g:alumnos:rx provaacl
```

Le da permisos rx al grupo alumnos sobre el directorio provaacl y todo lo que contiene

```
setfacl -Rm g:profesores:rwx provaacl
```

Le da permisos rwx al grupo profesores sobre el directorio provaacl y todo lo que contiene

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

```
setfacl -dm g:alumnos:rx provaacl
```

Le da permisos rx por defecto al grupo alumnos sobre los nuevos archivos o directorios que se crean dentro de provaacl.

```
setfacl -dm g:profesores:rwx provaacl
```

Le da permisos rwx por defecto al grupo profesores sobre los nuevos archivos o directorios que se crean dentro de provaacl.

6 Gestión de enlaces

Los enlaces proveen una forma alternativa de acceder a un archivo. Existen dos tipos de enlaces los enlaces simbólicos (en Windows llamados Accesos directos) y los enlaces rígidos (no soportados por los sistemas de archivo FAT ni NTFS).

6.1 Enlaces simbólicos

Un enlace simbólico es un archivo que lo único que contiene es la ruta de otro archivo, por tanto su tamaño será de unos pocos bytes.

Al hacer doble clic en un enlace simbólico el sistema mira la ruta que tiene guardada y abre el archivo que hay en esa ruta.

Podemos crear enlaces simbólicos a archivos de datos, programas o directorios. Desde el entorno gráfico es muy sencillo hacerlo tanto en Windows (se llaman *Accesos directos*) como en Gnu/Linux (se llaman *Lanzadores*).

Para crear un enlace simbólico en Linux desde la línea de comandos hacemos:

```
ln -s archivo_a_enlazar nombre_del_enlace
```

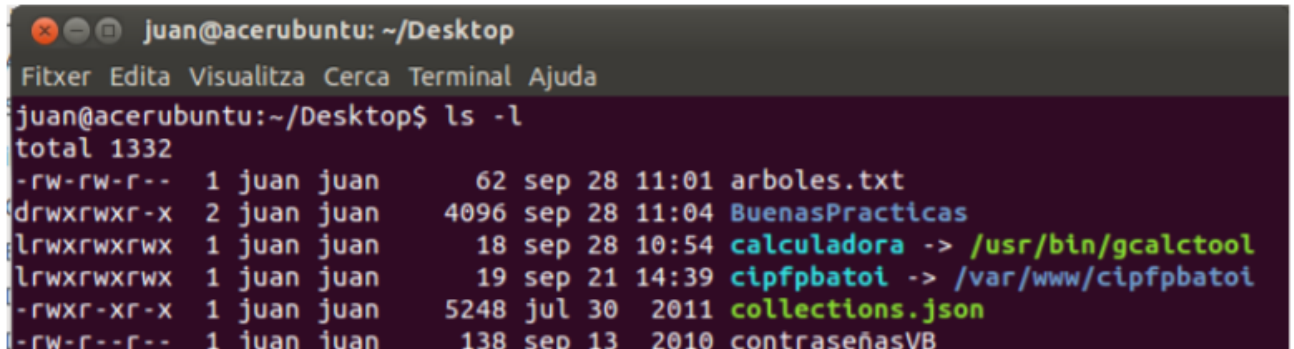
Por ejemplo para hacer en el escritorio un enlace simbólico a la calculadora (qué es el archivo gcalctool que se encuentra en /usr/bin) haremos:

```
ln -s /usr/bin/gcalctool /home/juan/Desktop/calculadora
```

Desde el entorno gráfico un enlace simbólico es un icono (como cualquier otro archivo o carpeta) pero en una flecha en su esquina inferior derecha. Si miramos sus propiedades podemos ver el archivo al que enlaza. Desde la terminal un acceso un enlace simbólico tiene junto al nombre el archivo al que enlaza y sus permisos comienzan por la letra l:

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF



```
juan@acerubuntu: ~/Desktop
Fitxer Edita Visualitza Cerca Terminal Ajuda
juan@acerubuntu:~/Desktop$ ls -l
total 1332
-rw-rw-r-- 1 juan juan      62 sep 28 11:01 arboles.txt
drwxrwxr-x 2 juan juan    4096 sep 28 11:04 BuenasPracticas
lrwxrwxrwx 1 juan juan      18 sep 28 10:54 calculadora -> /usr/bin/gcalctool
lrwxrwxrwx 1 juan juan      19 sep 21 14:39 cipfpbatoi -> /var/www/cipfpbatoi
-rwxr-xr-x 1 juan juan    5248 jul 30 2011 collections.json
-rw-r--r-- 1 juan juan     138 sep 13 2010 contraseñasVB
```

6.2 Enlaces rígidos

Un archivo está troceado en clusters y repartido por el disco duro. Para acceder a él en el directorio donde se encuentra se guarda, junto a su nombre, una información de cómo encontrarlo en el disco. En el caso de un sistema de archivos FAT esa información es el número de su primer clúster y accediendo a la FAT podemos conocer a los demás clústeres. En el directorio también se guardan los atributos del archivo.

En el caso de un sistema de archivos ext esa información es su i-nodo. Un i-nodo es una tabla en la que se guarda toda la información de un archivo, excepto su nombre: sus atributos y su localización en el disco.

Esta estructura les permite tener en un directorio dos archivos diferentes con el mismo número de i-nodo, lo que significa que los dos en realidad son el mismo archivo!!! También podemos tener un mismo archivo en diferentes directorios.

La diferencia con un enlace simbólico es que en el enlace simbólico hay un archivo "de verdad" (que si lo borramos perderemos la información que contiene y el enlace no funcionará) y un enlace que apunta a él (y si el borramos no pasa nada porque el archivo sigue estando en su sitio). En el caso de enlaces rígidos todos son el archivo real.

¿Y qué ocurre cuando borramos un enlace rígido? Como hemos dicho en realidad un enlace rígido es el archivo real y por tanto si lo borramos estamos borrando el archivo... sino quedan más enlaces rígidos al mismo.

Linux lleva la cuenta de enlaces rígidos de cada archivo: es el número que aparece entre los permisos y el propietario. Normalmente ese número siempre es 1 para los archivos (en el caso de los directorios lo que indica es el número de subdirectorios que contiene, incluyendo . y ..). Si creamos otro enlace rígido del archivo su número sube a 2, etc.

Cuando borramos un archivo lo que hace Linux es que borra la entrada de ese archivo del directorio y, si su número de enlaces es 1 elimina el archivo del disco y si es mayor

COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW
DPT INF

que 1 decrementa su número sin eliminar el archivo del disco.

Para crear un enlace simbólico en Linux desde la línea de comandos hacemos:

```
ln archivo_existente nuevo_nombre_por_al_fichero
```

Por ejemplo para tener un nuevo archivo llamado plantas.txt del archivo árboles.txt de mi escritorio haré:

```
ln /home/juan/Desktop/arboles.txt /home/juan/Desktop/plantas.txt
```

Podemos ver haciendo un `ls -l` que el número de enlaces de los 2 archivos es 2. Para saber si realmente esos 2 archivos son el mismo debemos ver su número de i-nodo. Para eso utilizamos el parámetro `-i` del comando `ls`:

```
4227546 -rw-rw-r-- 2 juan juan 62 sep 28 11:01 arboles.txt
4228607 -r----- 1 juan juan 82 ene  4 2010 kk.txt
4227546 -rw-rw-r-- 2 juan juan 62 sep 28 11:01 plantas.txt
```

Podemos ver que el número de i-nodo en ambos ficheros es 4227546.