

## **Tema 4**

# **GESTIÓN DE LA INFORMACIÓN**



<b>1. INSTALACIÓN DE SOFTWARE .....</b>	<b>1</b>
1.1. WINDOWS .....	1
1.2. LINUX 3	
<b>2. MANTENIMIENTO Y OPTIMIZACIÓN DE DISCOS .....</b>	<b>6</b>
2.1. DESFRAGMENTACIÓN.....	6
2.2. CHEQUEO Y REPARACIÓN DE DISCOS .....	7
2.2.1 Windows.....	7
2.2.2 Linux .....	8
2.3. CIFRADO.....	8
2.3.1 Windows.....	8
2.3.2 Linux .....	9
2.4. COMPRESIÓN .....	10
2.4.1 Windows.....	10
2.4.2 Linux .....	11
2.5. CUOTAS DE DISCO .....	11
2.5.1 Windows.....	11
2.5.2 Linux .....	12
<b>3. COPIAS DE SEGURIDAD.....</b>	<b>15</b>
3.1. WINDOWS .....	18
3.2. LINUX 20	
<b>4. GESTIÓN DE USUARIOS .....</b>	<b>22</b>
4.1. CONTRASEÑAS .....	22
4.2. WINDOWS .....	23
4.2.1 Gestión de Cuentas de Usuario y Grupos. ....	23
4.2.2 Gestión del inicio de sesión de los usuarios .....	27
4.2.3 Gestión de las contraseñas. ....	27
4.2.4 Bloqueo de las cuentas. ....	28
4.2.5 Recursos Locales. Gestión de ACL.....	29
4.3. LINUX 32	
4.3.1 Conceptos de gestión de usuarios. ....	32
4.3.2 Gestión de grupos.....	34
4.3.3 Permisos. ....	35

<b>5. GESTIÓN DE PROCESOS Y SERVICIOS.....</b>	<b>36</b>
5.1. WINDOWS .....	36
5.1.1 El Administrador de Tareas.....	36
5.1.2 Los Servicios del Sistema .....	38
5.2. LINUX 40	
5.2.1 Control de tareas. ....	40
5.2.2 Administración de procesos. ....	44
5.2.3 Servicios.....	46
<b>6. AUTOMATIZACIÓN DE TAREAS .....</b>	<b>46</b>
6.1. WINDOWS .....	46
6.2. LINUX 48	
<b>7. RECURSOS COMPARTIDOS .....</b>	<b>51</b>
7.1. WINDOWS .....	51
7.1.1 Recursos compartidos mediante cuenta local.....	52
7.1.2 Recursos compartidos y acceso anónimo.....	53
7.1.3 Recursos compartidos. Impresoras. ....	54
7.2. LINUX 55	
7.2.1 ¿Qué es NFS? .....	55
7.2.2 Instalación de NFS .....	55
7.2.3 Configuración del servidor NFS.....	55
Arranque y parada manual de NFS.....	56
Acceso a carpetas compartidas por NFS.....	56

# 1. INSTALACIÓN DE SOFTWARE

## 1.1. WINDOWS

### INSTALACIÓN

La instalación de un programa es algo relativamente sencillo de hacer, pero hay ciertas cosas que se deberían evitar. La correcta instalación de un programa evitará que se creen iconos de acceso directo innecesarios y lo más importante evitará ralentizar el inicio de Windows.

Los pasos a seguir son simples:

**1. Tener el disco de instalación o descargar el programa de la página Web oficial del desarrollador**

**2. Instalar el programa**

En caso de descarga, una vez terminada ésta suele aparecer automáticamente una ventana avisándonos que un programa desea ejecutarse (si no aparece, ir a la carpeta donde se guardó la descarga y doble clic para que se ejecute):

En la mayoría de programas aparecen las condiciones del contrato, que habrá que aceptar ya que de lo contrario la instalación será interrumpida

A veces se nos pide elegir el idioma, siendo el inglés el idioma por defecto. La elección del idioma muchas veces es posible realizarla cuando el programa está instalado, en las opciones del programa.

Hasta aquí no deberíamos tener problemas, luego el asistente de instalación nos propondrá instalar un icono de acceso directo en el escritorio (*Desktop shortcut*). A no ser que vayamos a utilizar mucho el programa debemos evitar los accesos directos de escritorio ya que este debe mantenerse lo más limpio y ordenado posible.

Luego se nos pide elegir la carpeta de destino, es decir la carpeta donde será instalado el programa. Aquí debemos comprobar que la ruta sea *Programas Files...*, algunos programas pueden elegir otra ubicación pero por razones de organización lo mejor es hacer la instalación en la carpeta *Programs Files*. Si instalamos software en el mismo disco donde tenemos los datos lo único que podemos conseguir es complicarnos la vida a la hora de hacer copias de seguridad (ver apartado 0).

En algunos casos, los programas pueden tener aplicaciones suplementarias, como barras de herramientas Yahoo, Google, etc. Estas por lo general ya están presentes en nuestro sistema y lo único que harán es ralentizar aún más el PC.

Para terminar, algunos autores de programas ofrecen informarnos de las actualizaciones del programa o de diversas ofertas de sus socios.

Debemos ser prudentes, ya que las actualizaciones pueden ser pertinentes pero dando nuestro correo electrónico y autorizando recibir anuncios de sus socios, podemos correr el riesgo de recibir *spam*.

Algunos programas se instalan de manera que se inician al arrancar (Dropbox, messenger, antivirus, etc.), lo cual no tiene ninguna utilidad, excepto en el caso de antivirus, antispywares o cortafuegos, ya que sólo ralentizan el inicio de Windows. Si durante la instalación se nos permite indicar esta opción debemos considerar si nos interesa o no. En otros casos, podemos deshabilitar el autoarranque en las opciones de configuración de la aplicación.



Si un programa se autoinicia al arrancar y, en sus parámetros de configuración no hay opción de deshabilitarlo, podemos hacer lo siguiente:

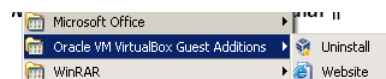
- Abre el menú Inicio y haz clic en Ejecutar
- Escribe `msconfig` (*Microsoft configurer*) y presiona `Enter`
- En la ventana que se abre, haz clic en la pestaña Inicio. Allí se encuentra la lista de los programas que están activos en el arranque.
- Desmarca todas las casillas de los programas deseados (excepto el antivirus y cortafuegos). Los nombres de los procesos no corresponden exactamente a los nombres comerciales.

## DESINSTALACIÓN

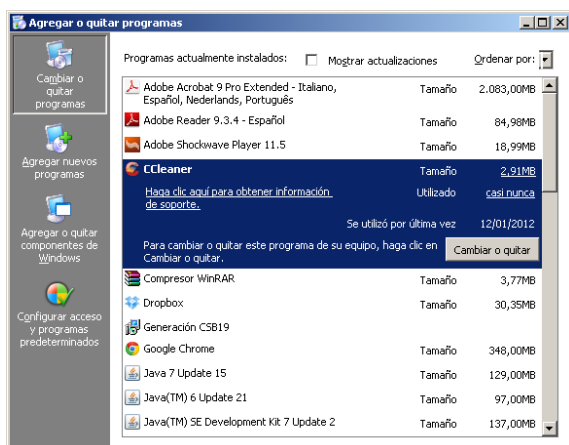
Hay cuatro procesos distintos a seguir para desinstalar un programa.

### 1. Utilizar el desinstalador del propio programa

Suele estar accesible en la carpeta de iconos de acceso al programa que se crea en el menú inicio-> todos los programas cuando se instala.

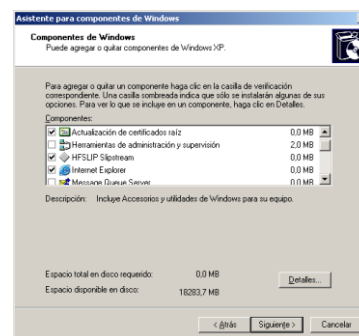


### 2. Usar el desinstalador de Windows



Accedemos a él mediante Inicio -> configuración -> Panel de Control -> Agregar o quitar programas

En esta ventana también tenemos la opción de agregar o quitar componentes Windows. Aquí están una serie de aplicaciones que no son programas propiamente dichos sino elementos de Windows que no se instalan automáticamente o que podemos desinstalar a voluntad, tales como Internet Explorer, el reproductor Windows Media o los servicios de Fax



### 3. Usar algún software específico para desinstalar

Existen aplicaciones como *CCleaner* que nos permiten desinstalar software igual que el desinstalador de Windows. No obstante, el uso de estas aplicaciones (tanto en la modalidad de desinstalación de software como de limpieza) requiere prudencia pues se pueden eliminar accidentalmente librerías o claves de registro necesarias para otro software.



### 4. Borrar manualmente

Es la opción menos segura de todas. Al instalar un programa se suelen instalar librerías que a veces son comunes a otros programas. Además, los instaladores añaden claves de registro específicas para la aplicación, tales como las extensiones de ficheros que utiliza, los enlaces a las librerías, etc. Borrar manualmente la carpeta de un programa no libera ni las librerías ni las entradas de registro.

Por eso, lo correcto al desinstalar un programa es intentarlo primero mediante su propio desinstalador, si no existe o no funciona, usar el de Windows. Si este también falla, recurrir a aplicaciones como *Ccleaner*. Sólo si nos falla todo esto podremos intentar un borrado manual, eso sí, sacando antes una copia de seguridad del sistema.

## 1.2. LINUX

Antes de entrar en materia recordar que **sólo el root puede instalar software**.

Un sistema de gestión de paquetes es una colección de herramientas que proporciona un método para la instalación, actualización y eliminación de software en su sistema operativo. Por lo general, las distribuciones de GNU/Linux consisten de miles de paquetes de software distintos.

El software se distribuye por medio de paquetes a los que están vinculados **metadatos** que contienen información suplementaria, como la descripción de la finalidad del software y una lista de dependencias necesarias para que el software funcione correctamente. Esos paquetes se proporcionan mediante los **repositorios**, ya sea mediante medios de almacenamiento local (CD, DVD o disco duro) o bien desde repositorios en Internet. Tras la instalación, los metadatos se almacenan en una base de datos local de paquetes que se utiliza para recuperar los paquetes del software.



Los paquetes son una colección de ficheros que incluyen todos los archivos necesarios que componen un software (como una aplicación en sí, librerías compartidas, los paquetes de desarrollo que contiene los archivos necesarios para construir software, biblioteca de utilidades, ...) y, finalmente, instrucciones sobre la manera de hacer que esos archivos funcionen. Un paquete se integra adecuadamente en la distribución para la que ha sido construido en lo que respecta a las rutas de instalación, las dependencias que requiere, la integración en el escritorio que utilice, etc. Por esta razón, siempre se deben instalar los paquetes que se han construido para la distribución esté utilizando, incluyendo la versión exacta de la distribución.

Un aspecto importante son las interacciones que contienen. Efectivamente, los paquetes también se relacionan con los archivos de otros paquetes, las aplicaciones empaquetadas necesitan un entorno de ejecución (otras herramientas, bibliotecas, etc.) para ejecutar correctamente la aplicación principal. Los **paquetes de dependencias** se utilizan para expresar estas relaciones. Las bibliotecas de dependencias (por lo general los paquetes con un nombre que comienza con `lib`) son muy comunes y casi cada aplicación depende de un conjunto de paquetes de bibliotecas.

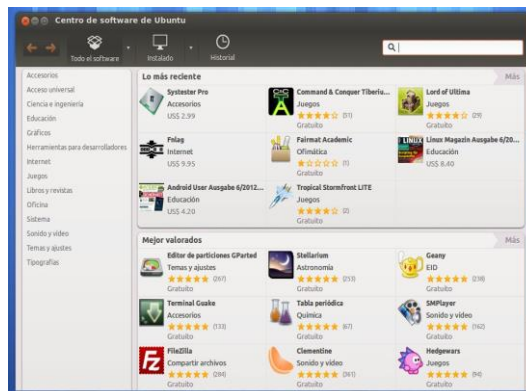
Las ventajas que tiene usar el sistema de paquetes es que te permite instalar, desinstalar y actualizar los programas de forma muy sencilla. Además si la aplicación que se instala depende de otros paquetes extra, el gestor los añadirá automáticamente.

Lo más fácil es usar el Gestor de paquetes que traen todas las distribuciones. Con él podemos instalar muchísimos programas de forma inmediata usando los repositorios<sup>1</sup> desde Internet o desde un CD. Por ejemplo, en Ubuntu puedes instalar aplicaciones yendo al *Centro de software* en el entorno gráfico.

<sup>1</sup> Un **repositorio** es un sitio centralizado donde se almacena y mantiene información digital, habitualmente bases de datos o archivos informáticos, en este caso, paquetes

También se puede optar por descargar los programas desde alguna página Web, para ello lo mejor es bajarlos en un formato específico para la distribución Linux en cuestión: en Debian y derivados (Ubuntu), se usa el formato de paquetes `.deb`. En Red Hat y otras (SuSe) se usan `.rpm`.

Estos programas son autoinstalables igual que en Windows.



### INSTALACIÓN DESDE EL CÓDIGO FUENTE

Los archivos `.tar`, `.tar.gz` o `.tar.bz2` son carpetas comprimidas que suelen traer el programa en código fuente. La ventaja es que sirven para todas las distribuciones y la desventaja que son más difíciles de instalar. Muchas veces basta con descomprimir la carpeta y ejecutar el archivo ejecutable que trae, aunque otras veces hace falta compilarlos desde un terminal (si eres un usuario nuevo mejor que huyas de esto). Hay programas en código fuente para descargar en [softonic.com/linux](http://softonic.com/linux).

Tras descomprimir estos archivos veremos una carpeta nueva con el código fuente, donde tendremos en cuenta los siguientes archivos: `README` e `INSTALL`. En uno de los dos encontraremos qué dependencias requiere el programa y qué comandos teclear en el terminal para compilarlo.

Las dependencias pueden ser librerías de lenguajes de programación, como Python, Perl o C++. Otras serán necesarias simplemente para compilar el código, como `make`, `cmake` o `gcc`. Y otras, simplemente, son librerías que realizan funciones por separado y en las que se apoya nuestro programa a compilar.

El principal problema viene cuando el autor del programa no nos da ninguna instrucción de qué dependencias requiere su creación. Por suerte, en algunos casos al intentar compilar veremos mensajes de error que, si nos fijamos bien, nos alertarán de que faltan ciertos paquetes.

En cualquier caso, hay que resolver las dependencias antes de comenzar la instalación.

Una vez resuelto el problema de las dependencias sólo tendremos que situarnos en la carpeta que se nos creó antes y teclear:

```
make install
```

Para desinstalar usaremos debemos ejecutar `make uninstall` y luego `make clean` pero dentro de la carpeta. Por ello es conveniente almacenar los paquetes que se descarguen de Internet en una carpeta específica por si es necesario desinstalarlos a posteriori.

Ejemplo de `make install` con el paquete del administrador de dispositivos múltiples:

Buscar el paquete correspondiente en Internet y bajarlo a nuestro sistema

```
mdadm-3.2.6.tar.gz
```

Descomprimir el paquete, situarse en el directorio que se crea y ejecutar el comando

```
tar xvf mdadm-3.2.6.tar.gz
```

```
cd mdadm-3.2.6
```

```
sudo make install
```

comprobar que se ha instalado el programa

```
ls /sbin/md*
```

retroceder al padre y borrar la carpeta y el paquete

```
cd ..
```

```
rm -r mdadm-3.2.6
```



## GESTORES DE PAQUETES EN MODO COMANDO

Ubuntu presenta dos herramientas para la gestión de paquetes en modo comando:

### apt-get

Utiliza la lista de repositorios definida en `/etc/apt/sources.list`

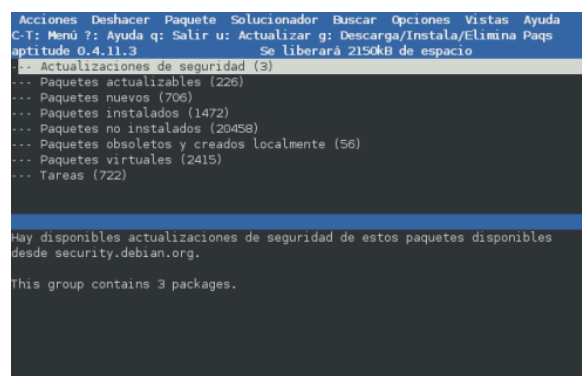
Comprobar que hay repositorios Web activos o, en su defecto, el del CD.

Acción	Comando
Actualizar los repositorios.	<code>apt-get update</code>
Actualiza los paquetes a una versión más nueva, tras usar <code>update</code> .	<code>apt-get upgrade</code>
Muestra los paquetes que contengan “palabra” en nombre o descripciones.	<code>apt-cache search palabra</code>
Información sobre el paquete.	<code>apt-cache show paquete</code>
Muestra los paquetes de los que depende <code>paquete</code>	<code>apt-cache depends paquete</code>
Instala el paquete con sus dependencias de forma ordenada.	<code>apt-get install paquete</code>
Baja el paquete y sus dependencias pero no se instala, esto es útil cuando queramos instalar estos paquetes en varios equipos.	<code>apt-get -d paquete</code>
Actualizar el paquete o solucionar los problemas que pueda tener.	<code>apt-get -reinstall install paquete</code>
Desinstalar el paquete junto con todas sus dependencias.	<code>apt-get remove paquete</code>
Esta opción además de borrar todos los paquetes y dependencias borra los archivos de configuración.	<code>apt-get -purge</code>

### aptitude

Aunque también funciona en modo comando como `apt-get`, si se pone solo `aptitude` se abre una interfaz semigráfica que es bastante cómoda de utilizar.

La diferencia principal entre `apt-get` y `aptitude` son las dependencias. Cuando se instala algún paquete, aparece en la Terminal que se van a instalar XX dependencias. Esta información la da el comando `apt-get` y `aptitude`.



Pero cuando se desinstala un paquete, lo que sucede es que `aptitude` “recuerda” esas dependencias que instalo y también las elimina. El comando `apt-get` no “recuerda” las dependencias, y por lo tanto no puede desinstalarlas, quedándose estas instaladas.

### zypper

Igual que `apt-get` pero para las versiones SuSe

`zypper [opciones] comando [opciones-comando] [parametros] ...`

Acción	Comando
Manejo de repositorios	<code>refresh, repos, addrepo, removerepo, modifyrepo, namerepo</code> <code>refresh-services, services, addservice, removeservice, modifyservice</code>
Manejo de paquetes	<code>ninstall, remove, source-install</code>
Manejo de actualizaciones	<code>patch, list-patches, patch-check, patches, update, list-updates, dist-upgrade</code>
Consultas	<code>search, info, what-provides, list-updates, patch-check, patches, packages, patterns, products</code>
Bloqueo	<code>locks, addlock, removelock, cleanlocks</code>
Utilidades	<code>verify, install-new-recommends</code> <code>help, licenses, versioncmp, targetos</code>

### YaST

SuSe tiene también un gestor de paquetes en modo texto y en modo comando incorporado a su utilidad YaST.

## 2. MANTENIMIENTO Y OPTIMIZACIÓN DE DISCOS

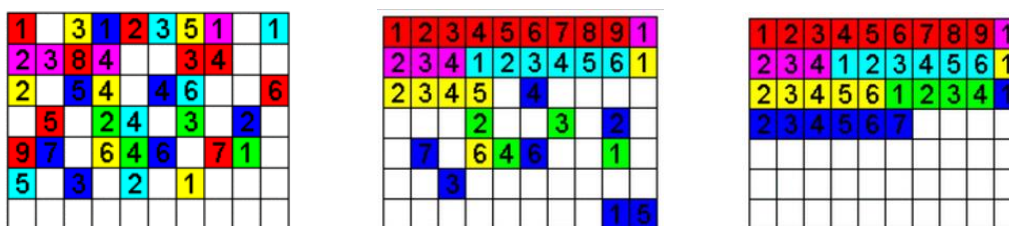
### 2.1. DESFRAGMENTACIÓN

**Importante:** las unidades SSD no deben ser desfragmentadas.

La fragmentación (almacenamiento no contiguo de bloques de datos de los archivos) es un problema que surge debido al ordenamiento interno de los datos en algunos sistemas de archivos.

Ciertos sistemas de archivos son más susceptibles a la fragmentación que otros. Por ejemplo, una partición del tipo FAT se fragmenta más rápido que una de partición del tipo NTFS (Windows), EXT 4 (Linux) o HFS + (MacOSX)

Al irse escribiendo y borrando archivos continuamente en el disco duro, los bloques de éstos tienden a no quedar en áreas contiguas, así, un archivo puede quedar "partido" en muchos pedazos a lo largo del disco, se dice entonces que el archivo está "fragmentado". Al tener los archivos esparcidos por el disco, se vuelve ineficiente el acceso a ellos



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

Desfragmentar no hace que el ordenador trabaje más rápido, sino que agiliza el proceso de la navegación por los archivos.

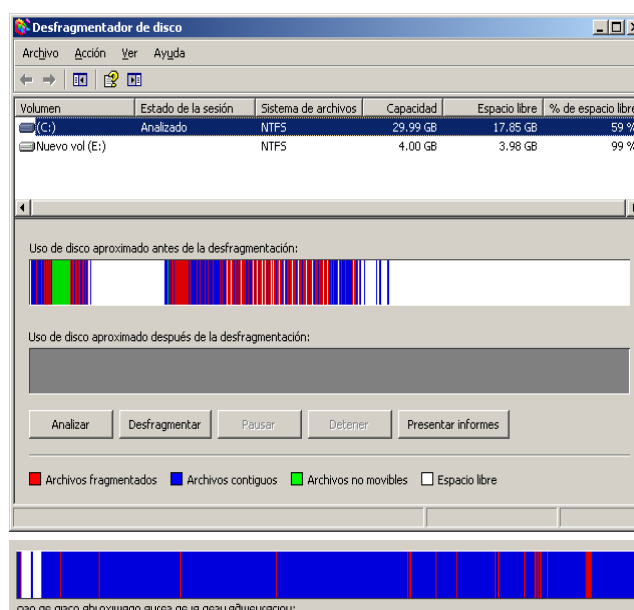
Para desfragmentar los discos Windows incorpora la herramienta `DEFRAG` a la que se accede en Inicio -> Programas -> Accesorios -> Herramientas de Sistema o bien Mi PC, clic derecho en unidad, Propiedades -> Herramientas.

Tras analizar la unidad, nos informará si es preciso desfragmentarla o no.

El proceso puede llevar cierto tiempo dependiendo de la cantidad de ficheros y el tamaño del disco.

En Linux la razón más importante para que no sea necesaria la desfragmentación es el hecho de que la mayoría de los ficheros del sistema necesitan permisos de superusuario para moverse de un lado a otro, así que normalmente los ficheros importantes no se mueven, y solo son nuestros directorios (en `/home`) los que puede que tengan más actividad.

La otra razón es que en los sistemas de ficheros de Linux la búsqueda de huecos para almacenar los ficheros es distinta, y es probable que la única situación en la que una fragmentación sea recomendable es cuando la unidad está llena a más del 95% de su capacidad (para lo que habrá que buscar las herramientas oportuna en Internet).



Aspecto de un disco que precisa desfragmentarse y otro que no

## 2.2. CHEQUEO Y REPARACIÓN DE DISCOS

**IMPORTANTE.** Si tras un apagado brusco del sistema al reiniciar nos avisa de que hay que revisar una o más unidades **NUNCA** debe cancelarse este proceso.

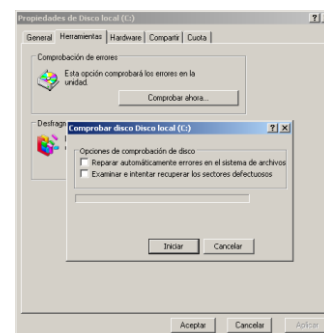
Tras un apagado brusco es posible que haya daños en algún sector del disco o, lo mas probable, no se hayan grabado en disco los últimos cambios realizados en el sistema de archivos (éste suele estar en RAM para agilizar el acceso a los ficheros y se graba periódicamente en disco) por lo que pueden presentarse errores lógicos como clústeres perdidos, archivos con vínculos cruzados o errores en directorios.

La comprobación del disco es imprescindible para corregir las incongruencias del sistema de archivos y, en su caso, marcar como defectuosos los sectores del disco que hayan podido resultar dañados con el fin de no volver a utilizarlos.

Tanto en Windows como en Linux, para realizar un chequeo del disco con verificación y/o corrección de errores hay que tener el volumen desmontado. Si se solicita dicho chequeo para el volumen del sistema, el proceso se realizará tras reiniciar y antes de cargar el sistema operativo.

### 2.2.1 WINDOWS

La utilidad `CHKDSK.EXE` muestra el estado y la integridad del sistema de archivo de los discos duros, memorias, tarjetas y otros medios de almacenamiento. Permite escanear, revisar y reparar problemas físicos en la superficie de discos duros como sectores defectuosos y recuperar los datos de ser posible. También es capaz de reparar errores lógicos del sistema de archivos. Hay 3 formas de usar la aplicación:



- Desde el entorno gráfico, aunque de forma limitada:
  - Mi PC -> unidad -> clic derecho, Propiedades -> Herramientas, botón *Comprobar ahora*, se abrirá la ventana **Comprobar Disco**, donde hay 2 opciones disponibles:
  - Reparar automáticamente errores en el sistema de archivos.
  - Examinar e intentar recuperar sectores defectuosos.
  - Utilizar la primera opción y después, en caso de que sea necesario, usar la segunda ya que ésta tarda bastante tiempo en completarse en discos grandes.
- Desde la línea de comandos o consola `CMD` mediante el comando `CHKDSK`, de esta forma se accede a sus opciones avanzadas.

`CHKDSK [unidad:] [opciones]`

Sin especificar ninguna opción `CHKDSK` revisa el disco y al final muestra el informe de su estado y de la integridad del sistema de archivos pero no realiza ninguna acción.

Opciones que se pueden emplear con `CHKDSK`:

- `/F` Corrige errores en el disco
- `/R` Encuentra sectores dañados y recupera la información que sea legible. Implica `/F`
- `/X` Fuerza al volumen a desmontarse si es necesario, es necesario usar con `/F`
- `/V` En FAT/FAT32 muestra la ruta completa y el nombre de cada archivo en el disco, si es NTFS muestra mensajes de limpieza si hay.
- `/I` (sólo NTFS) Realiza una comprobación menos exhaustiva de entradas de índice
- `/C` (sólo NTFS) Omite la comprobación de ciclos dentro de la estructura de carpetas

Los dos últimos reducen la cantidad de tiempo necesario para ejecutar `CHKDSK` ya que omiten ciertas comprobaciones en el volumen.

- Por último, posible ejecutarlo desde la Consola de recuperación.

## 2.2.2 LINUX

`fsck` es una utilidad Linux que se utiliza ante alguna inconsistencia del sistema de archivos para corregir los posibles errores que hubiese. Para verificar un sistema de archivos se aconseja hacerlo mientras este está desmontado. Generalmente se ejecuta automáticamente al inicio del sistema ante alguna anomalía:

```
fsck [-opciones] /dev/sdXXX (o hdXXX)
```

donde `sdXXX` o `hdXXX` por el nombre de la partición que queramos verificar.

Los parámetros básicos son:

- f forzar la verificación aunque todo parezca normal.
- p reparar automáticamente cualquier problema que pueda ser resuelto sin intervención humana
- c con verificación de solo-lectura para buscar bloques dañados.
- cc con verificación de lectura-escritura no-destructiva para buscar bloques dañados.
- k añadir a la lista de bloques dañados existente los nuevos bloques dañados
- y asume *yes* de respuesta para todas las preguntas que realice.
- v (*verbose*) despliega más información.
- r modo interactivo. Espera nuestra respuesta.
- C muestra una barra de de progreso.

También pueden usarse las `smartmontools`, previa instalación del paquete correspondiente, en el entorno gráfico.

### Ejercicios:

Investiga los comandos `DEFRAG`, `CHKDSK` y `fsck`

Busca e instala el *CCleaner* y algún programa para mantenimiento de discos. Fíjate bien en las ventanas que se muestran durante la instalación. Estudia que se puede hacer con dichas aplicaciones. Desinstálalas.

En Linux, busca e instala el PGP.

## 2.3. CIFRADO

El cifrado o encriptación es un método que permite aumentar la seguridad de un mensaje o de un archivo mediante la codificación del contenido, de manera que sólo pueda leerlo la persona que cuente con la clave de cifrado adecuada para descodificarlo. Proporciona un alto nivel de protección de la información en sistemas multiusuario.

### 2.3.1 WINDOWS

No es compatible con la compresión NTFS.

Windows tiene su propia herramienta de cifrado llama EFS, aunque solo funciona sobre volúmenes NTFS. Al copiar o mover archivos cifrados a volúmenes no NTFS se pierde el cifrado.

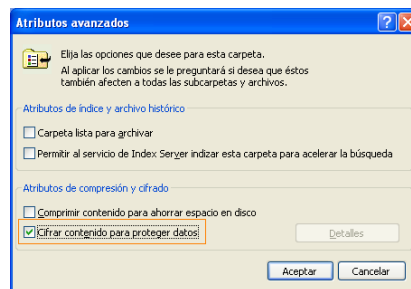
Al mover archivos sin cifrar a una carpeta cifrada se cifrarán automáticamente, aunque no ocurre lo mismo al pasar archivos cifrados a carpetas no cifradas.

Hay que tener en cuenta que para cifrar los archivos Windows utiliza como clave de cifrado el SID del usuario<sup>2</sup>. Si se borra un usuario que tuviese archivos cifrados no se podrán recuperar dichos archivos ya que, aunque creemos un nuevo usuario con el mismo nombre, el SID será distinto.

<sup>2</sup> El *Security Identifier* (Identificador de Seguridad) es el DNI que usa el sistema para referirse a las cuentas de usuario. Tiene el formato S-1-5-21-448539723-413027322-839522115-1003 y las tres cifras grandes se generan automáticamente y al azar cada vez que instalamos un XP. La última cifra será distinta para cada usuario que se cree en el sistema.

Para acceder a esta funcionalidad hay que situarse en la carpeta que se desee cifrar y pulsar el botón derecho del ratón -> Propiedades -> Opciones avanzadas... y finalmente marcar la casilla Cifrar contenido para proteger datos.

Al cifrar y descifrar directorios nos preguntará si deseamos actuar sólo sobre la carpeta o también sobre su contenido lo que indicaremos marcando la opción *Aplicar los cambios a esta carpeta, subcarpetas y archivos*.



Si se cifra sólo la carpeta, los ficheros que contenga permanecerán descifrados pero los nuevos que agreguemos se cifrarán.

Los directorios y archivos cifrados se muestran en color verde.

Para cifrar y descifrar en modo comando se usa el comando:

```
CIPHER {/E | /D } [/S:directorio] /A
```

Los parámetros básicos son:

- /E para cifrar.
- /D para descifrar
- /S: realiza la operación indicada el directorio indicado y en todos sus subdirectorios.
- /A actúa tanto sobre directorios como sobre los archivos que contenga.

## 2.3.2 LINUX

Linux no incorpora un sistema interno de cifrado por lo que hay que recurrir a herramientas externas al sistema operativo. Estas herramientas, por tanto, no utilizan ninguna clave interna del sistema sino que utilizan palabras clave que el usuario que no debe olvidar si desea recuperar la información.

Además, al contrario que Windows que descifra automáticamente el archivo al cargarlo en memoria si somos el usuario que lo cifro, para poder acceder al contenido del archivo hay que descifrarlos previamente, lo que puede hacer cualquier usuario que conozca la clave que se usó para encriptar.

gpg (*GNU Privacy Guard*) es una herramienta de encriptado y firmado para sistemas operativos Linux que suele venir incluida en todas las distribuciones.

Se puede usar para encriptar datos y para crear firmas digitales. Incluye una facilidad avanzada para el manejo de claves.

Para cifrar un archivo, se usa el comando gpg:

```
gpg [-c] archivo
```

-c para encriptar.

Tras pulsar **Enter** nos pedirá la contraseña de cifrado:

```
Enter passphrase: Clave
```

```
Repeat passphrase: Clave
```

Esto creará un nuevo archivo llamado `archivo.gpg`. El archivo original, sin cifrar, se mantiene por lo que hay que tener la precaución de borrarlo.

Cuando desenscriptamos (`gpg archivo`) nos mostrará:

```
gpg archivo.gpg
```

```
gpg: CAST5 encrypted data
```

```
Enter passphrase: Clave
```

Con lo que volveremos a tener el archivo sin encriptar y el encriptado.

## 2.4. COMPRESIÓN

Cuando hablamos de compresión no nos referimos a los conocidos archivos `zip` o `rar`, que se usan a través de herramientas de compresión como `Rar`, `p7Zip`, etc. sino a la capacidad de algunos sistemas de archivos de guardar automáticamente los archivos en disco de forma comprimida, restableciéndolos al usarlos, sin que el usuario tenga que hacer nada especial. Tanto Windows sobre NTFS como Linux nos ofrecen esta característica.

Esta facultad permite ahorrar espacio en disco si bien ralentiza los tiempos de acceso a los datos. Por ello es especialmente recomendable su uso con archivos grandes que se utilicen poco. Sin embargo, no conviene aplicarla sobre archivos de video, imágenes y audio ya que estos formatos de almacenamiento (`.avi`, `.mp4`, `.jpg`, `.bmp`, `.mp3`, etc) son codificaciones de por sí comprimidas por lo que las opciones de compresión no les afectan en tamaño resultante pero sí en tiempo de recuperación de los datos.

Para poder comprobar bien como trabaja la compresión conviene trabajar con archivos de distintos formatos y tamaños. Crea una carpeta y almacena en ella un video, una imagen y archivos de tamaño variado (desde menos de 1 bloque a varios) de documento (pdfs, texto, base de datos, etc.)

Toma nota del tamaño de cada uno de los archivos y de su ocupación en disco

(botón derecho -> propiedades -> Detalles)

Tamaño: 60,9 KB (62.367 bytes)

Tamaño en disco: 64,0 KB (65.536 bytes)

; stat fichero)

### 2.4.1 WINDOWS

No es compatible con el cifrado NTFS.

Para acceder a esta funcionalidad hay que situarse en la carpeta que se desee comprimir y pulsar el botón derecho del ratón -> Propiedades -> Opciones avanzadas... y finalmente marcar la casilla Comprimir contenido para ahorrar espacio en disco.

Al comprimir o descomprimir directorios nos preguntará si deseamos actuar sólo sobre la carpeta o también sobre su contenido lo que indicaremos marcando la opción *Aplicar los cambios a esta carpeta, subcarpetas y archivos*.

Si creamos o copiamos archivos, estos se comprimirán.

Si movemos archivos desde el mismo volumen permanecerán como estuviesen.

Si se comprime sólo la carpeta, los ficheros que contenga permanecerán sin comprimir pero los nuevos que agreguemos se cifrarán.

Los directorios y archivos comprimidos se muestran en color azul.

Para comprimir y descomprimir en modo comando se usa el comando:

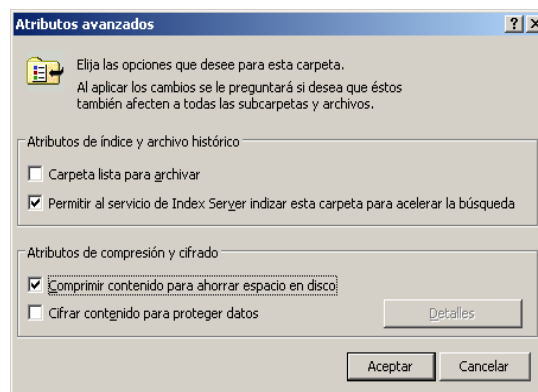
```
COMPACT {/C | /U } [/S:directorio]
```

Los parámetros básicos son:

`/C` para comprimir.

`/U` para descomprimir

`/S` realiza la operación indicada el directorio indicado y en todos sus subdirectorios.



## 2.4.2 LINUX

Linux incorpora la compresión automática de archivos en sistemas de archivos ReiserFS .

### Ejercicios:

Comprueba el funcionamiento del cifrado y comprimido de archivos en Windows tanto a través del entorno gráfico como en modo comando.

### Ejercicios:

1. Conecta a una máquina Linux 1 disco de 2GB con dos particiones: 1 primaria de 1GB y otra lógica (ambas sin formatear y creadas con alguna utilidad de Hiren's)
2. Comprobar si se han conectado correctamente los dos volúmenes.
3. Formatear la primaria en ext4 y la lógica en fat32.
4. Montarlos en los directorios `miext4` y `mifat` respectivamente.
5. Chequear el volumen `miext4`.

## 2.5. CUOTAS DE DISCO

Las cuotas de disco permiten controlar el uso del espacio de disco de los volúmenes por parte de los usuarios. Evitan que los usuarios monopolicen el disco.

No conveniente establecer cuota de disco en el volumen del sistema ya que al arrancar el sistema realiza distintas escrituras en el disco y si hay establecida una cuota y ésta llega a su límite, el sistema no arrancará.

Por eso, es conveniente utilizar dos volúmenes (uno para el sistema y otro para los datos) y establecer la cuota de disco en la correspondiente a los datos.

Los administradores de sistema puedan configurar el sistema para:

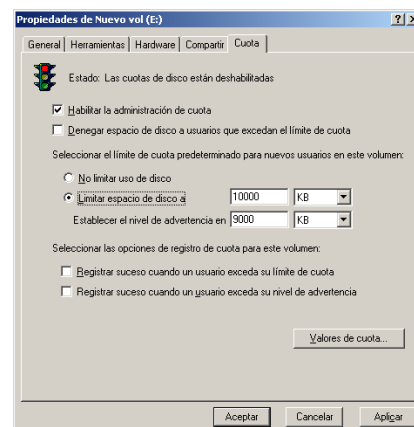
- Evitar que se utilice más espacio de disco del asignado y registrar un suceso cuando un Usuario sobrepase dicho límite.
- Registrar un suceso cuando un usuario sobrepase el espacio del nivel de advertencia.

### 2.5.1 WINDOWS

Desde la Administración de discos (o desde Equipo o Mi PC), sitúese en el volumen en el que desea establecer las cuotas de disco, clic derecho -> Propiedades -> Cuotas.

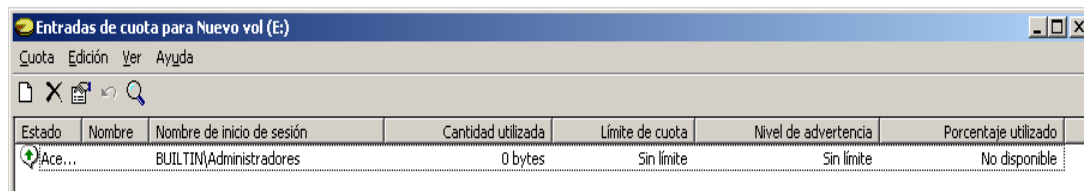
Al habilitar la administración de cuota podemos indicar las siguientes opciones:

- **Denegar espacio de disco a usuarios que excedan el límite de la cuota.** Al activar esta casilla se evita que los usuarios que hayan sobrepasado su límite de espacio en el volumen puedan grabar más archivos.
- **No limitar uso del disco.** Al activar esta casilla se permite asignar una cuota ilimitada de espacio en disco a los usuarios del volumen.
- **Limitar espacio de disco.** Al activar esta casilla se permite asignar un límite de espacio.
- **Establecer el nivel de advertencia.** Para indicar el nivel de espacio usado para que el sistema mande un aviso al usuario.



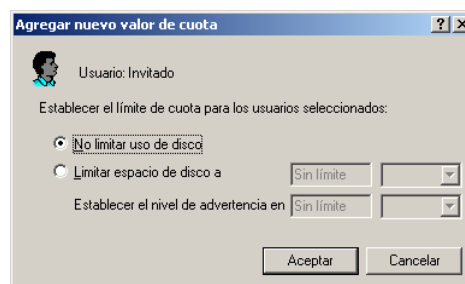


- **Registrar un evento cuando algún usuario supere su límite de cuota.** Se generará una entrada en el registro de eventos cuando algún usuario sobrepase su límite.
- **Registrar un evento cuando a algún usuario supere su nivel de advertencia.** Se generará una entrada en el registro cuando algún usuario sobrepase su nivel de advertencia.
- **Valores de cuota:** el límite de la pantalla anterior se aplicará a todos los usuarios. En esta nueva pantalla se pueden establecer límites distintos para cada usuario:



Estado	Nombre	Nombre de inicio de sesión	Cantidad utilizada	Límite de cuota	Nivel de advertencia	Porcentaje utilizado
	Ace...	BUILTIN\Administradores	0 bytes	Sin límite	Sin límite	No disponible

- Para añadir una entrada a la lista, abra el menú Cuota -> Nueva entrada de cuota -> Avanzadas -> Buscar ahora y se mostrará la lista con todos los nombres de usuarios. Seleccionar el deseado, pulsar Aceptar y verá que se añade a la lista inferior. Cuando haya acabado, pulsar Aceptar y se le mostrará la siguiente pantalla, donde podremos especificar las condiciones de cuota para dicho usuario.
- Podemos eliminar usuarios o modificar las condiciones de cuota a través del menú contextual de cada usuario.
- El símbolo que aparece a la izquierda de cada usuario nos indica:



El usuario está por debajo del nivel de advertencia



Se ha sobrepasado el nivel de advertencia pero aún no se ha superado el límite de cuota



Se ha sobrepasado el nivel de cuota

## 2.5.2 LINUX

Algunos sistemas de archivo permiten limitar el uso del disco a los usuarios y grupos.

Tipos de cuotas:

- **Por bloques (blocks):** Un bloque corresponde a 1 kb y una cuota por bloques correspondería al total de bloques que un usuario puede utilizar en el sistema.
- **Por i-nodos (inodes):** indica el total de i-nodos a los que el usuario tiene derecho, casi representaría el total de archivos que el usuario puede crear ya que podemos crear enlaces sobre archivos ya existentes que no aumentan la cantidad de i-nodos.

Límites de uso son:

- **Límite duro (hard):** se deniega cualquier intento de escribir datos después de este límite.
- **Límite débil (soft):** (siempre será menor que el límite duro) si la cuenta del usuario o del grupo supera el límite débil y no se tiene establecido un tiempo de gracia, el usuario podrá seguir usando bloques o i-nodos hasta llegar al límite *hard*.

Si se tiene establecido un tiempo de gracia el usuario podrá seguir usando bloques o i-nodos hasta que termine el tiempo o llegue al límite *hard*.

Las cuotas se establecen por sistema de archivos, es decir, debe decidirse en donde es más conveniente instalar un sistema de cuotas, pero no hay ningún problema si se instala en todos. Las cuotas pueden establecerse por usuario, por grupos o ambos.

Si el kernel está configurado para soportar cuotas, los pasos a seguir son:



1. Instalar el paquete `quota`
2. Modificar `/etc/fstab` para marcar los sistemas de archivo que tendrán cuotas:
 

```
/dev/sda1    /      ext3    defaults    0      2
/dev/sda3    /home  ext3    defaults,usrquota,grpquota 0      2
```
3. Reiniciar el sistema
4. El comando `quotacheck` (debe ejecutarse con las cuotas desactivadas) crea, verifica o repara el control de cuotas en los sistemas que lo soporten, en este caso creará el soporte. Al no existir un sistema de cuotas previo nos mostrará una serie de errores, lo que es normal. Cuando las cuotas estén en pleno uso, es conveniente ejecutar `quotacheck` periódicamente para que verifique inconsistencias y se corrijan a tiempo.  
 En el directorio raíz de cada sistema de archivos con cuotas aparecerán los ficheros binarios `aquota.user` y `aquota.group`, donde se guarda información sobre el sistema de cuotas.
5. Activar el sistema de cuotas para usuarios y grupos con `quotaon -ug /home`. Se podrá desactivar con `quotaoff`.
6. Usar el comando `edquota` para editar las cuotas de usuarios y grupos

### **APLICANDO LA CUOTA A USUARIOS**

Ahora hay que aplicar la cuota por usuario a través del comando `edquota`, que abrirá el editor de texto que se tenga por defecto y mostrará lo siguiente:

```
# edquota -u usuario
Disk quotas for user usuario (uid 502):
Filesystem blocks soft  hard  inodes soft  hard
/dev/sda3  56      0      0      14      0      0
```

Las columnas *blocks* e *inodes* son informativas y nos indican la cantidad de bloques o i-nodos utilizados actualmente por el usuario, y las que podemos editar son las columnas *soft* y *hard* de cada caso. Como ya se explicó en la primera parte de este artículo, se puede indicar libremente cualquiera de los cuatro valores, es perfectamente posible establecer valores por bloques, por i-nodos o ambos, solo recuerda que el límite *soft* debe ser menor al *hard*. Si se establece solo el *hard*, no habrá advertencias previas y el usuario ya no podrá guardar archivos cuando se llegue al valor. Si se establece *soft* y *hard*, avisará cuando se rebase el límite *soft* y entrará en juego el periodo de gracia. Si se acaba el tiempo de gracias o se llega al *hard* (lo que sea primero) ya no se podrán crear más archivos hasta que no se eliminen algunos de los que se tengan actualmente.

El comportamiento por defecto es modificar cuotas para ese usuario en todos los sistemas de archivos que tengan activo el control de cuotas (`quotaon`). Si se desea control de cuotas para un directorio concreto entonces se agrega la opción `-f`: (`edquota -u usuario -f /home`)

```
# edquota -u usuario
Disk quotas for user usuario (uid 502):
Filesystem blocks soft  hard  inodes soft  hard
/dev/sda3  56      100    120    14      0      0
```

Para modificar cuotas a nivel grupo, la opción `-g` (`edquota -g grupo`).

La opción `-p` copia la configuración de cuota de un usuario a los usuarios indicados.

### **VERIFICANDO EL USO DE LAS CUOTAS**

Como usuario administrador *root* puede ver el uso de cuotas de cualquier usuario, ya sea individualmente o por medio de un reporte global. Como usuario individual se usa el comando `quota`, si somos *root* podemos usar `quota usuario`:

```
# quota -s -u [usuario]
Disk quotas for user usuario (uid 502):
Filesystem blocks quota limit grace files quota limit grace
/dev/sda3  41582M      0    48829M   34905 0      0
```

Ahora bien, si se desea un informe global de las cuotas de todos los usuarios o por grupos, siendo *root* utiliza el comando *repquota*:

```
# repquota /home
*** Report for user quotas on device /dev/sda3
Block grace time: 7days; Inode grace time: 7days

              Block limits              File limits
User          used  soft  hard  grace  used  soft  hard  grace
-----
root          -- 34920    0    0          6    0    0
usuario       --   56   100  120         14    0    0
```

Con *repquota* es también posible utilizar la opción *-s* para observar los tamaños en formato legible. Si se usa la opción *-a (all)* en vez del sistema de archivos */home*, el informe será para todos los sistemas de archivos en el equipo que soporten cuotas. Así mismo este informe por defecto es por usuarios, si se requiere que *repquota* reporte por grupos, añade entonces la opción *-g*.

Obsérvese en la segunda línea del reporte el tiempo de gracia (*grace time*), que es de 7 días tanto para cuotas por bloque como para cuotas por archivos o i-nodos. Esto aplica para todos los usuarios en global, como se aprecia en el listado que ninguno tiene establecido un tiempo de gracia diferente al global.

### **ESTABLECIENDO EL TIEMPO DE GRACIA**

A nivel global, un periodo de gracia para todos, utiliza la opción *-t* del comando *edquota*, como en el siguiente ejemplo, recuerda que debes ser *root*:

```
# edquota -t
Grace period before enforcing soft limits for users:
Time units may be: days, hours, minutes, or seconds
Filesystem          Block grace period    Inode grace period
/dev/sda3            7days                7days
```

7 días es el periodo por defecto, si lo cambias a digamos 12 horas, sería *12hours*. El tiempo de gracia puede ser distinto para el límite *soft* por bloques o por i-nodos.

Por usuario específico se realiza con la opción *-T* del mismo comando e indicando el usuario:

```
# edquota -u usuario -T
Times to enforce softlimit for user usuario (uid 502):
Time units may be: days, hours, minutes, or seconds
Filesystem          block grace    inode grace
/dev/sda3            unset         unset
```

Cambiar el periodo y grabar

```
/dev/sda3            3days        unset
```

Lo único que hay que considerar es que los tiempos de gracias por usuario deben ser menores al global. Y que este empieza a correr una vez que se ha llegado al límite *soft*. Cuando esto suceda, si entras a editar de nuevo el tiempo de gracia del usuario (*edquota -u user -T*) se reflejara en segundos el tiempo que le queda, pudiéndolo aumentar de nuevo si eres *root*. O dejarlo en cero y entonces el global será el que se utilice.

```
# repquota /home
*** Report for user quotas on device /dev/sda3
Block grace time: 7days; Inode grace time: 7days

              Block limits              File limits
User          used  soft  hard  grace  used  soft  hard  grace
-----
root          -- 34920    0    0          6    0    0
usuario       --   56   100  120   3days   14    0    0
```

### **AVISOS DE CUOTAS EXCEDIDAS (WARNQUOTA)**

Cuando un usuario llega al límite suave o *soft* al crear o modificar un documento, algo como lo siguiente a aparecerá:

```
$ ls -l > directorio.txt
sda3: warning, user block quota exceeded.
```

En este instante como el usuario no ha llegado al límite *hard* ni ha expirado el tiempo de gracia, el sistema permite crear el archivo pero se le notifica con un *warning*.

Pero si lo que deseamos es notificar inmediatamente y vía correo electrónico que un usuario llega a su límite usaremos el comando `warnquota`. Este comando simplemente invocado desde la línea de comandos, sin argumentos, revisará los sistemas de archivos con cuotas activadas (`quotaon`) y revisará todos los usuarios buscando quien ha excedido el límite *soft* tanto por bloques como por i-nodos, y a aquellos que lo hayan excedido les enviará un correo notificándoles de lo anterior.

Puedes agregar en cron una línea como la siguiente para que `warnquota` haga su trabajo cada 12 horas:

```
# vi /etc/crontab
0 0,12 * * * root /usr/sbin/warnquota
```

`warnquota` viene con los mensajes en inglés por defecto, el archivo de configuración es `/etc/warnquota.conf`, es muy intuitivo y fácil de cambiar, personalízalo con los mensajes a español para que sea más fácil entender a tus usuarios que han excedido sus cuotas.

#### **Ejercicios:**

Para poder ver el funcionamiento de las cuotas y los avisos, crea un usuario nuevo y asígnale una cuota pequeña sobre un volumen que no contenga el sistema.

Prueba todos los comandos Linux.

## **3. COPIAS DE SEGURIDAD**

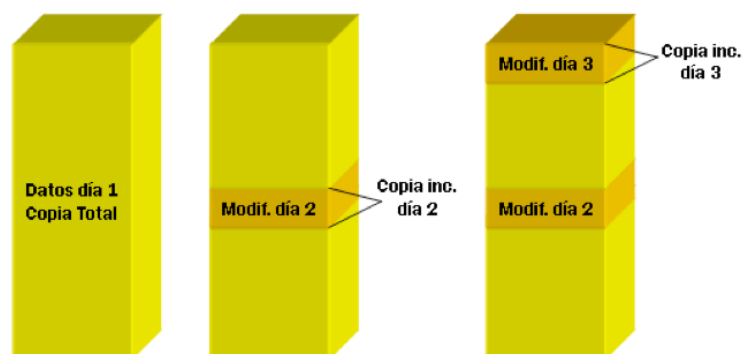
Suele ocurrir que no nos damos cuenta de que necesitábamos una herramienta de **backup** (**copia de seguridad**) hasta 10 segundos después de haber perdido todos los datos del disco, bien porque olvidamos esos datos a la hora de formatear, o por algún tipo de desastre.

Las causas que pueden provocar la pérdida de información son muy variadas, desde el mal funcionamiento de una aplicación hasta una rotura de un disco duro, pasando por todo tipo de programas maliciosos. Es por lo tanto imprescindible, planificar y llevar a cabo las tareas de prevención correspondientes. Para estar preparados ante cualquier desastre que elimine la información de los discos duros del servidor, debemos planificar una política de realización de copias de seguridad periódicas que salvaguarden tanto los datos de los usuarios como los archivos de la configuración del sistema y los servicios.

Para realizar una copia de seguridad debemos decidir el tipo de soporte donde vamos a almacenar los datos. Lo ideal es utilizar un medio de almacenamiento extraíble como cintas magnéticas, aunque es muy frecuente realizar las copias en discos duros. Actualmente están muy extendidos los discos extraíbles USB cuyas capacidades alcanzan los 32 GB o más, aunque al ser un dispositivo fácilmente manipulable, existe la posibilidad de un borrado fácil de la copia de seguridad en él almacenada.

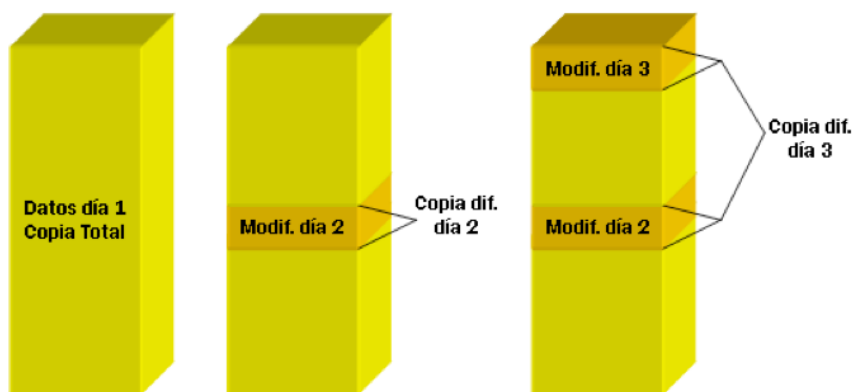
La segunda decisión que tomaremos es la planificación de la forma en que realizaremos la copia de seguridad. En función de la cantidad de datos a salvaguardar, podemos elegir entre tres tipos de tareas de copia de seguridad. Es importante seleccionar la tarea apropiada puesto que ello nos permitirá minimizar el número de cintas (u otros medios) y el tiempo empleado en realizar dicha tarea.

- Copia de seguridad **normal, total o íntegra**: una copia de todos los archivos y directorios seleccionados.
- Copia de seguridad **incremental**: se hace una copia de seguridad sólo de los archivos que han cambiado desde la última copia de seguridad realizada. Por ejemplo, si hacemos copia de seguridad total el día 1 de cada mes y copia de seguridad incremental el resto de los días, cada copia incremental solo guardará los archivos que se hayan modificado ese día. Si tenemos que realizar la restauración de archivos ante un desastre, debemos disponer de la copia total y de todas las copias incrementales que hayamos realizado desde la copia total.



Las copias incrementales guardan solo los archivos modificados desde la última copia incremental

- Copia de seguridad **diferencial**: es una copia de todos los archivos que han cambiado desde la última copia de seguridad total que hayamos hecho. Por ejemplo, si hacemos copia de seguridad total el día 1 de cada mes y copia de seguridad diferencial el resto de los días, cada copia diferencial guardará los archivos que se hayan modificado desde el día 1. La ventaja es que requiere menos espacio que la copia total y que en el proceso de restauración únicamente necesitaremos la última copia total y la última copia diferencial. Una copia diferencial anula a la copia diferencial anterior. Por el contrario, se consume más tiempo en realizar la copia y también más espacio que en el caso de copia incremental.



Las copias diferenciales guardan solo los archivos modificados desde la última copia total

### RECOMENDACIÓN SOBRE EL TIPO DE COPIA A REALIZAR

Si el volumen de datos de nuestra copia de seguridad no es muy elevado (menos de 100 MB), lo más práctico es realizar siempre copias totales ya que en caso de desastre, tan solo debemos recuperar la última copia.

Si el volumen de datos de nuestra copia de seguridad es muy elevado (varios GB) pero el volumen de datos que se modifican no es elevado (menos de 500 MB), lo más práctico es realizar una primera copia total y posteriormente realizar siempre copias diferenciales. Así, en caso de desastre, tan solo debemos recuperar la copia total y la última diferencial. Periódicamente debemos realizar una copia total y así empezar de nuevo.

Si el volumen de datos de nuestra copia de seguridad es muy elevado (varios GB) y el volumen de datos que se modifican también lo es, las copias diferenciales ocuparán mucho espacio, por lo tanto en este caso lo más práctico será realizar una primera copia total y posteriormente realizar siempre copias incrementales ya que son las que menos espacio ocupan. El problema es que en caso de desastre debemos recuperar la última copia total y todas las incrementales realizadas desde que se hizo la última copia total. En estos casos, conviene hacer copias totales más a menudo para no tener que mantener un número muy elevado de copias incrementales.

En grandes compañías donde la realización de copias de seguridad está perfectamente planificada, se suelen utilizar sistemas mixtos. Por ejemplo en un caso típico se realizarían las siguientes tareas:

- Todos los días 1 de cada mes, a las 23:00 horas: copia de seguridad total.
- Todos los viernes a las 23:00 horas: copia de seguridad diferencial desde la copia de día 1.
- Todos los días (excepto los viernes y el día 1) a las 23:00 horas: copia de seguridad incremental desde la copia del día anterior.

Con ésta planificación nos aseguramos disponer de copia de seguridad diaria. En caso de desastre deberíamos recuperar la copia total, la última diferencial y todas las incrementales desde la última diferencial. En una política de este tipo se pueden utilizar por ejemplo 5 juegos diferentes de cintas de forma que se almacenen las copias de seguridad diarias de los últimos 3 meses. Luego se van reutilizando pero no más de 20 veces ya que las cintas se deterioran y la fiabilidad disminuye.

Un sistema bien configurado tiene el sistema operativo y resto de software en un volumen y los datos de los usuarios en otro. En Linux serían los directorios `/` y `/home` y en Linux tendríamos el `so` en `C:` y los datos en `D:`:

Una vez instalado y actualizado el software y, en su caso, hecho la configuración básica del servidor hay que sacar una copia normal o una imagen del volumen del sistema.

A partir de aquí, debemos determinar cuáles son las carpetas que queremos salvaguardar en nuestro proceso de copias de seguridad. En un sistema informático que da servicio a usuarios, la información más importante es precisamente la información de los usuarios, por lo tanto, debemos salvaguardar la carpetas de estos `D:`, `\Documents and Settings\` o `/home`.

El objetivo de la realización de copias de seguridad es el reestablecimiento del servicio en el mínimo tiempo posible, por eso es conveniente realizar una copia de seguridad periódica (aunque con menos frecuencia que la de los datos de usuario) de los archivos de configuración, que se encuentran en el disco `C:` o en las carpetas `/etc`, `/var/log` (donde se almacenan las incidencias del sistema) y la carpeta personal del usuario `root` (`/root`)

Cuando realizamos copias de seguridad, los datos deben comprimirse siempre por tres razones:

- La copia se realiza más rápidamente
- El tamaño de la copia es menor
- La compresión garantiza la integridad de los datos

El volumen de los datos a copiar en el soporte de almacenamiento es mucho menor que lo que ocupan los datos descomprimidos; eso unido al hecho de que los datos estén compactados en un único archivo, hace que el tiempo en transmitir los datos desde el servidor al soporte, sea menor que si no se comprime. La integridad de los datos queda garantizada porque el algoritmo de compresión añade un código de redundancia cíclica (CRC) que se consulta a la hora de descomprimir los datos de forma que tenemos seguridad si están correctos o no lo están.

### **NOMBRE DE LOS ARCHIVOS RESULTANTES**

Normalmente, el nombre del archivo suele incluir el tipo de copia, las carpetas que contiene y la fecha (en el caso de copias totales) o fechas (en el caso de copias diferenciales e incrementales) de los datos.

Ejemplo, si hoy fuera 1 de febrero de 2015 y deseáramos crear una copia de seguridad total de las carpetas /etc y /home, lo normal es que el nombre del archivo fuera:

```
15feb01_CopiaTotal_etc-home.tar.bz2
```

Si una semana después, el 8 de febrero de 2015 deseáramos crear una copia de seguridad diferencial desde la copia total del día 1 de las carpetas /etc y /home, lo normal es que el nombre del archivo fuera:

```
15feb01-15feb08_CopiaDiferencial_etc-home.tar.bz2
```


Si el día siguiente, 9 de febrero de 2015, deseáramos crear una copia de seguridad incremental desde la copia diferencial del día 8 de las carpetas /etc y /home, lo normal es que el nombre del archivo fuera:

```
15feb08-15feb09_CopiaIncremental_etc-home.tar.bz2
```

Con esta nomenclatura será más fácil identificar los datos que contienen los archivos de copia de seguridad ya que el nombre del archivo lleva implícito el tipo de copia, las carpetas de datos que contiene y la fecha o fechas de los archivos salvaguardados.

### 3.1. WINDOWS

Para acceder al programa vamos a Inicio -> Programas -> Accesorios -> Herramientas del sistema ->

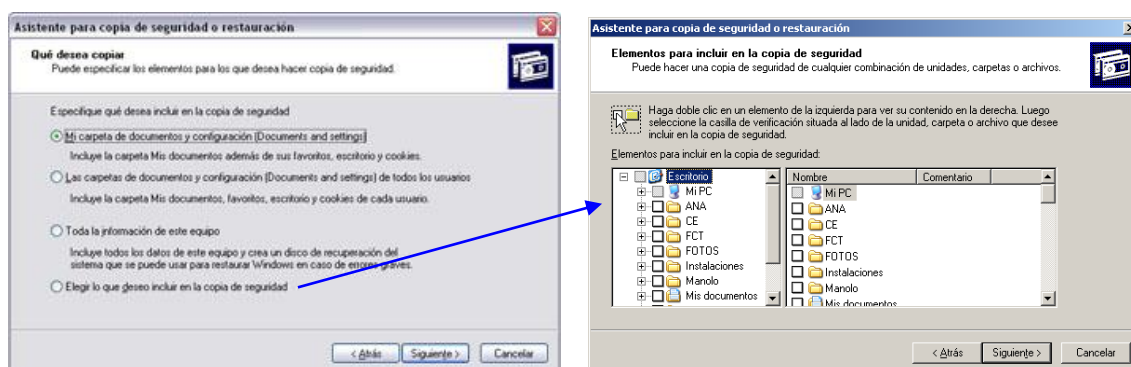
 Copia de seguridad

Una vez ahí tenemos esta pantalla:



Para una mayor sencillez de uso, se utiliza el programa en modo asistente. En la siguiente pantalla nos pregunta si queremos realizar una copia de seguridad o recuperar datos de una antigua.

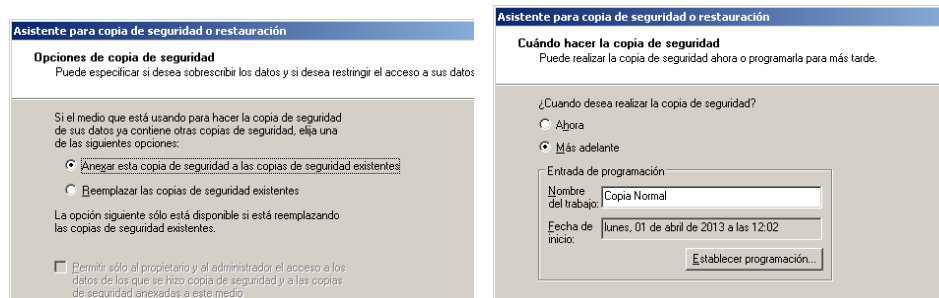
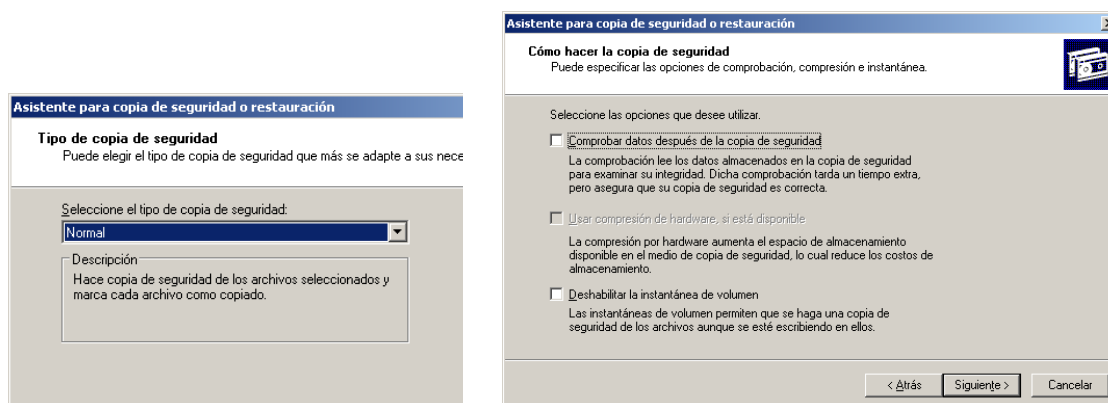
En la siguiente pantalla se indica que se quiere incluir en la copia: La carpeta Mis Documentos y la configuración de los programas (de un usuario o de todos), toda la información del equipo o libre selección por parte del usuario.



Elegimos una opción deseada y nos da a escoger donde queremos guardar la copia y el nombre del fichero.

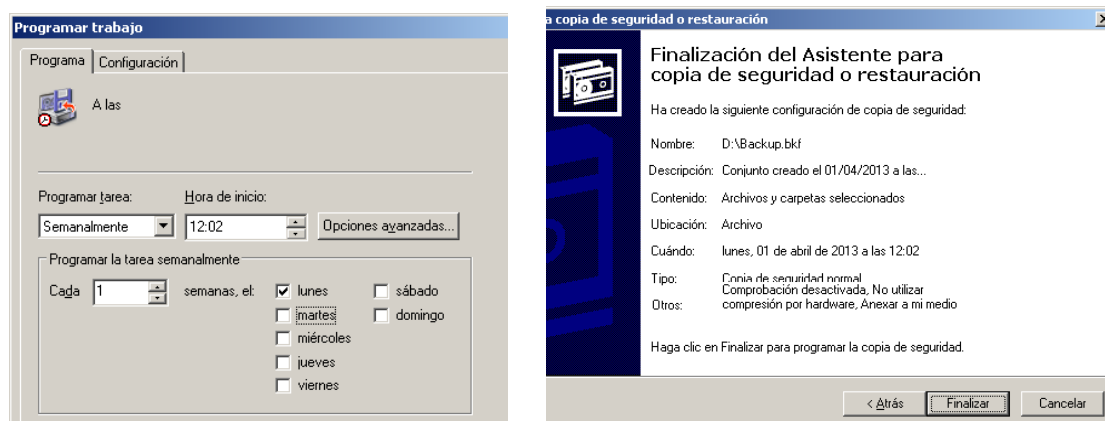


En la siguiente pantalla pulsamos *Opciones avanzadas* para poder definir el tipo de copia deseada y otras características:



En la última pantalla mostrada puedes optar por realizar ya la copia o programarla. En este último caso, si se pulsa *Establecer programación* nos abrirá el asistente de *Tareas programadas*.

Por último, nos muestra una pantalla resumen con las opciones que hemos seleccionado.



Cuando se realiza una copia normal o incremental, Windows elimina el atributo A (atributo de modificación que Windows pone a los ficheros al crearlos o modificarlos) de los archivos copiados por lo que no serán copiados en las próximas copias incrementales (que sólo considera los archivos que aún tienen el atributo A). En la copia normal copia todos los archivos de las carpetas indicadas y en la incremental sólo los que fueron modificados desde la última copia, ya fuese normal o incremental.

Al realizar copias de tipo copia, diaria o diferencial no toca los atributos por lo que los ficheros serán incluidos en las próximas copias.

La diferencia entre las copias diarias y copia consiste en la diaria sólo copia los archivos modificados en el día y en que en la copia se copian todos los archivos de las carpetas indicadas.

**Ejercicios:**

Con el asistente para copias (observa lo que ocurre con el atributo A antes y después de realizar cada paso):

Realiza una copia de seguridad normal de varias carpetas.

Crea nuevos ficheros y modifica alguno de los existentes. Realiza una copia diferencial.

Crea una copia incremental de los mismos directorios.

Borra una de las carpetas y restáurala desde la copia de seguridad.

Investiga el comando NTBACKUP.

Repite el ejercicio anterior usando el comando NTBACKUP

## 3.2. LINUX

### CREACIÓN MANUAL DE LA COPIA DE SEGURIDAD

Para crear copias de seguridad de una carpeta o carpetas, habitualmente se utiliza el comando `tar` que permite crear un único archivo que contenga todos los datos y además, permite comprimirlos en diferentes formatos.

```
tar opciones [-f fichero-destino] carpeta1 [carpeta2... carpetaN]
```

#### Opciones:

- `j`      Comprimir utilizando bzip2 (extensión tar.bz2, es una de las que más comprime)
- `z`      Comprimir con gzip (extensión tar.gz)
- `x`      Extraer (descomprimir)
- `v`      Mostrar los archivos añadidos/extraídos
- `c`      Crear nuevo archivo
- `f archivo`      Escribir hacia un archivo / Extraer desde un archivo
- `t`      mostrar el contenido
- `N fecha`      (newer) mas nuevo que fecha indicada a las 0 horas, 0 minutos

#### Ejemplos:

Utilización de tar para crear copia de seguridad de varias carpetas

```
tar -zcvf CopiaTotal.tar.gz carpeta1 carpeta2 carpeta3 ...
```

Para extraer los archivos que contiene el archivo tar.gz

```
tar -zxvf copia.tar.gz
```

Para extraer solo un archivo del archivo tar.gz

```
tar -zxvf copia.tar.gz ruta-del-archivo/nombre-del-archivo
```

Para ver una lista de los archivos que contiene el archivo tar.gz

```
tar -ztf copia.tar.gz
```

Crear backup de los archivos modificados tras una fecha dada

```
tar -zcvf CopiaDiferencial.tar.gz -N 1feb2015
```

Si hoy fuera 1 de febrero de 2015 y deseamos realizar una copia de seguridad total en la carpeta /tmp (temporal) de las carpetas /home y /etc, el comando que debemos lanzar será:

```
tar -zcvf /tmp/15feb01_CopiaTotal_etc-home_.tar.gz /home /etc
```

Pero si en lugar de escribir directamente 15feb01-15feb08 escribimos ``date +%y%b`01-`date +%y%b%d`` nos servirá el mismo comando para todos los días.



## **AUTOMATIZACIÓN**

Para lanzar la realización automática de copias utilizaremos `cron`. Cron es un servicio que nos permite lanzar comandos automáticamente los días y a las horas que deseemos. Cada usuario tiene su propio cron en el que puede configurar sus tareas programadas mediante el comando `crontab -e` o con alguna aplicación gráfica como `gnome-schedule`. En nuestro caso, como realizamos copia de seguridad de carpetas que solamente tiene acceso el usuario `root`, debemos programar la copia mediante el cron de `root`.

Si utilizamos el comando `date` podemos hacer que se ponga automáticamente la fecha actual en el nombre del archivo y nos servirá para cualquier día ya que tomará la fecha del sistema.

### **Ejemplo**

Si queremos que muestre la fecha en un formato especial como por ejemplo 15may31, debemos escribir `date +%y%b%d`.

```
tar -jcvf /tmp/CopiaTotal_etc-home_`date +%y%b%d`.tar.bz2 /home /etc
```

Si hoy fuera 8 de febrero de 2015 y deseáramos realizar una copia de seguridad diferencial de los cambios producidos desde el día 1 de febrero de 2015 en la carpeta `/tmp` de las carpetas `/home` y `/etc`, el nombre del archivo será `CopiaDiferencial_etc-home_15feb01-15feb08.tar.bz2` y el comando que debemos lanzar será:

```
tar -jcvf /tmp/CopiaDiferencial_etc-home_15feb01-15feb08.tar.bz2 /home /etc -N 01-feb-15
```

Supongamos que deseamos crear una copia de seguridad total los días 1 de cada mes y una copia de seguridad diferencial el resto de días en la carpeta `/tmp` (temporal), de las carpetas `/home` y `/etc`. El comando que ejecutaremos el día 1 de cada mes será:

```
// Comando a ejecutar los días 1 de cada mes
```

```
tar -jcvf /tmp/CopiaTotal_etc-home_`date +%y%b%d`.tar.bz2 /home /etc
```

Como puede verse, utilizamos ``date +%y%b%d`` que si hoy es 1 de febrero de 2015 se sustituirá por `15feb01`. De esta forma nos sirve el mismo comando para todos los meses.

El comando que ejecutaremos todos los días para realizar la copia diferencial, será:

```
// Comando a ejecutar los días para hacer copia diferencial respecto al día 1
```

```
tar -jcvf /tmp/CopiaDif_etc-home_`date +%y%b`01-`date +%y%b%d`.tar.bz2  
/home /etc -N `date +%y%b`01
```

Como puede verse, utilizamos ``date +%y%b`01-`date +%y%b%d`` que si hoy es 13 de febrero de 2015 se sustituirá por `15feb01-15feb13`. También en la opción `-N` ponemos ``date +%y%b`01` para que añada únicamente los archivos más nuevos que el día 1 del mes actual. De esta forma nos sirve el mismo comando para todos los días. Podemos crear scripts para guardar los comandos, ejemplo: `copia-normal.sh` y `copia-diferencial.sh`.

## **COPIAS DE SEGURIDAD EN SERVIDORES REMOTOS**

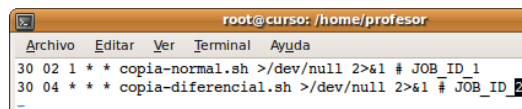
Lo comentado anteriormente permite realizar copias de seguridad en un disco duro local. Una mejora añadida sería la creación de la copia en una carpeta remota. Al igual que se automatiza la creación de la copia, se podría ejecutar automáticamente un comando que, vía `nfs`, `samba`, `ftp` o `ssh`, vuelque los archivos en un servidor remoto para mayor seguridad. También existen herramientas para realizar directamente copias de seguridad remotas:

`rsync`: permite realizar copias en carpetas remotas

`unison`: permite mantener sincronizadas dos carpetas remotas

## **AUTOMATIZACIÓN DE LAS COPIAS DE SEGURIDAD**

Al final nuestro archivo cron para que se ejecuten automáticamente los comandos que realizan las copias de seguridad quedará como el de la figura:



### **APLICACIONES PARA LA REALIZACIÓN DE COPIAS DE SEGURIDAD**

Existen aplicaciones, tanto libres como de pago, que facilitan la tarea de realización de copias de seguridad. Entre las aplicaciones libres para hacer copias de seguridad de PCs de la red destacamos:

- BackupPC
- Amanda
- afbackup

Estas aplicaciones tienen la ventaja de ser muy completas ya que disponen de un sinfín de posibilidades, pero son más complejas de manejar.

## **4. GESTIÓN DE USUARIOS**

En la mayoría de los sistemas operativos actuales, aparecen dos conceptos relacionados con la seguridad del sistema: Autenticación y Autorización.

- **Autenticación:** Para usar el sistema es necesario abrir una sesión de trabajo (login) para lo cual tendremos que autenticarnos, proporcionando al sistema un nombre de usuario y una contraseña. En caso de no tener una cuenta de usuario abierta en el sistema, será imposible entrar en el mismo.
- **Autorización:** Una vez que el usuario se ha autenticado y abierto sesión, cada vez que quiera usar un recurso (un fichero, una carpeta, una impresora, etc) el sistema comprobará si está autorizado o no para realizar esa acción. Los administradores del sistema pueden modificar estas autorizaciones mediante unas listas de acceso.

### **4.1. CONTRASEÑAS**

El uso de contraseñas para la autenticación de los usuarios es un método básico de control de la seguridad en sistemas. Tanto los equipos conectados a la red como los dispositivos de interconexión, necesitan el uso de contraseñas de acceso, para evitar que usuarios no autorizados puedan hacer uso de los equipos o modificar su configuración.

El proceso de autenticación de un usuario en un equipo no genera ningún riesgo si la contraseña permanece en ese equipo, aunque debe establecerse algún mecanismo de seguridad para que un usuario no pueda consultar la contraseña de otro. Sin embargo, cuando el proceso de autenticación requiere de la intervención de otro equipo (por ejemplo, un servidor que compruebe los usuarios y las contraseñas), entonces las contraseñas deben ser enviadas por la red. Esto supone un riesgo de seguridad, ya que otro usuario situado en otro equipo puede interceptar y capturar esas contraseñas. Esta situación también se produce cuando se accede a la configuración de un dispositivo de interconexión a través de la conexión de red.

Almacenar las contraseñas o enviarlas por la red supone un riesgo, por lo que se recomienda cifrarlas. Sin embargo, hay que tener en cuenta que todavía existen protocolos que no utilizan el cifrado de las contraseñas, con lo que habrá que tener cuidado en su uso.

La elección de una contraseña compleja resulta de vital importancia debido a la posibilidad de que un atacante pueda realizar un ataque basado en diccionario. Esta técnica consiste en utilizar un programa que prueba una lista de palabras, llamada diccionario, hasta obtener aquella que coincide con la contraseña. Su uso está condicionado por el hecho de que las técnicas de cifrado no son reversibles.

Otra cuestión importante que hay que tener en cuenta es el hecho de que muchos equipos facilitan las listas de usuarios registrados, sobre todo aquellas cuentas de usuarios que se definen de forma predeterminada (Administrador, admin, root, etc.). En principio, esto puede no plantear ningún problema de seguridad si no se conocen las contraseñas, pero su conocimiento puede ayudar a romper la seguridad si se utilizan otras técnicas para conseguir las contraseñas.

Por todas estas razones, no se recomiendan las siguientes prácticas:

- Utilizar como contraseña el mismo nombre de usuario.
- Definir la contraseña de acuerdo con información personal que pueda ser fácilmente investigada.
- Utilizar palabras del diccionario como contraseñas, ya que se pueden descifrar utilizando programas de fuerza bruta.
- Siempre se recomienda utilizar contraseñas que estén compuestas de letras y números entremezclados, así como utilizar signos o mezclar letras mayúsculas con minúsculas.

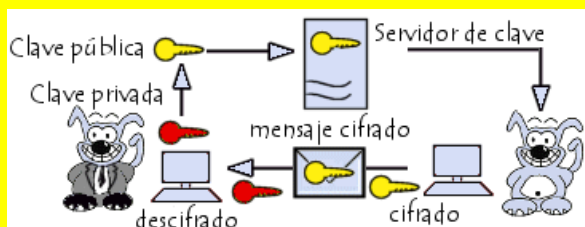
### TÉCNICAS DE CIFRADO

En un criptosistema asimétrico (o *criptosistema de clave pública*), las claves se dan en pares:

- Una clave pública para el cifrado;
- Una clave secreta para el descifrado.

En un sistema de cifrado con clave pública, los usuarios eligen una clave aleatoria que sólo ellos conocen (ésta es la *clave privada*). A partir de esta clave, automáticamente se deduce un algoritmo (la clave pública). Los usuarios intercambian esta clave pública mediante un canal no seguro.

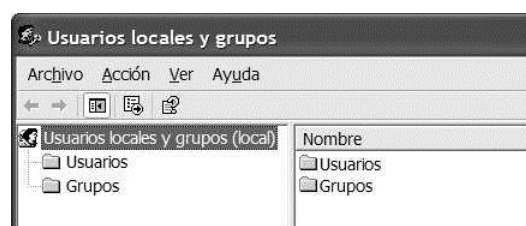
Cuando un usuario desea enviar un mensaje a otro usuario, sólo debe cifrar el mensaje que desea enviar utilizando la clave pública del receptor (que puede encontrar, por ejemplo, en un servidor de claves como un directorio LDAP). El receptor podrá descifrar el mensaje con su clave privada (que sólo él conoce).



Para ilustrarlo con un ejemplo, sería como si un usuario creara de forma aleatoria una pequeña llave metálica (la clave privada) y luego produjera una gran cantidad de candados (claves públicas) que guarda en un casillero al que puede acceder cualquiera (el casillero sería el canal no seguro). Para enviarle un documento, cada usuario puede usar un candado (abierto), cerrar con este candado una carpeta que contiene el documento y enviar la carpeta al dueño de la clave pública (el dueño del candado). Sólo el dueño podrá abrir la carpeta con su clave privada.

## 4.2. WINDOWS

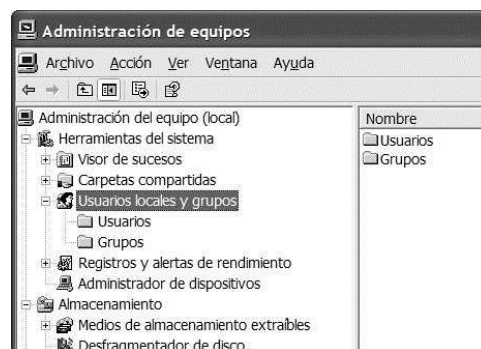
### 4.2.1 GESTIÓN DE CUENTAS DE USUARIO Y GRUPOS.



La mejor opción que tenemos para gestionar cuentas de usuario, es la consola de usuarios locales y grupos. Podemos llegar a dicha consola de varias formas.

- Podemos ejecutar desde Inicio → Ejecutar y escribir `LUSRMGR.MSC`.
- Desde Panel de Control → Herramientas Administrativas → Administración de Equipos y en ella escogemos la carpeta de usuarios locales y grupos.

Si fijamos, veremos que aparece la palabra local. Esto es así porque Windows distinguen dos ámbitos al hablar de usuarios: los usuarios locales y los usuarios de dominio. Mientras no tengamos instalado un dominio (para lo cual necesitaremos algún servidor Windows de la familia NT) siempre estaremos trabajando con cuentas locales.



Lleguemos desde donde lleguemos, veremos que tenemos dos carpetas, una para los usuarios y otra para los grupos. Podemos crear usuarios nuevos accediendo a las propiedades de la carpeta usuarios (botón derecho sobre ella) y seleccionando la opción de Usuario Nuevo. Podemos modificar un usuario accediendo a sus propiedades. Del mismo modo podemos crear nuevos grupos y modificar los ya existentes.

Podemos tanto asignar a un usuario varios grupos, como asignar a un grupo varios usuarios.

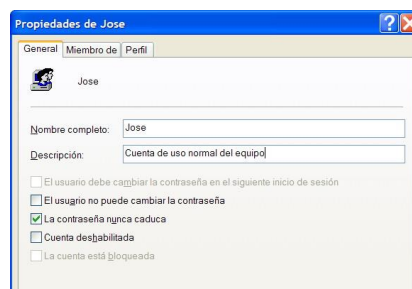
Podemos crear todas las cuentas de usuarios que deseemos, pero aparte de estas cuentas normales, existen dos cuentas de usuario especiales en Windows XP, ya creadas y que no pueden (no deben) ser modificadas o eliminadas.

- La cuenta del **Administrador del sistema** (Administrador). Todos los sistemas XP tienen una cuenta especial conocida como Administrador. Esta cuenta tiene todos los derechos sobre todo el equipo. Puede crear otras cuentas de usuario y es el responsable de gestionar el sistema. Muchas funciones del sistema están limitados para que sólo puedan ser ejecutadas por el Administrador. Es posible crear cuentas de usuario y darles derechos de administrador (integrándolas como miembros del grupo Administradores), aunque Administrador solo puede haber uno. Esta cuenta siempre debe contar con contraseña y se crea en el momento de la instalación del sistema.
- La cuenta de **Invitado**. (Guest). Es la contraria a la cuenta de Administrador, está totalmente limitada, no cuenta apenas con ningún permiso o derecho pero permite que cualquier usuario pueda entrar en nuestro sistema sin contraseña (lo que se denomina acceso anónimo) y darse un “paseo” por el mismo. Por defecto, en Windows XP Profesional esta cuenta esta desactivada. Es altamente recomendable nunca activar dicha cuenta, ya que representa un riesgo altísimo de seguridad.

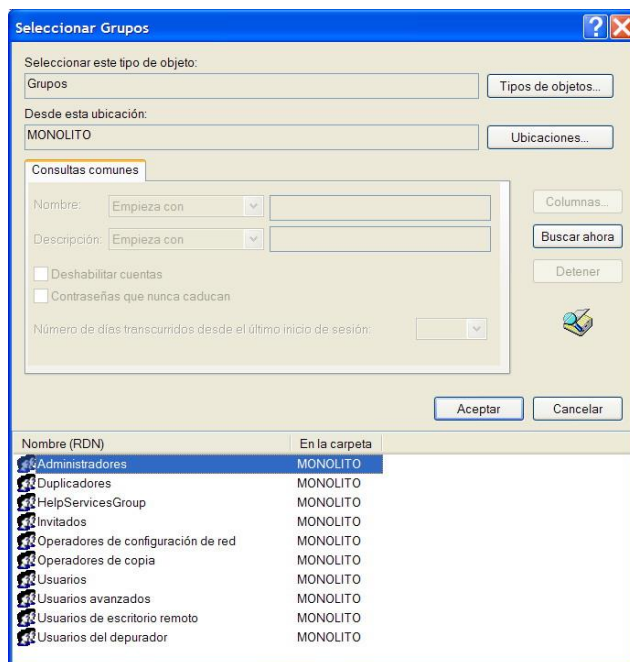
Si accedemos a las propiedades de un usuario, veremos tres pestañas con las que trabajar:

- **General:** Podemos indicar el nombre completo de la cuenta, una descripción, e indicar algunas opciones de la cuenta.
  - El usuario debe cambiar la contraseña en el siguiente inicio de sesión. Cuando el usuario inicie sesión la próxima vez se verá obligado a cambiar su contraseña.
  - El usuario no puede cambiar la contraseña. Prohibimos que el usuario pueda cambiar su contraseña.
  - La contraseña nunca caduca. Ya veremos que en Windows XP las contraseñas se consideran material fungible, es decir, que tras un cierto tiempo de uso el sistema obligará a cambiar dichas contraseñas. Mediante esta opción indicamos que la contraseña podrá usarse sin que caduque nunca.
  - Cuenta deshabilitada: No borra la cuenta, pero impide que sea usada. Es el estado por defecto de la cuenta Invitado.
  - La cuenta está bloqueada: Por determinados mecanismos de seguridad que ya veremos, se puede llegar a bloquear una cuenta, que implicará que dicha cuenta estará deshabilitada. Desde esta opción podemos volver a desbloquearla, simplemente desmarcando la casilla.
- **Miembro de:** aquí podemos introducir al usuario en grupos. Los grupos se usan para dar permisos y derechos a los usuarios fácilmente, sin tener que ir usuario por usuario. Así por ejemplo, si introducimos a un usuario como miembro del grupo Administradores, le estaremos dando todos los permisos del grupo Administradores.

En la pestaña miembro de veremos todos los grupos a los que el usuario pertenece actualmente. Si le damos al botón **agregar** podremos escribir directamente el nombre de un grupo donde agregarlo. Si queremos escoger dicho grupo de una lista de los grupos posibles, hay que escoger la opción **Avanzada** y luego **Buscar ahora**, que nos mostrará una lista



de todos los grupos del sistema. Basta con seleccionar el que queramos (o los que queramos) y pulsar aceptar.



- **Perfil:** Nos permite indicar la ruta del perfil, los archivos de inicio de sesión y las carpetas personales del usuario. Como en un apunte posterior veremos el tema de perfiles, de momento lo dejamos pendiente.

#### DESDE EL CONSOLA DE COMANDOS: NET USER

Esta opción puede parecer la más engorrosa, pero resulta ser la más práctica en muchísimas ocasiones, sobre todo si conocemos como hacer scripts de sistema.

Agrega o modifica cuentas de usuario o muestra información acerca de ellas.

```
net user [nombreDeUsuario [contraseña | *] [opciones]] [/domain]
net user nombreDeUsuario {contraseña | *} /add [opciones] [/domain]
net user [nombreDeUsuario [/delete] [/domain]]
```

- **NombreDeUsuario** Especifica el nombre de la cuenta de usuario que se desea agregar, eliminar, modificar o ver. El nombre de la cuenta de usuario puede tener hasta 20 caracteres.
- **Contraseña** Asigna o cambia una contraseña para la cuenta de usuario. Escriba un asterisco (\*) si desea que se le pida la contraseña. Los caracteres de la contraseña no se muestran en la pantalla a medida que los escribe.
- **/domain** Realiza la operación en el controlador principal del dominio principal del equipo.
- **Opciones:** Especifica una opción de la línea de comandos. La tabla siguiente enumera las opciones válidas de la línea de comandos que puede utilizar:

Sintaxis de las opciones de la línea de comandos	Descripción
/active:{no   yes}	Habilita o deshabilita la cuenta de usuario. Si no está activa, el usuario no puede tener acceso a los recursos del equipo. La opción predeterminada es <b>yes</b> (activa).
/comment:"texto"	Proporciona un comentario descriptivo acerca de la cuenta de usuario. Puede tener hasta 48 caracteres. Escriba el texto entre comillas.
/countrycode:nnn	Usa los códigos de país o región del sistema operativo para instalar los archivos de Ayuda y mensajes de error en el idioma especificado. 0 significa el código de país predeterminado.
/expires: {fecha   never}	Provoca que la cuenta de usuario caduque si se especifica <i>fecha</i> . La fecha puede tener el formato [mm/dd/aaaa], [dd/mm/aaaa] o [mmm,dd,aaaa],

Sintaxis de las opciones de la línea de comandos	Descripción
	según el código de país o región. Tenga en cuenta que la cuenta caduca al comienzo de la fecha especificada.
/fullname:"nombre"	Especifica un nombre de usuario completo en lugar de un nombre de usuario normal. Escriba dicho nombre entre comillas.
/homedir:rutaDeAcceso	Establece la ruta de acceso del directorio particular del usuario. Dicha ruta de acceso debe ser una ya existente.
/passwordchg:{yes   no}	Especifica si los usuarios pueden cambiar su contraseña. La opción predeterminada es <b>yes</b> .
/passwordreq:{yes   no}	Especifica si una cuenta de usuario debe tener una contraseña. La opción predeterminada es <b>yes</b> .
/profilepath:[rutaDeAcceso]	Establece una ruta de acceso al perfil de inicio de sesión del usuario. Esta ruta de acceso señala a un perfil de registro.
/scriptpath:rutaDeAcceso	Establece una ruta de acceso a la secuencia de comandos de inicio de sesión del usuario. El parámetro <i>rutaDeAcceso</i> no puede ser una ruta de acceso absoluta. <i>RutaDeAcceso</i> es relativa a %raízSistema%\System32\Repl\Import\Scripts.
/times:{dia,hora  all}	Especifica las horas en las que se permite al usuario el uso del equipo. El parámetro <i>Hora</i> está limitado a incrementos de 1 hora. Para los valores de <i>día</i> , puede escribir el día o usar abreviaturas (L, Ma, Mi, J, V, S, D). Para las horas puede usar la notación de 12 horas o de 24 horas. Para el formato de 12 horas, use am, pm, a.m. o p.m. El valor <b>all</b> significa que un usuario puede iniciar una sesión en cualquier momento. Un valor nulo (en blanco) significa que un usuario nunca puede iniciar la sesión. Separe el día y la hora mediante comas y las unidades de día y hora con punto y coma (por ejemplo, <b>L,4AM-5PM;Ma,1AM-3PM</b> ). No use espacios en la especificación de horas.
/usercomment:"texto"	Especifica que un administrador puede agregar o cambiar el "Comentario de usuario" de la cuenta. Escriba el texto entre comillas.
/workstations:{Equipo[,...] *}	Enumera hasta ocho estaciones de trabajo desde las que un usuario puede iniciar una sesión en la red. Separe los nombres de las estaciones con una coma. Si <b>/workstations</b> no es una lista o ésta es igual a un *, el usuario puede iniciar una sesión desde cualquier equipo.

Utilizado sin parámetros, `net user` muestra una lista de las cuentas de usuario en el equipo.

Para obtener ayuda, `net help user`.

La contraseña debe tener la longitud mínima establecida con `net accounts /minpwlen`. Puede tener hasta 127 caracteres.

Para mostrar una lista de todas las cuentas de usuario del equipo local, escriba:

```
net user
```

Para ver información acerca de la cuenta de usuario `juanh`, escriba:

```
net user juanh
```

Para agregar una cuenta de usuario para Enrique Pérez, con derechos de inicio de sesión desde las 8 a.m. a 5 p.m. de lunes a viernes (sin espacios en las especificaciones de las horas), una contraseña obligatoria (`enriquep`) y el nombre completo del usuario, escriba:

```
net user enriquep enriquep /add
    /passwordreq:yes /times:lunes-viernes,8am-5pm
    /fullname:"Enrique Pérez"
```



## 4.2.2 GESTIÓN DEL INICIO DE SESIÓN DE LOS USUARIOS

Desde Inicio → Ejecutar o bien desde una ventana del intérprete de comandos:

CONTROL USERPASSWORDS2

Con este gestor de cuentas de usuario, tenemos un control mucho mayor sobre las cuentas de usuario que el que obtenemos con el asistente. Si estamos conectados a un dominio, este es el gestor de usuarios que usaremos por defecto.

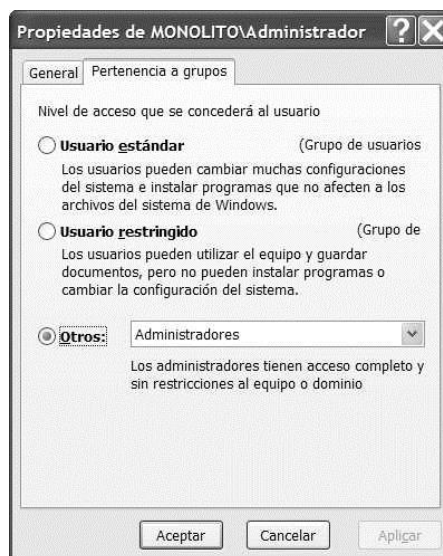
La primera opción que vemos en pantalla, “Los usuarios deben escribir su nombre y contraseña para usar el equipo” nos permite indicar si queremos usar la autenticación o no. Si lo desactivamos, obtendremos un XP que se iniciará automáticamente con la cuenta que indiquemos sin mostrarnos siquiera la pantalla de bienvenida. Obviamente, esta opción solo debería usarse en ambientes domésticos donde solo un usuario usa el ordenador.

Para agregar cuentas de usuario usamos el botón agregar, para eliminar cuentas el botón quitar, etc. Si seleccionamos una cuenta de usuario y pulsamos el botón propiedades, pasamos a la siguiente pantalla.

Desde esta pantalla, podemos incluir al usuario en algún grupo de usuarios, bien uno de los dos incluidos en el gestor (usuarios estándar y usuarios restringidos) o bien seleccionando otro grupo como puede ser el de administradores, etc. Este control de grupos es mucho más amplio que el que nos ofrece el asistente donde las opciones son usuarios administradores o usuarios restringidos.

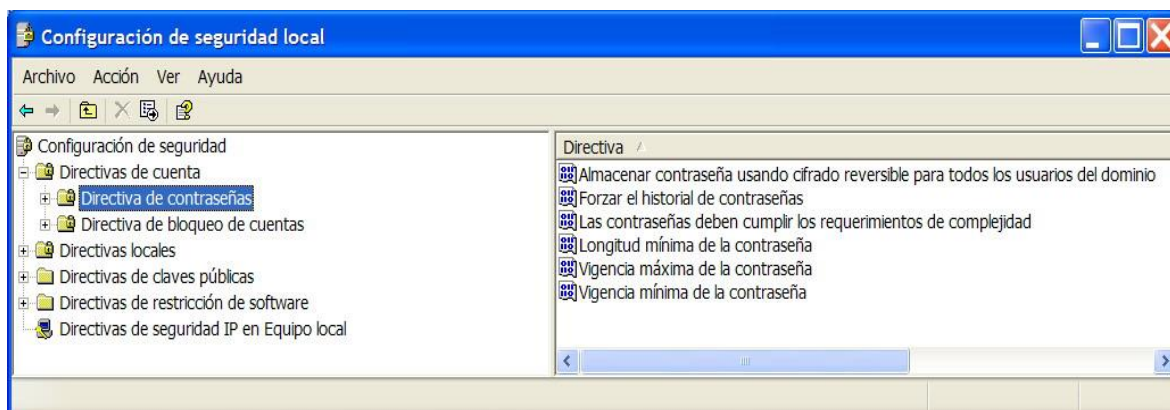
En las distintas pestañas de este gestor, podemos ver opciones muy interesantes como el Passport de MSN, opciones para modificar el inicio del sistema, opciones para almacenar las contraseñas de nuestro equipo, etc.

Una opción muy interesante que nos podemos encontrar en la primera pantalla (y que es un fallo de seguridad tremendo por parte de Microsoft, que no sería admisible en un sistema servidor) es la posibilidad de cambiar la contraseña del Administrador. Para hacerlo, basta con que nos hayamos autenticado con una cuenta de usuario que pertenezca al grupo Administradores. Esta opción ha sido incluida ya que a veces los usuarios no recuerdan con el tiempo la contraseña que le asignaron a la cuenta de Administrador en el momento de la instalación del equipo.



## 4.2.3 GESTIÓN DE LAS CONTRASEÑAS.

Los sistemas operativos de la familia Windows NT son sistemas muy configurables por parte del usuario. Aunque estas configuraciones suelen estar algo ocultas para que no sean accesibles por los usuarios normales, y solo pueden ser modificadas desde las consolas del sistema.



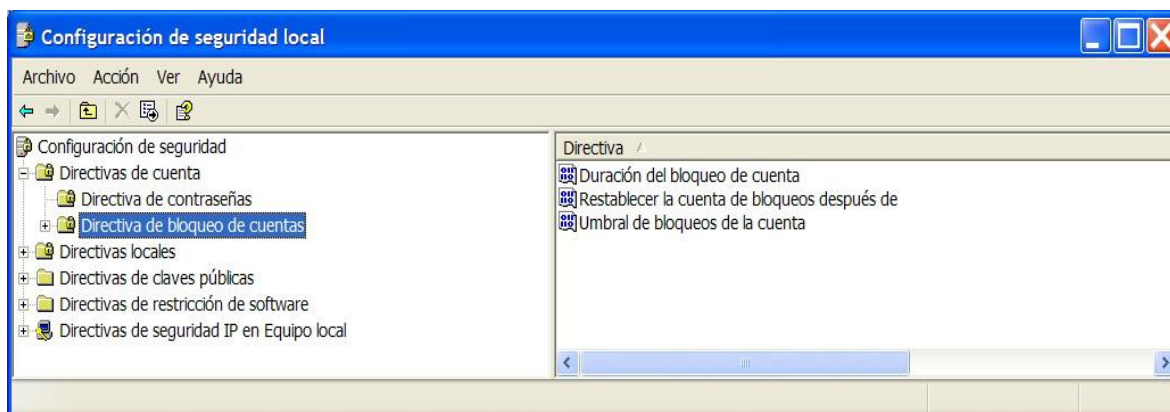
En concreto, desde la consola de Configuración de Seguridad Local (Inicio → Ejecutar: `SecPol.msc` – Configuración de Seguridad – Directivas de Cuenta – Directivas de Contraseñas o bien: panel de control- Herramientas Administrativas- Directiva de seguridad local-...), podemos gestionar varios aspectos sobre las contraseñas

Las configuraciones más útiles que podemos gestionar desde aquí son:

- **Forzar el historial de contraseñas.** Impide que un usuario cambie su contraseña por una contraseña que haya usado anteriormente, el valor numérico indica cuantas contraseñas recordará XP.
- **Las contraseñas deben cumplir los requerimientos de complejidad.** Obliga a que las contraseñas deban cumplir ciertos requerimientos, como son mezclar letras mayúsculas, minúsculas y números, no parecerse al nombre de la cuenta, etc.
- **Longitud mínima de la contraseña.** Indica cuantos caracteres debe tener la contraseña como mínimo, un valor cero en este campo indica que pueden dejarse las contraseñas en blanco.
- **Vigencia máxima de la contraseña.** Las contraseñas de los usuarios caducan y dejan de ser válidas después del número de días indicados en esta configuración, y el sistema obligará al usuario a cambiarlas. (Recordemos que al crear una cuenta de usuario podemos indicar que la contraseña nunca caduca para esa cuenta).
- **Vigencia mínima de la contraseña.** Indica cuanto tiempo debe transcurrir desde que un usuario se cambia la contraseña, hasta que puede volver a cambiarla. Esta configuración de seguridad local se usa para evitar que un usuario cambie continuamente su contraseña a fin de volver a quedarse con su contraseña original caducada.

#### 4.2.4 BLOQUEO DE LAS CUENTAS.

Desde `secpol.msc` también podemos gestionar un comportamiento de las cuentas de usuario relacionado con las contraseñas, y es el bloquear las cuentas si se intenta acceder al sistema con las mismas pero usando contraseñas incorrectas. Esta configuración la encontramos en Inicio → Ejecutar: `SecPol.msc` – Configuración de Seguridad – Directivas de Cuenta – Directivas de Bloqueo de Cuentas:



Aquí podemos configurar:

- **Duración del bloqueo de cuenta.** Durante cuánto tiempo permanecerá una cuenta bloqueada si se supera el umbral de bloqueo. Un valor cero indica que la cuenta se bloqueará hasta que un Administrador la desbloquee.
- **Restablecer la cuenta de bloqueos después de.** Indica cada cuanto tiempo se pone el contador de intentos erróneos a cero.
- **Umbral de bloqueo de la cuenta.** Indica cuantos intentos erróneos se permiten antes de bloquear la cuenta.



## 4.2.5 RECURSOS LOCALES. GESTIÓN DE ACL

El sistema operativo no usa para referirse a las cuentas su nombre o su contraseña. Esos son campos que usamos nosotros, al igual que nos llamamos entre nosotros con nuestro nombre, no con nuestro DNI. El DNI que usa el sistema para referirse a las cuentas de usuario se denomina SID. (*Security Identifier* o Identificador de Seguridad) y es del tipo: S-1-5-21-448539723-413027322-839522115-1003.

El último número, en este caso 1003, se conoce como RID (identificador relativo del usuario) y todo lo que está delante del mismo identifica el dominio al que pertenece ese usuario. En concreto, esos tres grandes números que se observan (448539723-413027322-839522115) se generan automáticamente y al azar cada vez que instalamos un Windows, y aparecerán en todas las cuentas que creemos en dicho sistema.

La parte del SID S-1-5-21 nos da información sobre el objeto con el que estamos trabajando. Así por ejemplo si hablamos de algunos grupos o usuarios especiales tenemos:

S-1-1-0	SID del grupo Todos (Everyone)
S-1-2-0	SID del grupo Usuarios locales
S-1-3-1	SID de Creator – Owner
S-1-5	Este inicio de SID nos indica que estamos trabajando con un usuario o grupo normal. Dentro de este grupo algunos SID conocidos son:
S-1-5-32-544	SID del grupo Administradores
S-1-5-32-545	SID del grupo Usuarios

Cada recurso del sistema cuenta con una lista donde aparecen los usuarios que pueden usar dicho recurso y de qué forma pueden usarlo. Dicha lista se conoce como ACL (*Access Control List*, o Lista de Control de Acceso) y contiene una serie de SIDs y los permisos que cada uno de esos SIDs tiene sobre el recurso.

Cuando un usuario intenta acceder a un recurso pide autorización al mismo, que comprobará entonces si en su ACL está el SID del usuario. Si no, comprobará si está el SID de algún grupo al que pertenezca el usuario. Si aparece en la ACL algún SID del usuario, el recurso comprueba si la acción que quiere realizar (leer, borrar, escribir, etc.) está permitida, o no.

Si no aparece en la ACL ningún SID del usuario, el recurso niega el acceso.

En ocasiones, puede ocurrir que un usuario tenga permisos contradictorios... imaginemos que el usuario PACO, perteneciente al grupo PROFESORES, quiere acceder a la carpeta *margarita*.

En la ACL de *margarita* aparece que el usuario PACO puede escribir en la carpeta, pero el grupo PROFESORES no tiene derecho a escribir.

Bien, en este caso se aplica la siguiente regla:

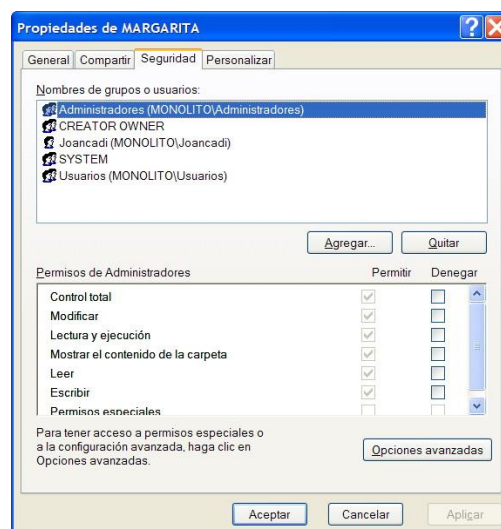
1ª Lo que más pesa es la denegación implícita de permisos. Si un permiso esta denegado, no se sigue mirando, se deniega inmediatamente.

2ª Basta con que un permiso esté concedido en cualquier SID para que se considere concedido. (A excepción de la regla 1ª, es decir, que no esté denegado implícitamente en ningún sitio)

Esto se entiende mejor gestionando el ACL de algún recurso.

Por ejemplo, creemos en el raíz de nuestro volumen una carpeta con nombre *margarita*. Una vez creada, accedemos a sus propiedades y en ellas a la pestaña *Seguridad*:

Podemos ver como en la parte superior tenemos las SID a las que concedemos permisos (usuarios y grupos) y en la parte inferior tenemos los permisos concretos que le concedemos a dicha SID. Si veis las dos columnas por cada



permiso, podemos tanto Permitir como Denegar un permiso. La denegación de un permiso es la que más pesa, y se aplica inmediatamente. De hecho, en Windows se *aconseja no denegar permisos, a menos que sea absolutamente necesario*.

Puede ocurrir que en las propiedades de vuestro recurso no aparezca la pestaña Seguridad. Esto ocurre porque aún tendréis activado el uso compartido simple de archivos de Windows, que es una forma de olvidarnos de las ACL y trabajar de una forma muy simple, indicada para usuarios que no desean preocuparse por estos temas. Para desactivar este uso compartido simple, accedemos a Mi PC, y allí en el menú Herramientas – Opciones de carpeta – Ver accedemos al final de la lista y allí encontramos dicha opción que hay que desactivar.

Más o menos todo lo que se ve en la ACL deberíais entenderlo sin problemas. Con los botones agregar y quitar podemos añadir o quitar SID de la ACL.

En la parte inferior podemos pulsar en las casillas de Permitir y Denegar para dar y quitar permisos.

Pero... ¿por qué aparece la columna de Permitir en gris y no nos deja cambiarla? Bien, ha llegado el momento de hablar de la herencia.

Imaginemos que creamos una carpeta, por ejemplo CONTABLES, y la preparamos minuciosamente para que pueden leer y escribir en ella los usuarios que sean miembros del grupo CONTABLES, para que solo puedan leer los del grupo JEFES pero no escribir, y que los demás usuarios no puedan ni leer en ella ni escribir. Bien, si ahora dentro de la carpeta CONTABLES creamos una nueva carpeta INFORMES, ¿no sería lógico que esta carpeta INFORMES “heredara” la ACL de su carpeta madre CONTABLES para que no tuviera que configurarla nuevamente?

Pues precisamente eso es lo que hace Windows, cualquier recurso que creamos, heredará automáticamente la ACL de su recurso padre si es que existe. En nuestro caso, la carpeta *margarita* ha heredado la ACL de la raíz de nuestro volumen. De modo que no podremos quitar usuarios, quitar permisos, etc.

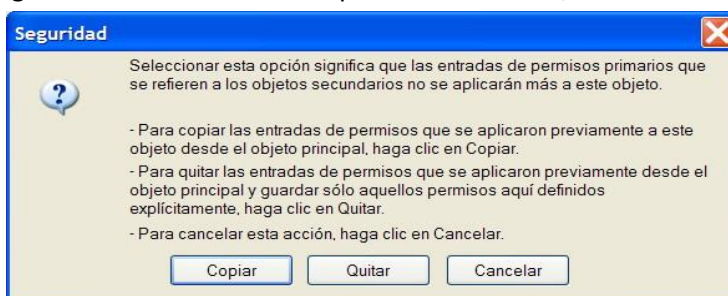
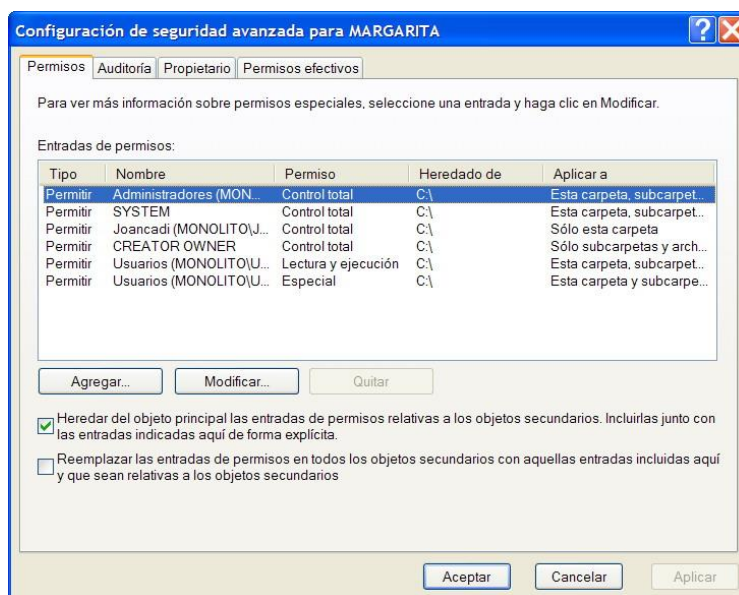
Para realizar cambios en la ACL de nuestra carpeta *margarita*, debemos indicarle que “rompa” la herencia, es decir, que deseamos retocar manualmente su ACL.

Para ello, accedemos al botón de **Opciones Avanzadas** que está en la pestaña Seguridad. Podemos ver 4 pestañas, de momento nos quedamos en la primera, Permisos.

Podemos ver en la parte inferior de esta ventana que está marcada la opción de “Heredar del objeto principal las entradas de permiso relativas a los objetos secundarios. Incluirlas junto con las entradas indicadas aquí de forma explícita”. Si desmarcamos dicha opción quitaremos la relación de herencia de nuestro recurso, y podremos gestionar su ACL. Una vez quitada la herencia, el sistema nos da a elegir entre dos opciones:

Si escogemos la opción Copiar, la herencia se interrumpirá, y podremos retocar la ACL como nos plazca.

Si escogemos la opción Quitar, la ACL se borrará totalmente, se interrumpirá la herencia y la podremos crear desde cero.

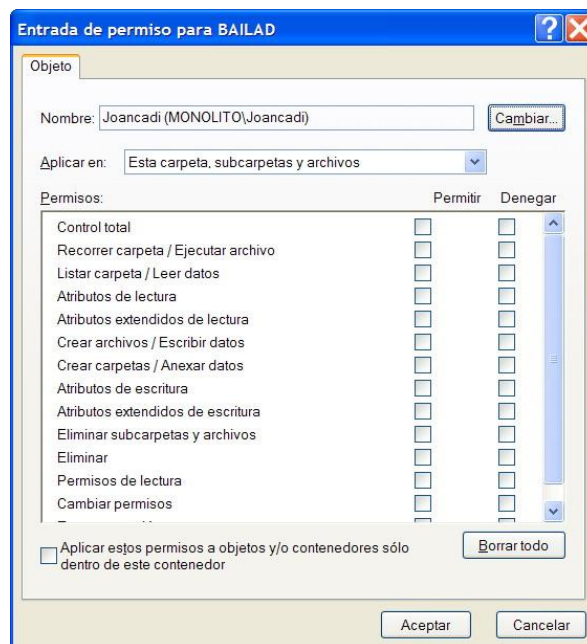


Si elegimos Quitar y empezar desde cero, hay que tener en cuenta que en las ACL no solo deben aparecer nuestras SID normales, sino que grupos como Creator Owner o System son necesarios para que el sistema pueda trabajar sin problemas con dichas carpetas. Si quitamos estos SID tendremos problemas en el futuro (copias de seguridad, auditorías, etc).

Vemos como debajo de la opción de Heredar del objeto principal, tenemos otra opción que nos permite activar que los objetos hijos del nuestro hereden las modificaciones que hagamos en nuestra ACL. Esto es importante si queremos que los cambios que hagamos en la ACL se hereden, ya que hemos roto la herencia y a veces tendremos que forzar dichos cambios.

Los distintos permisos que se pueden aplicar para cada SID en la ACL no son únicamente los que vemos en las propiedades de la carpeta. Si entramos en Opciones avanzadas y allí en Permisos – agregar veremos cómo podemos indicar otro tipo de permisos.

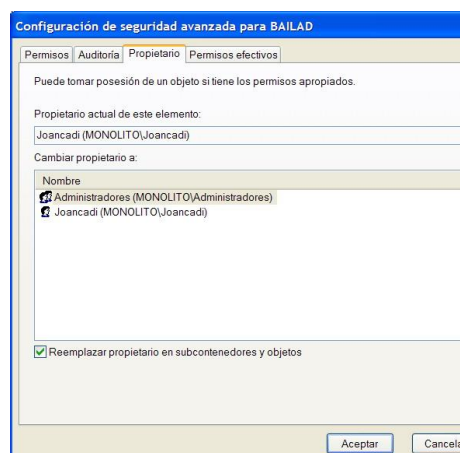
- El permiso **Recorrer carpeta** permite o impide que el usuario pase de una carpeta a otra para llegar a otros archivos o carpetas, incluso aunque el usuario no tenga permisos para las carpetas recorridas (sólo se aplica a carpetas).
- El permiso **Atributos de lectura** permite o impide que el usuario vea los atributos de un archivo o de una carpeta, como sólo lectura y oculto. Los atributos están definidos por el sistema de archivos NTFS..
- El permiso **Atributos de escritura** permite o impide que el usuario cambie los atributos de un archivo o de una carpeta, como sólo lectura y oculto. Los atributos están definidos por el sistema de archivos NTFS.
- El permiso **Leer permisos** permite o impide que el usuario lea permisos del archivo o de la carpeta, como Control total, Leer y Escribir.
- El permiso **Tomar posesión** permite o impide que el usuario tome posesión del archivo o de la carpeta. El propietario de un archivo o de una carpeta puede cambiar los permisos correspondientes, cualesquiera que sean los permisos existentes que protegen el archivo o la carpeta.



Un permiso muy especial es el de **Control Total**. Si este permiso se lo otorgamos a un usuario en una carpeta, este usuario podrá eliminar cualquier cosa que haya en esa carpeta, incluso si le denegamos el permiso de eliminación en esos recursos. Hay que tener mucho cuidado al conceder este permiso.

Lo último que vamos a ver sobre las ACL es el concepto de Creator Owner. El propietario de un recurso es aquel usuario que creó el recurso, y este propietario siempre tiene el poder último sobre el recurso. No importa que dicho usuario desaparezca de su ACL, o incluso que se le deniegue el acceso al recurso; su propietario siempre podrá modificar su ACL.

Uno de los principales poderes del Administrador del sistema (y de hecho, de cualquier miembro del grupo Administradores) es que puede tomar posesión de cualquier recurso. Es decir, que el Administrador puede indicarle a cualquier recurso que él es el creador o propietario del recurso, por lo que pasa a tener todos los poderes sobre el mismo.



Para conseguir esto, tenemos que entrar en las Opciones avanzadas de la pestaña Seguridad del recurso, y allí irnos a la pestaña Propietario.

Desde aquí podemos cambiar el propietario actual del objeto, e indicar que el propietario actual es el grupo Administradores (recomendado antes que asignar el propietario al usuario individual). Basta con que seleccionemos el grupo Administradores y marquemos abajo “Reemplazar propietario en subcontenedores y objetos” y demos aplicar – aceptar. Seremos propietarios de la carpeta y con qué actualicemos la ventana de permisos y ya podremos modificar la ACL del recurso como queramos.

Ojo, esto no nos permite acceder a la carpeta directamente, nos permite modificar su ACL, donde tendremos que introducir el SID del grupo Administradores para, así si, poder acceder a la carpeta con los permisos que indiquemos.

#### Ejercicios:

Crea un usuario nuevo en el sistema, abre sesión como dicho usuario (o usar `runas`), crea una carpeta cualquiera y configura su ACL para que únicamente el usuario nuevo tenga algún permiso sobre dicha carpeta.

Abre sesión ahora como un miembro del grupo de Administradores y consigue acceder a la carpeta para, por ejemplo, crear un archivo dentro.

## 4.3. LINUX

Independientemente de que tengas muchos usuarios o no en el sistema, es importante comprender los aspectos de la gestión de usuarios bajo Linux. Incluso si se es el único usuario, se debe tener, presumiblemente, una cuenta distinta de `root` para hacer la mayor parte del trabajo.

Cada persona que utilice el sistema debe tener su propia cuenta. Raramente es una buena idea el que varias personas compartan la misma cuenta. No sólo es un problema de seguridad, sino que las cuentas se utilizan para identificar unívocamente a los usuarios al sistema. Es necesario ser capaz de saber quién está haciendo qué.

Recordamos que con la distribución `SuSe` podemos gestionar usuarios desde la línea de comandos o utilizando la herramienta `Yast`.

### 4.3.1 CONCEPTOS DE GESTIÓN DE USUARIOS.

El sistema mantiene una cierta cantidad de información acerca de cada usuario. Dicha información se resume a continuación.

- **nombre de usuario** El nombre de usuario es el identificador único dado a cada usuario del sistema. Ejemplos de nombres de usuario son `ana`, `karl` y `mdw`. Se pueden utilizar letras y dígitos junto a los caracteres `_` (subrayado) y `.` (punto). Los nombres de usuario se limitan normalmente a 8 caracteres de longitud.
- **user ID** El *user ID*, o UID, es un número único dado a cada usuario del sistema. El sistema normalmente mantiene la pista de la información por UID, no por nombre de usuario.
- **group ID** El *group ID*, o GID, es la identificación del grupo del usuario por defecto. En un apartado anterior discutimos los permisos de grupo; cada usuario pertenece a uno o más grupos definidos por el administrador del sistema.
- **clave** El sistema también almacena la clave encriptada del usuario. El comando `passwd` se utiliza para poner y cambiar las claves de los usuarios.
- **Información personal** El nombre real y otros datos que puedan ser de interés pueden ser almacenados junto con el nombre de usuario. Por ejemplo, el usuario `schmoj` puede tener el nombre *Jos Schmo* en la vida real.

- **directorio de trabajo** El directorio de trabajo es en el que se sitúa al usuario al conectar y sobre el que, en principio, tendrá todos los permisos. Cada usuario debe tener su propio directorio de trabajo, normalmente hijo de `/home`. Suele denominarse directorio *home*.
- **intérprete de comandos** El intérprete de comandos (*Shell*) del usuario es el intérprete de comandos en modo texto que se inicia cuando conecta el usuario. Ejemplos pueden ser `/bin/bash` y `/bin/tcsh`.

El fichero `/etc/passwd` contiene la información anterior acerca de los usuarios, excepto la clave que, junto con información relativa a los permisos de conexión, se encuentra en `/etc/shadow`. Cada línea del fichero contiene información acerca de un único usuario; el formato de cada línea es

```
nombre:clave:UID:GID:nombre_completo:dir_trabajo:shell
```

Un ejemplo puede ser:

```
/etc/passwd
```

```
kiwi:x:102:100:Laura Poole,,,:/home/kiwi:/bin/bash
```

```
/etc/shadow
```

```
kiwi:$v8Q981g71oKK:16538:0:99999:7:::
```

Como puede verse, el primer campo, *kiwi*, es el nombre de usuario.

El segundo nos indica que el usuario *kiwi* está correctamente registrado en el sistema. Si aún no se le ha asignado contraseña aparecerá !

El tercer campo *102*, es el UID. Este debe ser único para cada usuario.

El cuarto campo, *100*, es el GID. Este usuario pertenece al grupo numerado 100. La información de grupos, como la información de usuarios, se almacena en el fichero `/etc/group`.

El quinto campo es el nombre completo del usuario: *Laura Poole*. Los dos últimos campos son el directorio de trabajo del usuario (`/home/kiwi`) y el intérprete de comandos (`/bin/bash`), respectivamente. No es necesario que el directorio inicial de un usuario tenga el mismo nombre que el del nombre de usuario.

En el fichero `/etc/shadow` el segundo campo, *Xv8Q981g71oKK*, es la clave encriptada, normalmente en sistema MD5. Las claves se encriptan utilizándose a sí mismas como clave secreta. En otras palabras, sólo si se conoce la clave, ésta puede ser desencriptada. Esta forma de encriptación es bastante segura.

## **AÑADIR USUARIOS.**

Aunque no es difícil el añadir usuarios a mano (editando el fichero `/etc/passwd`), cuando se está ejecutando un sistema con muchos usuarios, es fácil el olvidarse de algo. La manera más simple de añadir usuarios es utilizar un programa interactivo (en SuSe *Yast*) que vaya preguntando por la información necesaria y actualice todos los ficheros del sistema automáticamente.

En su defecto, usaremos el comando `useradd` o `adduser` (este último es más complejo de manejar y no suele usarse nada más que para añadir usuarios a grupos) dependiendo del software que esté instalado.

```
useradd [-g UID] [-g grupo] [-G lista-grupo] [-d dir-trabajo[--m]] [-s shell] usuario
useradd -u 300 -g users -s /bin/bash -d/home/ana -m ana
```

donde :

- u ID del usuario
- g grupo principal al que pertenece
- G grupo o grupos secundarios
- s shell
- d directorio de trabajo

- m copia todos los archivos el directorio `/etc/skel` al directorio indicado por `-d`. En el directorio `/etc/skel` deberán estar todos los archivos que el administrador considere que tengan que tener todos los usuarios del sistema

Cuando no se indican modificadores, el sistema toma los valores por defecto que se hayan indicado en los ficheros `/etc/login.defs` y `/etc/default/useradd`.

El primero de ellos, `/etc/login.defs`, contiene parámetros que establecen la localización por defecto del buzón de correo, el plazo para la expiración y el mínimo de caracteres del *password* o el rango de UID's y GID's disponibles para su uso. Además, determina si serán creados los directorios *home* de los usuarios durante la creación de su cuenta.

El archivo `/etc/default/useradd` contiene información acerca del grupo primario, la localización de los directorios *home*, el número de días por defecto para deshabilitar cuentas con claves expiradas, la Shell utilizada y el directorio *skel* utilizado que necesita el comando `useradd` cuando se usa sin modificadores para establecer los valores por defecto.

La orden `useradd` no asigna una contraseña a la nueva cuenta por lo que el nuevo usuario no podrá conectarse al sistema (está bloqueado). El administrador del sistema debe definir una contraseña para cada usuario para desbloquearlo. Para conseguir esto deberemos utilizar la orden

```
passwd [usuario]
```

para asignar una contraseña a la cuenta de un usuario por primera vez o para modificar una ya existente. El comando entabla un diálogo con el operador solicitándole que escriba la contraseña dos veces para asegurarse de que ha sido introducida correctamente.

Un usuario puede cambiar su propia contraseña si le apetece. Para ello deberá usar el comando `passwd` sin especificar cuenta alguna (sólo el administrador del sistema tiene permiso para especificar un nombre de cuenta en `passwd`).

### **BORRAR USUARIOS.**

Para eliminar un usuario del sistema se usa

```
userdel [-r] usuario
```

donde `-r` elimina el directorio *home* del usuario

### **MODIFICAR DATOS DEL USUARIO**

El comando `usermod` tiene los mismos parámetros que `useradd`, pero al ser utilizado para modificar los parámetros de los usuarios ya existentes incorpora una serie de opciones extra:

```
usermod [-L][[-U] [-l nuevo-login] usuario
```

- L bloquea el acceso al sistema del usuario indicado (aparece una ! precediendo la contraseña en el fichero `/etc/shadow`)
- U desbloquea al usuario
- l cambia el login del usuario

## **4.3.2 GESTIÓN DE GRUPOS**

La asignación de un usuario a uno o varios grupos nos permitirá mejorar la autorización de acceso a los recursos del sistema. Los grupos se almacenan en el fichero `/etc/group`:

```
<nombregrupo>::<GID>:<listadeusuarios>
```

### **CREAR GRUPOS**

```
groupadd [-gGID[-o]][-r][-f] nombre_grupo
```

- g GID Asigna el identificador de grupo especificado al nuevo grupo. Este número debe ser único (a menos que se especifique `-o`) y positivo. Por defecto se asigna un valor mayor que 500 y que cualquier GID ya existente. Los valores comprendidos entre 0-999 se reservan normalmente para cuentas del sistema. Podemos especificar el rango de los GID's en el archivo `/etc/login.defs`



- r Añade un grupo del sistema. Sólo en algunas distribuciones (Red Hat y SuSe, por ejemplo). El GID estará comprendido entre los valores de SYSTEM\_GID\_MIN y SYSTEM\_GID\_MAX que se inicializan en el archivo `/etc/login.defs` y por defecto valen 0 y 500 respectivamente. En estas mismas distribuciones los valores para GID están comprendidos entre GID\_MIN y GID\_MAX, definidas en el mismo archivo y cuyos valores predeterminados son respectivamente 1000 y 60000.
- f (force) (No existe en SuSe) Provoca que el comando termine con un error si el grupo a dar de alta existe ya (esta acción está por defecto activada en SuSe). El grupo existente no será alterado. Modifica también el funcionamiento del parámetro `-g`: si especificamos un GID ya existente en vez de dar error actúa como si no hubiésemos especificado `-g`, es decir, fuerza la creación del grupo con un GID asignado por el sistema.

### **MODIFICAR GRUPOS**

```
groupmod [-gGID[-o]] [-n nuevonombre] nombre_grupo
```

### **BORRAR GRUPOS**

```
groupdel grupo
```

No se puede eliminar el grupo primario de un usuario cuya cuenta existe aún (primero habremos de eliminar la cuenta del sistema).

### **¿A QUÉ GRUPOS PERTENECEMOS?**

```
groups [usuario]
```

Muestra primero el grupo principal del usuario y después los grupos secundarios a los que pertenece.

### **AÑADIR O ELIMINAR UN USUARIO**

Para añadir un nuevo usuario a un grupo podemos utilizar el comando `adduser` seguido del nombre del usuario y del nombre del grupo al que queremos incorporarlo:

```
adduser nombre-de-usuario grupo
```

Para eliminar un usuario de un grupo usaremos:

```
deluser usuario grupo
```

## **4.3.3 PERMISOS.**

Después de crear los usuarios y/o grupos, tenemos que controlar el acceso a los recursos a través de los permisos de los mismos. Para ello, hemos de controlar la máscara de permisos, el usuario propietario y el grupo propietario del fichero o directorio. Dicha información está contenida en el i-nodo del archivo y podemos verla mediante el comando `ls`:

```
# ls -l Mifichero
-rwxr---x----- 1 mode grmode 890 Feb 5 20:30 Mifichero
```


# Falta todo el tema de permisos

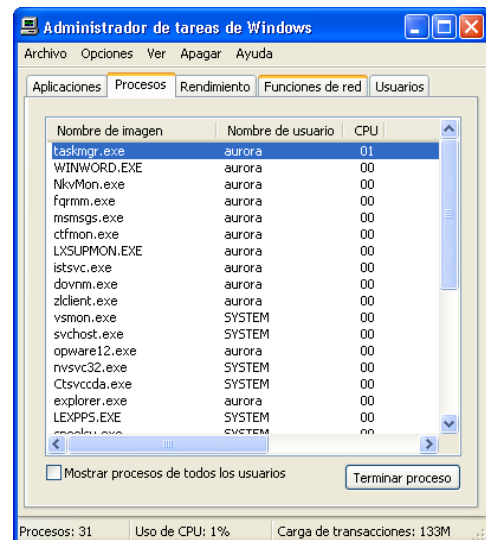
## 5. GESTIÓN DE PROCESOS Y SERVICIOS

### 5.1. WINDOWS

#### 5.1.1 EL ADMINISTRADOR DE TAREAS

El **Administrador de Tareas** es una herramienta de diagnóstico rápido para problemas de rendimiento. Da información sobre aplicaciones que se están ejecutando, procesos en ejecución, uso de memoria, sobre la red y usuarios. Para acceder al Administrador de Tareas:

- **Ctrl+Alt+Supr** si el ordenador está unido a un dominio o la ventana de bienvenida está inhabilitada aparecerá el cuadro de diálogo **Windows Security**, donde hay un botón para acceder al **Administrador de Tareas**. De otra forma el Administrador de tareas se activa al pulsar esa combinación de teclas.
- Clic derecho en la **barra de tareas** y seleccionar **Administrador de Tareas**.
- También se puede llegar tecleando `taskmgr` en una ventana de comando ( Símbolo del sistema Inicio/Accesorios/Símbolo del sistema) o en Inicio/Ejecutar.
- Ctrl+Shift+Esc

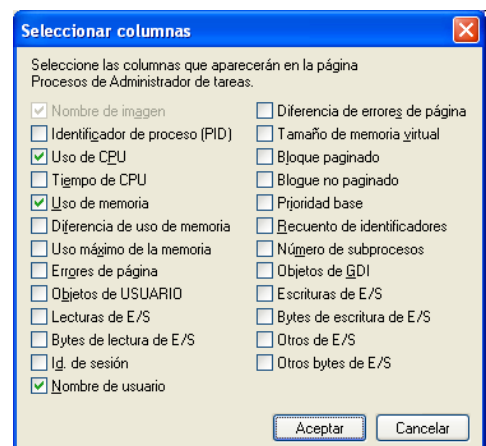


##### FICHA APLICACIONES

Al seleccionar esta ficha se muestran las aplicaciones que están actualmente ejecutándose con información básica sobre si está ejecutándose o no responde en este momento. Desde aquí se pueden iniciar aplicaciones haciendo clic en el botón **Nueva Tarea** o seleccionando **Archivo/Nueva tarea(Ejecutar.....)** y tecleando el nombre del fichero ejecutable de la aplicación que se quiere iniciar. También se puede usar el botón **Finalizar Tarea** para terminar una aplicación que no responde (está colgada), o el botón **Pasar a** para cambiar a otra aplicación y traer el programa a primer plano.

##### FICHA PROCESOS

Observar que en esta ventana se puede ver información detallada sobre procesos en ejecución, la cuenta de usuario que ejecuta el proceso, el uso de la CPU, y la memoria usada por el proceso. Se puede terminar un proceso botón **Terminar Proceso**.



Se pueden añadir contadores a esta ventana añadiendo columnas. Para añadirlas, seleccionar **Ver/Seleccionar columnas**. Existe la posibilidad de seleccionar un gran número de áreas que monitorizar, incluyendo lectura y escritura de I/O, memoria, información de fallo de página,...

Campo	Descripción
Nombre de Imagen	El nombre del proceso.
Identificador de proceso (PID)	Identificador numérico asignado al proceso mientras se está ejecutando.
Uso de CPU	Porcentaje de tiempo de procesador que ha usado el proceso desde la última actualización.



Tiempo de CPU	Tiempo total de procesador (en segundos) usado por el proceso desde que empezó a ejecutarse.
Prioridad base	Determina el orden de ejecución de los procesos. Se puede cambiar la prioridad de los procesos seleccionando el proceso y clic en el proceso <i>menú contextual/Establecer prioridad</i> .
Bytes de lectura de E/S	Número de bytes leídos en operaciones de entrada o salida generadas por un proceso
Lecturas de E/S	Número de operaciones de entrada o salida de lectura generadas por un proceso, incluidas las operaciones con archivos, la red y dispositivos de E/S
Bytes de escritura de E/S	Número de bytes escritos en operaciones de entrada o salida generadas por un proceso.
Escrituras de E/S	Número de operaciones de entrada o salida de escritura generadas por un proceso.
Bloque no paginado	No hay paginación de memoria a disco.
Bloque paginado	Memoria virtual asignada por el sistema atribuida a un proceso que se puede pagar a disco.
Error de página	Interrupción que se produce cuando un programa intenta leer o escribir en una ubicación de la memoria virtual marcada como no presente.
Uso de memoria	Número de páginas de un proceso que residen actualmente en la memoria.
Número de subprocesos	Número de subprocesos que se ejecutan en un proceso.

### FICHA RENDIMIENTO

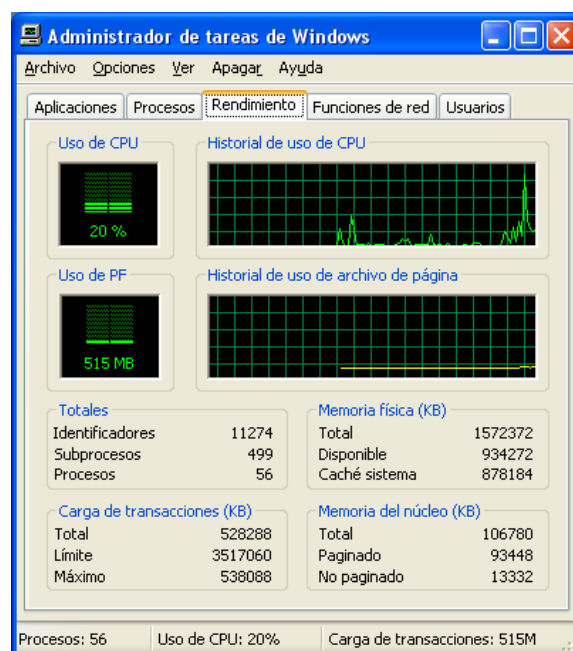
Presenta una gráfica del rendimiento que ofrece un análisis de la CPU y uso de “page file” para resolver problemas sobre procesos o memoria. La ventana Historial de uso de archivo de página, incluyen también secciones de texto con información actualizada dinámicamente sobre el uso de la memoria (memoria en uso por el sistema operativo, memoria física instalada en la máquina y memoria asignada a programas y al sistema).

### FICHA FUNCIONES DE RED

Se puede usar para ver problemas del adaptador, condiciones de red y rendimiento de la red. La visualización por defecto muestra una representación gráfica del rendimiento de la red. Se pueden elegir las columnas para tener estadísticas o un análisis más en profundidad. Por defecto la gráfica muestra lo siguiente:

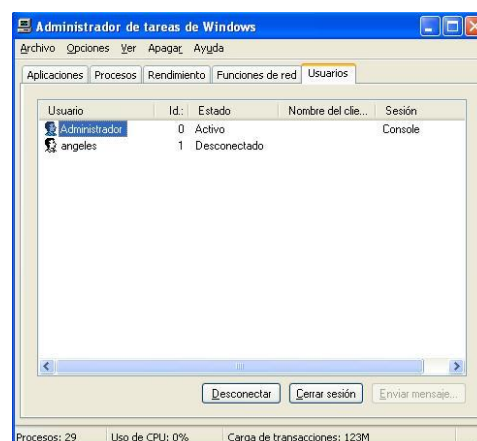
- Bytes recibidos.
- Bytes Totales.
- El porcentaje de utilización en la red

Para acceder a las columnas adicionales, elegir **Ver/Seleccionar columnas**.



### FICHA USUARIOS

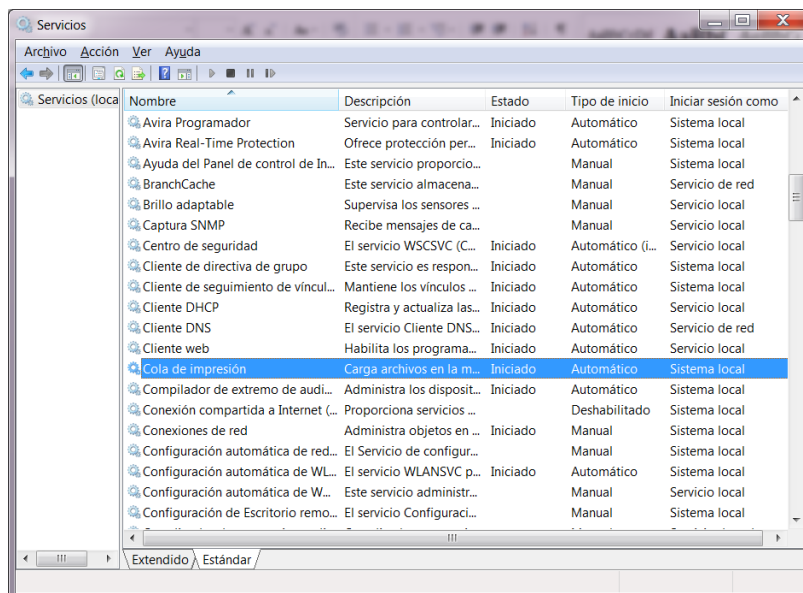
La ficha **Usuarios** muestra los usuarios que pueden tener acceso al equipo y el estado (activo o desconectado) y los nombres de sesión. **Nombre de cliente** especifica el nombre del equipo cliente que utiliza la sesión, si es aplicable. **Sesión** proporciona el nombre que puede utilizar para ejecutar tareas como enviar mensajes a otros usuarios. La ficha **Usuarios** sólo se muestra si el equipo en el que trabaja tiene habilitada la opción Cambio rápido de usuario, es un equipo independiente o es miembro de un grupo de trabajo. No está disponible en equipos que son miembros de un dominio de red.



## 5.1.2 LOS SERVICIOS DEL SISTEMA

Ya hemos visto cómo se gestionan los procesos, pero es importante conocer que dentro del sistema constantemente se están ejecutando otros procesos en segundo plano que nos sirven. Por ejemplo, si queremos imprimir un archivo, no nos preocupamos nada más que de hacer clic en el icono que indica tal acción y el archivo se imprime. Pero para que un archivo se pueda imprimir, el sistema debe ejecutar unos servicios de impresión.

Estos servicios, por ejemplo los de impresión, tienen una función clara: hacer que la impresora sea reconocida por el sistema; que cuando un usuario lance el trabajo de impresión, el sistema operativo sepa adónde tiene que ir a imprimir; utilizar los conectores externos del equipo para comunicarse con la impresora, etc. Sirva también como ejemplo que cuando hacemos clic en el icono de Internet Explorer, sencillamente navegamos y ya está. En este caso, intervienen otros servicios que hacen que la conexión del programa Internet Explorer se realice a través del módem o tarjeta de red, para solicitar las páginas deseadas.



Los **servicios** son programas o aplicaciones cargadas por el propio sistema operativo y que se ejecutan en segundo plano. Por defecto, con la instalación, se instalan y ejecutan algunos de estos servicios. Dependiendo de nuestras necesidades, podemos necesitarlos todos o no. Como sabemos, mientras más aplicaciones tengamos ejecutándose consumimos más recursos, por lo tanto, es interesante deshabilitar los que no utilizamos para acelerar el trabajo de nuestro ordenador.

En la **consola de servicios** encontraremos una lista de servicios con una breve descripción de cada uno de ellos y su estado. Podemos acceder a la consola de servicios desde:

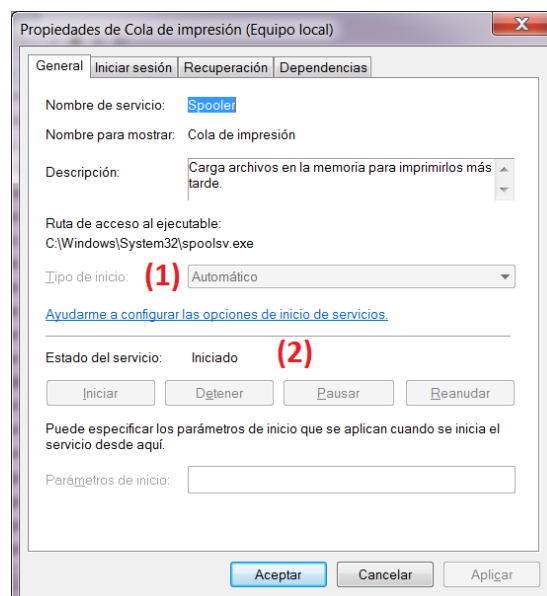
- Inicio/Panel de control/Rendimiento y mantenimiento/Herramientas administrativas/Servicios.
- Inicio/Panel de control /Herramientas administrativas/Servicios
- *Inicio/Ejecutar* y escribimos `services.msc`.

Normalmente, cuando instalamos algún tipo de software, como por ejemplo un antivirus, VMware, etc., se añaden nuevos servicios al sistema que harán que estos programas puedan ejecutarse y realizar sus funciones.

Si hacemos clic con el botón derecho del ratón, podremos realizar varias acciones: iniciarlo, detenerlo, pausarlo, reanudarlo o reiniciarlo; también acceder a sus propiedades, que es lo mismo que hacer doble clic sobre el servicio.

Si detenemos un servicio, éste dejará de funcionar en esta sesión de trabajo, pero dependiendo de cómo esté configurado el inicio del mismo, lo normal es que en la siguiente sesión se vuelva a ejecutar.

Iniciar un servicio es un proceso que se hará si el servicio, una vez iniciado el sistema, no se está ejecutando y queremos que se ejecute.



Reanudar y pausar un servicio es un procedimiento que se utiliza puntualmente cuando queremos que un servicio deje de funcionar para hacer alguna prueba, como por ejemplo el cortafuegos de Windows (Firewall).

El resto de pestañas de la figura anterior ahora no nos interesan, a excepción de la de Dependencias. En ella podremos ver si este servicio depende de otros o si de él dependen otros. Esto lo tendremos en cuenta si detenemos el servicio, ya que los que dependan de él también dejarán de funcionar.

En el caso de la Cola de impresión, podremos observar que, si la detenemos, también dejará de funcionar el Servicio de FAX.

- **Tipo de inicio.** Aquí indicaremos si el servicio queremos que se inicie de forma automática al iniciar el sistema, si lo queremos activar manualmente o si simplemente queremos deshabilitarlo.
- **Estado del servicio.** Vamos a detener el servicio de Cola de impresión. Para ello, haremos doble clic sobre él y, en la ventana que aparece en Estado del servicio, pulsaremos el botón Detener. Para ver qué efectos tiene el haber detenido este servicio, nos iremos a Panel de control > Impresoras. En una zona vacía de la pantalla, haremos clic con el botón derecho del ratón, seleccionaremos Agregar impresora y comprobaremos que no es posible porque el servicio de impresión está parado.

#### Ejercicios:

Ejecutar el Administrador de Tareas para monitorizar programas, procesos y el rendimiento del sistema.

2. Iniciar sesión como un usuario miembro del grupo Administradores
3. Presionar Ctrl+Shift+Esc para lanzar el Administrador de Tareas. ¿Qué programas están actualmente ejecutándose en el sistema?
4. Hacer clic en Nueva Tarea. Aparece un cuadro de diálogo.
5. Teclear Wordpad y clic en Aceptar. WordPad debe empezar y se listará como una aplicación ejecutándose.
6. Seleccionar la pestaña Procesos. ¿Cuántos procesos hay ejecutándose?
7. En el menú Vista, clic en Seleccionar Columnas. Aparece el cuadro de diálogo Seleccionar Columnas.
8. Seleccionar Uso máximo de memoria, y Identificador del proceso a continuación Aceptar. Dos nuevas columnas deben aparecer en la ficha Procesos.
9. Situar en la ficha Rendimiento. ¿Qué porcentaje de la capacidad de la CPU está siendo usado?
10. Desde la ficha Aplicaciones, seleccionar la tarea WordPad menú contextual/Ir al proceso y a continuación se situará en el proceso de la tarea seleccionada.
11. En la ubicación actual o desde la ficha Aplicaciones hacer clic Finalizar Tarea. WordPad se cierra y se borra de la lista de aplicaciones ejecutándose.
12. Cerrar el Administrador de Tareas.

## 5.2. LINUX

Los procesos que se encuentren en ejecución en Linux en un determinado momento serán, en general:

- **Procesos de sistema.** O bien procesos propios del núcleo (kernel del propio sistema operativo). También encontraremos procesos (denominados daemons o demonios) asociados al control de diferentes servicios locales o de red. La mayoría de estos procesos aparecerán asociados al usuario root, aunque no sea este usuario el que se ha validado al sistema.
- **Procesos del usuario Administrador.** En caso de estar identificados como root, los procesos o aplicaciones que se lancen también aparecerán como procesos asociados al usuario root.
- **Procesos de usuarios del sistema.** Asociados a la ejecución de sus aplicaciones, sean tareas interactivas en modo texto o en modo gráfico.

En general, un proceso en Linux se puede encontrar en uno de estos estados:

- **Preparado** ®. Proceso que está listo para ejecutarse. Simplemente está esperando a que el sistema operativo le asigne un tiempo de UCP.
- **Ejecutando** (0). Cuando el proceso pasa de estado Parado, a utilizar recursos de CPU, Memoria, y del resto de hardware.
- **Bloqueado** (5). Un proceso se encuentra suspendido o bloqueado si la UPC no le asigna tiempo de ejecución, debido especialmente a un bloqueo o a que al proceso le falta algo para poderse ejecutar.
- **Parado** (T). Un proceso parado tampoco entra en el reparto de UCP, pero no por que se encuentre bloqueado, sino porque el tiempo de UCP lo está utilizando otro proceso.
- **Zombie** (Z) o Dormido. Todo proceso, al finalizar, se debería eliminar del BCP. Si por lo que sea no se elimina de esta tabla, simplemente se queda en este estado.

### 5.2.1 CONTROL DE TAREAS.

#### TAREAS Y PROCESOS.

Control de tareas es una utilidad incluida en muchos shells (incluidas `Bash` y `Tcsh`), que permite el control de multitud de comandos o tareas al momento. Antes de seguir, deberemos hablar un poco sobre los procesos.

Cada vez que se ejecuta un programa, el usuario lanza lo que se conoce como proceso, que es simplemente el nombre que se le da a un programa cuando se está ejecutando. El comando `ps`<sup>3</sup> visualiza la lista de procesos que se están ejecutando actualmente, por ejemplo:

```
/home/ana# ps
  PID   TT     STAT   TIME   COMMAND
   24    3      S      0:03   (bash)
  161    3      R      0:00   ps
/home/ana#
```

La columna `PID` representa el identificador de proceso. La última columna `COMMAND`, es el nombre del proceso que se está ejecutando. Ahora solo estamos viendo los procesos que está ejecutando Ana. Vemos que hay dos procesos, `bash` (Que es el *shell* o intérprete de comandos que usa Ana), y el propio comando `ps`. Como podemos observar, la `bash` se ejecuta concurrentemente con el comando `ps`. La `bash` ejecutó `ps` cuando Ana tecleó el comando. Cuando `ps` termina de ejecutarse (después de mostrar la tabla de procesos), el control retorna al proceso `bash`, que muestra el *prompt*, indicando que está listo para recibir otro comando.

<sup>3</sup> Hay muchos más procesos aparte de estos corriendo en el sistema, para verlos todos, usar el comando "`ps -aux`".

Un proceso que está corriendo se denomina tarea para el *shell*. Los términos proceso y tarea, son intercambiables. Sin embargo, se suele denominar "tarea" a un proceso, cuando es usado en conjunción con control de tareas, que es un rasgo del *shell* que permite cambiar entre distintas tareas.

En muchos casos, los usuarios sólo ejecutan un trabajo cada vez, que es el último comando que ellos teclearon desde el *shell*. Sin embargo, usando el control de tareas, podremos ejecutar diferentes tareas al mismo tiempo, cambiando entre cada uno de ellos conforme lo necesitemos. ¿Cuán beneficioso puede llegar a ser esto? Supongamos que estás con tu procesador de textos, y de repente necesitas parar y realizar otra tarea, con el control de tareas, puedes suspender temporalmente el editor, y volver al *shell* para realizar cualquier otra tarea, y luego regresar al editor como si no lo hubiésemos dejado nunca. Lo siguiente solo es un ejemplo, hay montones de usos prácticos del control de tareas.

### **PRIMER PLANO Y SEGUNDO PLANO.**

Un proceso puede estar en Primer plano o en Segundo plano. Sólo puede haber un proceso en primer plano al mismo tiempo, siendo éste el que interactúa con el usuario, recibe entradas de teclado, y envía las salidas al monitor (salvo, por supuesto, que haya redirigido la entrada o la salida). El proceso en segundo plano, no recibe ninguna señal desde el teclado, por lo general, se ejecutan en silencio sin necesidad de interacción.

Algunos programas necesitan mucho tiempo para terminar, y no hacen nada interesante mientras tanto. Compilar programas es una de estas tareas, así como comprimir un fichero grande. No tiene sentido sentarnos y aburrirnos mientras estos procesos terminan. En estos casos es mejor lanzarlos en segundo plano, para dejar el ordenador en condiciones de ejecutar otro programa.

Los procesos pueden ser suspendidos. Un proceso suspendido es aquel que no se está ejecutando actualmente, sino que está temporalmente parado. Después de suspender una tarea, podemos indicar a la misma que continúe, en primer plano o en segundo, según necesite. Retomar una tarea suspendida no cambia en nada el estado de la misma, la tarea continuará ejecutándose justo donde se dejó.

Suspender un trabajo no es lo mismo que interrumpirlo. Cuando se interrumpe un proceso (generalmente con la pulsación de [ctrl-C]<sup>4</sup>), el proceso muere, y deja de estar en memoria y utilizar recursos del ordenador. Una vez eliminado, el proceso no puede continuar ejecutándose, y deberá ser lanzado otra vez para volver a realizar sus tareas. También se puede dar el caso de que algunos programas capturen la interrupción, de modo que pulsando [ctrl-C] no se pare inmediatamente. Esto se hace para permitir al programa realizar operaciones necesarias de limpieza antes de terminar<sup>5</sup>. De hecho, algunos programas simplemente no se dejan matar por ninguna interrupción.

### **ENVÍO A SEGUNDO PLANO Y ELIMINACIÓN PROCESOS.**

Empecemos con un ejemplo sencillo. El comando *yes* es un comando aparentemente inútil que envía una serie interminable de *y*-es a la salida estándar (realmente es muy útil: si se utiliza una tubería (*pipe*) para unir la salida de *yes* con otro comando que haga preguntas del tipo si/no, la serie de *y*-es confirmará todas las preguntas.)

Prueba con esto.

```
/home/ana# yes
y
y
y
```

---

<sup>4</sup> La tecla de interrupción puede definirse usando el comando *stty*. Por defecto, en la mayoría de sistemas es [ctrl-C]

<sup>5</sup> Pero no se puede garantizar que sea la misma en su sistema. Tiempo necesario para guardar algunos registros, etc.

La serie de `y`-es continuará hasta el infinito, a no ser que las eliminemos, pulsando la tecla de interrupción, generalmente `[ctrl-C]`. También podemos deshacernos de esta serie de `y`-es redigiendo la salida estándar de `yes` hacia `/dev/null`, que como recordarás es una especie de "agujero negro" o papelera para los datos. Todo lo que enviemos allí, desaparecerá.

```
/home/ana# yes > /dev/null
```

Ahora va mucho mejor, el terminal no se ensucia, pero el *prompt* de la *shell* no retorna. Esto es porque `yes` sigue ejecutándose y enviando esos inútiles `y`-es a `/dev/null`. Para recuperarlo, pulsamos la tecla de interrupción.

Supongamos ahora que queremos dejar que el comando `yes` siga ejecutándose, y volver al mismo tiempo a la *shell* para trabajar en otras cosas. Para ello nos enviaremos a `yes` a segundo plano, lo que nos permitirá ejecutarlo, pero sin necesidad de interacción.

Una forma de mandar procesos a segundo plano es añadiendo un carácter `&` al final de cada comando.

```
/home/ana# yes > /dev/null &
[1] 164
/home/ana#
```

Como podrá ver, ha regresado a la *shell*. ¿Pero que es eso de `[1] 164`?, ¿se está ejecutando realmente el comando `yes`?

`[1]` representa el número de tarea o `ID` del proceso `yes`. La *shell* asigna un número a cada tarea que se esté ejecutando. Como `yes` es el único comando que se está ejecutando, se le asigna el número de tarea 1. El número 164 es el número de identificación del proceso o `PID`, que es el número que el sistema le asigna al proceso. Ambos números pueden usarse para referirse a la tarea como veremos después.

Ahora tenemos el proceso `yes` corriendo en segundo plano, y enviando constantemente la señal `y` hacia el dispositivo `/dev/null`. Para chequear el estado del proceso, utilizaremos el comando interno de la *shell* `jobs`:

```
/home/ana# jobs
[1]+  Running                  yes >/dev/null &
/home/ana#
```

También podemos usar el comando `ps`, como mostramos antes, para comprobar el estado de la tarea.

Para eliminar una tarea, utilizaremos el comando `kill`. Este comando toma como argumento un número de tarea o un número de `ID` de un proceso. Ésta era la tarea 1, así que usando el comando

```
/home/ana# kill %1
```

matarás la tarea. Cuando se identifica la tarea con el número de tarea, se debe preceder el número con el carácter de porcentaje (%).

Ahora que ya hemos matado la tarea, podemos usar el comando `jobs` de nuevo para comprobarlo:

```
/home/ana# jobs
[1]+  Terminated              yes >/dev/null
/home/ana#
```

La tarea está, en efecto, muerta, y si usamos el comando `jobs` de nuevo, no mostrará nada. También podremos matar la tarea usando el número de identificador de proceso (`PID`), el cual se muestra conjuntamente con el `ID` de tarea cuando arranca la misma. En nuestro ejemplo el `PID` es 164, así que el comando

```
/home/ana# kill 164
```

es equivalente a

```
/home/ana# kill %1
```

No es necesario usar el % cuando nos referimos a una tarea a través de su PID.

### **PARADA Y RELANZAMIENTO DE TAREAS.**

Hay otra manera de poner una tarea en segundo plano. Podemos lanzarlo como un proceso normal (en primer plano), pararlo, y después relanzarlo en segundo plano. Primero, lanzamos el proceso `yes` en primer plano como lo haríamos normalmente:

```
/home/ana# yes > /dev/null
```

De nuevo, dado que `yes` corre en primer plano, no debe retornar el prompt de la *shell*. Ahora, en vez de interrumpir la tarea con `[ctrl-C]`, suspenderemos la tarea. El suspender una tarea no la mata: solamente la detiene temporalmente hasta que la retomamos. Para hacer esto debemos pulsar la tecla de suspender, que suele ser `[ctrl-Z]`.

```
/home/ana#yes > /dev/null
[ctrl-Z]
[1]+  Stopped          yes >/dev/null
/home/ana#
```

Mientras el proceso está suspendido, simplemente no se está ejecutando. No gasta tiempo de CPU en la tarea. Sin embargo, podemos retomar el proceso de nuevo como si nada hubiera pasado. Continuará ejecutándose donde se dejó. Para relanzar la tarea en primer plano, usamos el comando `fg` (*foreground*).

```
/home/ana# fg
yes >/dev/null
```

La *shell* muestra el nombre del comando de nuevo, de forma que tengamos conocimiento de que tarea es la que ha puesto en primer plano. Paramos la tarea de nuevo, con `[ctrl-Z]`. Esta vez utilizaremos el comando `bg` para poner la tarea en segundo plano. Esto hará que el comando siga ejecutándose igual que si lo hubiese hecho desde el principio con `&` como en la sección anterior.

```
/home/ana# bg
[1]+ yes >/dev/null &
/home/ana#
```

Y tenemos de nuevo el *prompt*. El comando `jobs` debería decirnos que `yes` se está ejecutando, y podemos matar la tarea con `kill` tal y como lo hicimos antes.

¿Cómo podemos parar la tarea de nuevo? Si pulsamos `[ctrl-Z]` no funcionará, ya que el proceso está en segundo plano. La respuesta es poner el proceso en primer plano de nuevo, con el comando `fg`, y entonces pararlo. Como se puede observar podremos usar `fg` tanto con tareas detenidas, como con las que estén en segundo plano.

Hay una gran diferencia entre una tarea que se encuentra en segundo plano, y una que se encuentra detenida. Una tarea detenida es una tarea que no se está ejecutando, es decir, que no usa tiempo de CPU, y que no está haciendo ningún trabajo (la tarea aún ocupa un lugar en memoria, aunque puede ser volcada a disco). Una tarea en segundo plano, se está ejecutando, y usando memoria, a la vez que completando alguna acción mientras nosotros hacemos otro trabajo. Sin embargo, una tarea en segundo plano puede intentar mostrar texto en su terminal, lo que puede resultar molesto si estamos intentando hacer otra cosa. Por ejemplo, si usamos el comando

```
/home/ana# yes &
```

sin redirigir `stdout` a `/dev/null`, una cadena de `y-es` se mostrarán en el monitor, sin modo alguno de interrumpirlo (no podemos hacer uso de `[ctrl-C]` para interrumpir tareas en segundo



plano). Para poder parar esas interminables *y-es*, tendríamos que usar el comando `fg` para pasar la tarea a primer plano, y entonces usar `[ctrl-C]` para matarla.

Otra observación. Normalmente, los comandos `fg` y `bg` actúan sobre el último proceso parado (indicado por un `+` junto al número de tarea cuando usa el comando `jobs`). Si tenemos varios procesos corriendo a la vez, podremos mandar a primer o segundo plano una tarea específica indicando el ID de tarea como argumento de `fg` o `bg`, como en

```
/home/ana# fg %2
```

(para la tarea de primer plano número 2), o

```
/home/ana# bg %3
```

(para la tarea de segundo plano número 3). No se pueden usar los PID con `fg` o `bg`. Además de esto, si usa el número de tarea por sí solo, como

```
/home/ana# %2
```

es equivalente a

```
/home/ana# fg %2
```

Solo recordar que el uso de control de tareas es una utilidad de la *shell*. Los comandos `fg`, `bg` y `jobs` son internos de la *shell*. Si por algún motivo utilizamos una *shell* que no soporta control de tareas, no esperemos disponer de estos comandos.

Y además, hay algunos aspectos del control de tareas que difieren entre `Bash` y `Tcsh`. De hecho, algunas *shells* no proporcionan ningún control de tareas sin embargo, la mayoría de las *shells* disponibles para Linux soportan control de tareas.

## 5.2.2 ADMINISTRACIÓN DE PROCESOS.

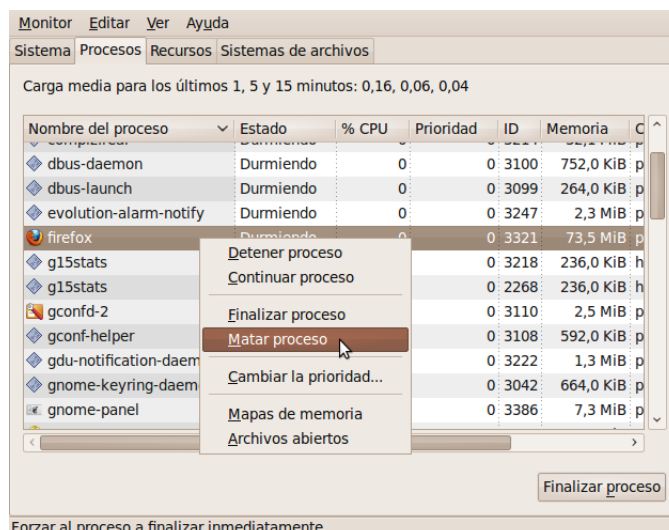
Para gestionar los procesos en Ubuntu utilizaremos una herramienta que normalmente viene instalada en el sistema. Esta herramienta es el **Monitor del sistema**, a la que podremos acceder directamente desde el buscador de *dash* tecleando *monitor* o desde *Inicio > Más aplicaciones > Sistema > Monitor del Sistema*.

Una vez lanzado se mostrará una pantalla como la de la figura que aparece a continuación.

Si pulsamos en la pestaña *Procesos*, accedemos a un cuadro de diálogo en el que se ven todos los procesos que hay en el sistema y sus estados. Sobre estos, seleccionando el proceso, y con el botón derecho del ratón, según el estado en el que se encuentren, se pueden realizar las siguientes operaciones:

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

Si pulsamos en la línea de menús *Ver > Dependencias* veremos qué procesos dependen de otros, de forma similar a como vimos en Windows.



### **COMANDOS TOP Y HTOP.**

Es el equivalente al Monitor del sistema en la línea de comandos. Muestra una lista de procesos. Permite realizar diferentes acciones sobre cada uno de ellos, como matarlo o cambiar su prioridad. Este comando es bastante complejo de utilizar, pero es conveniente conocer su existencia.

```
paco@ubuntupaco:~$ sudo top
```

Si deseamos ejecutar `htop`, tendremos que instalarlo como ya conocemos.

Este comando se utiliza para ver más información sobre los procesos, como los que hay en total, los que están corriendo, los que están parados, etc.

Además nos mostrará información del uso de CPU y memoria, o sea, de los recursos consumidos por los procesos que hay en el sistema. Ejecutaremos el comando y analizaremos los procesos activos, los detenidos, los pausados, el uso de CPU y de memoria.

### **COMANDO PS.**

El comando `ps` es el comando por excelencia utilizado para mostrar los procesos que se encuentran activos. Para ver todos los procesos activos en el terminal teclearemos:

```
paco@ubuntupaco:~$ sudo ps
paco@ubuntupaco:~$ sudo ps -Al
```

Podemos observar que el propio terminal es un proceso con el nombre `gnome-terminal` que tendrá asociado su `PID`. Lanzaremos, por ejemplo, el gestor de actualizaciones (proceso: `update-manager`), y el Centro de Software de Ubuntu (proceso: `software-center`) y analizaremos con el comando `ps` sus `PID` e incluso la identificación del proceso padre (`PPID`), si es que lo tiene.

### **COMANDO KILL Y KILLALL.**

Para finalizar un proceso mediante la consola es preciso conocer el `PID`. Por ejemplo podemos lanzar un proceso reloj (`xclock &`) y localizar su `PID` con el comando "`ps | grep xclock`"

Una vez que conozcamos el `PID` del proceso que queremos terminar, desde el terminal ejecutaremos la siguiente orden.

```
paco@ubuntupaco:~$ sudo kill 5800
```

donde 5800 es el `PID` del proceso a matar.

Es conveniente comprobar que el proceso ha finalizado y ya no se encuentra en ejecución con la orden `ps`.

Si el proceso, a pesar de ejecutar esta orden, no finaliza, podremos indicar una finalización incondicional del mismo, esté el proceso en el estado en el que esté. Ejecutaremos:

```
paco@ubuntupaco:~$ sudo -9 kill 5800
```

A veces puede ocurrir que al ver los procesos activos, el que se desea finalizar tiene distintos subprocesos lanzados, es decir, dispone de varios `PID`.

Para terminarlo definitivamente tendríamos que ejecutar varias veces la orden `kill`

El comando `killall` se utiliza para finalizar todos los subprocesos de un proceso, es decir, terminar el árbol de procesos.

Como cada subproceso tendrá su propio `PID`, no podremos indicar todos los `PID` en esta orden. Indicaremos el nombre del proceso. Supongamos que tenemos lanzado un terminal. El terminal es un proceso llamado `gnome-terminal`.

```
paco@ubuntupaco:~$ sudo killall gnome-terminal
```

Para pausar un proceso desde un terminal, también usaremos la orden `kill` de la siguiente forma. Conocido el `PID` del proceso a pausar, con `ps`, ejecutaremos la siguiente orden:

```
paco@ubuntupaco:~$ sudo kill -STOP 5800
```

Para reanudarlo basta con teclear:

```
paco@ubuntupaco:~$ sudo kill -CONT 5800
```

### 5.2.3 SERVICIOS

Linux ofrece multitud de servicios o servidores, estos pueden iniciar o arrancar junto con la carga del sistema o pueden después ser puestos a funcionar cuando se requieran (es lo mejor). Parte esencial de la administración de sistemas Linux es continuamente trabajar con los servicios que este proporciona, cosa que es bastante sencilla.

#### INICIANDO SERVICIOS MANUALMENTE, DIRECTORIO INIT.D

Dentro de esta carpeta ubicada en `/etc` o en `/etc/rc.d` dependiendo de la distribución, se encuentran una serie de *scripts* que permiten iniciar/detener la gran mayoría de los servicios/servidores que estén instalados en el equipo. Estos *scripts* están programados de tal manera que la mayoría reconoce los siguientes argumentos:

```
start
stop
restart
status
```

Los argumentos son autodescriptivos, y tienen permisos de ejecución, entonces siendo *root* es posible iniciar un servicio de la siguiente manera, por ejemplo *samba*:

```
#> /etc/rc.d/init.d/smb start
Starting Samba SMB daemon [OK]
```

Solo que hay que cambiar *start* por *stop* | *restart* | *status* para detenerlo, reiniciarlo (releer archivos de configuración) o chequear su estatus.

#### EL COMANDO SERVICE

En varias distribuciones, como Fedora o RedHat, existe el comando *service*, que permite también iniciar y/o detener servicios, de hecho funciona exactamente igual que si escribiéramos la ruta completa hacia el directorio *init.d*, con *service* se indica de la siguiente manera:

```
#> service mysql status
Checking for service MySQL: stopped
```

Si se desea iniciarlo:

```
#> service mysql start
Starting service MySQL [OK]
```

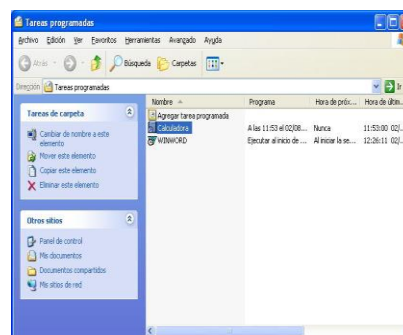
## 6. AUTOMATIZACIÓN DE TAREAS

### 6.1. WINDOWS


Con el Programador de tareas podemos programar secuencias de comandos, programas o documentos para ejecutarlos a la hora que se desee. El Programador de tareas se activa cada vez que se inicia Windows y se ejecuta en **segundo plano**.

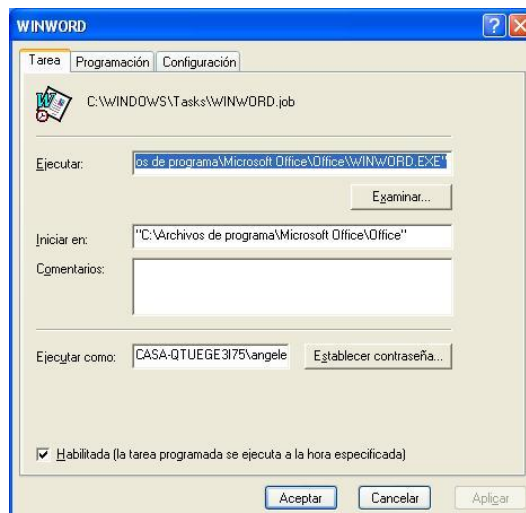
Mediante el Programador de tareas podemos hacer lo siguiente:

- Programar una tarea para ejecutarse diaria, semanal o mensualmente, o a determinadas horas, como al iniciar el sistema.
- Cambiar la programación de una tarea.
- Detener una tarea programada.
- Personalizar la forma en que se ejecutará una tarea en el momento programado.



### **PROGRAMAR UNA TAREA NUEVA**

1. Abrir el Programador de tareas (*Inicio/Todos los Programas/Accesorios/Herramientas del sistema/Tareas programadas*) o *Inicio/Panel de control/Tareas programadas*. Es imprescindible indicar la contraseña del usuario, en caso contrario no se realizara la tarea.
2. Hacer doble clic en  **AGREGAR TAREA PROGRAMADA**. Si se desea realizar la configuración avanzada de la tarea, activar la casilla de verificación **Abrir propiedades avanzadas para esta tarea** cuando hagas clic en **Finalizar**, que aparece en la página final del asistente.
3. Seguir las instrucciones. Confirmar que la fecha y la hora del equipo son correctas, ya que el Programador de tareas se basa en esta información para ejecutar tareas programadas. Para comprobar o cambiar esta información, hacer doble clic en el indicador de hora de la barra de herramientas.



### **MODIFICAR UNA TAREA PROGRAMADA**

1. Abrir el Programador de tareas.
2. Seleccionar la tarea que desea modificar y *Contextual/Propiedades*.
3. Elegir una o varias de las siguientes opciones:
  - Para cambiar el programa que se ejecutará, en la ficha **Tarea**, en el cuadro **Ejecutar** escribir la ruta de acceso del nuevo programa.
  - La ficha **Programación** nos permitirá cambiar la programación de la tarea.
  - La ficha **Configuración** nos permitirá personalizar la configuración de la tarea.

#### **Notas:**

- Si cambia la cuenta de usuario o el programa que se va a ejecutar, deberá suministrar la contraseña de la cuenta de usuario.
- Si el programa de la tarea requiere opciones de la línea de comandos, escríbalos en el cuadro **Ejecutar** después de la ruta de acceso de la tarea.
- Si la ruta de acceso al programa de la tarea incluye espacios, escribir la ruta de acceso completa de la tarea entre comilla dobles (" ")

### **DETENER UNA TAREA PROGRAMADA**

Abrir el Programador de tareas. Seleccionar la tarea que desee detener y *Contextual/Finalizar tarea*.

- Si se inicia una tarea programada y a continuación se detiene, *Finalizar tarea* no detiene el resto de los programas que la tarea programada haya iniciado.
- Si se detiene una tarea programada que está actualmente en ejecución, puede producirse un retraso (hasta 3 minutos) antes de que la tarea se detenga.
- Para reiniciar una tarea detenida, seleccionar la tarea que desee detener y *Contextual/Ejecutar*.

### **DESACTIVAR TEMPORALMENTE LAS TAREAS PROGRAMADAS**

Abrir el Programador de tareas. Abrir el menú *Avanzado/Pausar Programador de tareas*.

- El comando *Pausar Programador de tareas* es útil si no se desea que las tareas programadas se ejecuten al mismo tiempo que se instala software o se ejecuta otro programa, por ejemplo, un juego.
- Las tareas que debían ejecutarse mientras el Programador de tareas estaba en pausa no lo harán hasta la próxima hora programada.

- Para reanudar las programaciones de todas las tareas, abrir el menú *Avanzado/Continuar con Programador de tareas*.

### **ELIMINAR UNA TAREA PROGRAMADA**

Abrir el *Programador de tareas*. Seleccionar la tarea que desee quitar y *Contextual/Eliminar* (o **Supr**). Al quitar una tarea programada sólo se anula la programación de la tarea programada. El archivo de programa que ejecuta la tarea no se quita del disco duro.

En la opción *Avanzado/Ver registro* no permite hacer un seguimiento de las tareas que han sido programadas y las incidencias producidas en cada una de las tareas.

## **6.2. LINUX**

### **EL SERVICIO CRON**

El nombre cron viene del griego chronos que significa “tiempo”. El *cron* es un administrador regular de procesos en segundo plano (demonio) que ejecuta procesos o guiones a intervalos regulares (por ejemplo, cada minuto, día, semana o mes). Los procesos que deben ejecutarse y la hora en la que deben hacerlo se especifican en el fichero *crontab*.

### **FUNCIONAMIENTO DEL SERVICIO CRON**

El demonio *cron* inicia de */etc/rc.d/* o */etc/init.d* dependiendo de la distribución. *Cron* se ejecuta en el background, revisa cada minuto la tabla de tareas *crontab /etc/crontab* o en */var/spool/cron* en búsqueda de tareas que se deban cumplir. Como usuario podemos agregar comandos o scripts con tareas a *cron* para automatizar algunos procesos. Esto es útil por ejemplo para automatizar la actualización de un sistema o un buen sistema de respaldos.

### **EL FICHERO DE CONFIGURACION CRONTAB**

*Crontab* es un archivo de texto que guarda una lista de comandos (o *scripts*) a ejecutar en los tiempos especificados por el usuario. *Crontab* verificará la fecha y hora establecida para cada tarea y la ejecutara en segundo plano. Cada usuario puede tener su propio archivo *crontab*, de hecho */etc/crontab* se asume que es el archivo del *root*. Si los usuarios normales (e incluso *root*) desean generar su propio archivo de *crontab*, entonces utilizaremos el comando *crontab*.

*Crontab* es la manera más sencilla de administrar tareas de cron en sistemas multiusuario, ya sea como simple usuario de sistema o *root*.

### **AGREGAR TAREAS A CRONTAB**

Vamos a automatizar la actualización de un sistema.

Primero que nada haremos un script que será llamado por *cron* y contendrá todas las instrucciones que queremos que haga:

```
#!/bin/bash
apt-get update & apt-get -y upgrade
```

Guardamos el script como *actualizacion.sh* y cambiamos los permisos de ejecución del con:

```
chmod a+x ~/actualizacion.sh
```

Ejecutamos el script un par de veces para verificar que todo ejecute sin problemas, modificamos lo necesario (no debe contener errores, si no *cron* solo repetirá un error una y otra vez). Ahora a agregar la tarea a nuestro *crontab*.

Ejecutamos la edición del *crontab* con *crontab -e*, en algunas distribuciones (como ubuntu) nos da la opción de elegir el editor de textos que deseemos, los demás nos quedamos con *vi*. El archivo *crontab* quedará algo así:

```
# m h dom mon dow user command
```

donde:

- **m** corresponde al minuto en que se va a ejecutar el script, el valor va de 0 a 59
- **h** la hora exacta, los valores van de 0 a 23, siendo 0 las 12:00 de la medianoche.
- **dom** hace referencia al día del mes, por ejemplo se puede especificar 15 si se quiere ejecutar cada día 15
- **dow** significa el día de la semana, puede ser numérico (0 a 7, donde 0 y 7 son domingo) o las 3 primeras letras del día en inglés: mon, tue, wed, thu, fri, sat, sun.
- **user** define el usuario que va a ejecutar el comando, puede ser root, u otro usuario diferente siempre y cuando tenga permisos de ejecución del script.
- **command** refiere al comando o a la ruta absoluta del script a ejecutar, ejemplo: /home/usuario/scripts/actualizar.sh, si acaso llama a un script este debe ser ejecutable

Ejemplos:

```
15 10 * * * usuario /home/usuario/scripts/actualizar.sh
    Ejecutará el script actualizar.sh a las 10:15 a.m. todos los días
15 22 * * * usuario /home/usuario/scripts/actualizar.sh
    Ejecutará el script actualizar.sh a las 10:15 p.m. todos los días
00 10 * * 0 root apt-get -y update Usuario root
    Ejecutará una actualización todos los domingos a las 10:00 a.m
45 10 * * sun root apt-get -y update
    Usuario root ejecutará una actualización todos los domingos (sun) a las 10:45 a.m
30 7 20 11 * usuario /home/usuario/scripts/actualizar.sh
    El día 20 de noviembre a las 7:30 el usuario correrá el script
30 7 11 11 sun usuario /home/usuario/scripts/pastel_con_velitas.sh
    El día 11 de noviembre a las 7:30 a.m. y que sea domingo, el usuario festejará su sysadmin (o sea a mí)
01 * * * * usuario /home/usuario/scripts/molestorecordatorio.sh
    Un molesto recordatorio cada minuto de cada hora todos los días (NO recomendable).
```

También se pueden usar rangos especiales:

```
30 17 * * 1,2,3,4,5 root /root/aviso.sh
    A las 5:30 de la tarde todos los días de lunes a viernes.
00 12 1,15,28 * * root /root/aviso.sh
    A las 12 del día todos los días 1, 15 y 28 de cada mes (ideal para nóminas)
```

Si esto resulta confuso, crontab maneja cadenas especiales para definir estos rangos.

```
@reboot Ejecuta al inicio de sesión
@yearly o @annually ejecuta sólo una vez al año: 0 0 1 1 *
@monthly ejecuta una vez el primer día de cada mes: 0 0 1 * *
@weekly el primer minuto de la primer hora de cada semana. 0 0 * * 0".
@daily o @midnight a las 12:00A.M de cada día. 0 0 * * *
@hourly al primer minuto de cada hora: 0 * * * *
```

Su uso es muy sencillo.

```
@hourly usuario home/usuario/scripts/molestorecordatorio.sh
@monthly usuario /home/usuario/scripts/respaldo.sh
@daily root apt-get update && apt-get -y upgrade
```

### **ADMINISTRACIÓN DE TRABAJOS EN CRON**

Reemplazar el existente archivo crontab con un archivo definido por el usuario

```
crontab archivo
```

Editar el archivo crontab del usuario, cada línea nueva será una nueva tarea de crontab.

```
crontab -e
```

Lista todas las tareas de crontab del usuario

```
crontab -l
```

Borra el crontab del usuario

```
crontab -d
```

Define el directorio de crontab del usuario (este debe tener permisos de escritura y ejecución del usuario)

```
crontab -c dir
```

Manejar el crontab de otro usuario, ejemplos:

```
crontab -u usuario
```



## 7. RECURSOS COMPARTIDOS

### 7.1. WINDOWS

En el punto anterior hemos visto cómo podemos modificar las ACL de los recursos para que sean usadas por los usuarios y grupos LOCALES, es decir, aquellos que residen en nuestra propia máquina.

Vamos a ver ahora como modificamos esas ACL para permitir el uso por parte de usuarios y grupos que no pertenecen a nuestra propia máquina, sino que van a entrar en nuestro sistema a través de la red.

Windows presenta dos posibilidades a la hora de usar una red, trabajar en lo que se conoce como grupo de trabajo en la que todas las maquinas son iguales en derechos y obligaciones (red entre pares, o peer to peer) o bien trabajar en una red centralizada, donde existe una máquina que ejecuta un rol de control sobre las demás y que corre un sistema operativo servidor como Windows 2003 (Dominio).

Dejaremos el tema de cómo trabajar en un Dominio para los apuntes sobre Windows 2003, y nos centraremos ahora en los grupos de trabajo.

Obviamente para poder trabajar en un grupo de trabajo, debemos tener nuestro Windows bien configurado para trabajar en red. Para ello comprobad que:

Los equipos se encuentra en la misma LAN. Para ello se necesita usar el protocolo TCP/IP y contar con una dirección IP válida. (*Panel de control – Conexiones de Red – Propiedades de nuestra conexión – Propiedades de TCP/IP*).

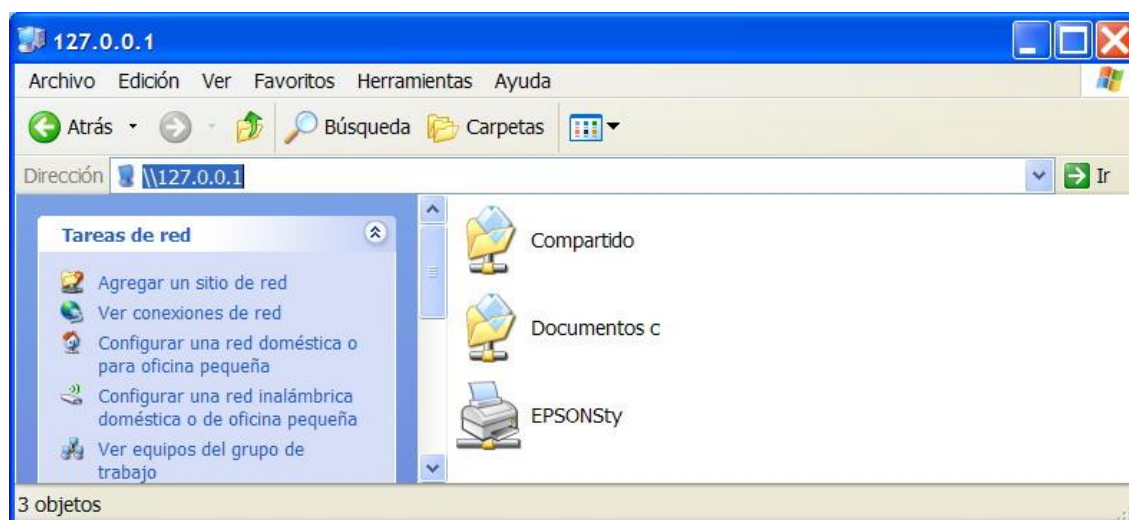
La máquina cuenta con un nombre descriptivo, y que está trabajando en el mismo grupo de trabajo que el resto de equipos. (*Propiedades de Mi PC – Nombre de Equipo*)

El cortafuegos del equipo está configurado para permitir la compartición en red. (*Panel de Control – Centro de Seguridad*).

La forma más fácil de ver lo que un equipo comparte es hacerlo desde el explorador de archivos. Para ello, iros al explorador y escribid en su barra de direcciones

\\Nombre\_Maquina

\\Dirección\_IP\_remota



En la imagen anterior, la máquina nos ha dado permiso porque es la nuestra, claro (127.0.0.1, localhost) pero ¿qué ocurre si le damos la dirección de otra máquina?

Pensemos un poco.... **El equipo remoto al que queremos conectarnos tiene sus recursos, con sus ACL y demás** como vimos anteriormente. De repente ve como un usuario intenta acceder, autorizarse, pero no desde el equipo local (con su SID y demás) sino desde la red. Es obvio que en este ambiente no se pueden usar las SID (recordar que la SID incluye dentro un número aleatorio que depende de cada equipo, por lo que una máquina no reconoce los SID normales de otra máquina).

Así que la máquina remota a la que intentamos acceder, insiste en autorizarnos, y lo hace de la siguiente forma:



Comprueba el nombre de usuario y la contraseña que está usando en su máquina el usuario que intenta colarse por la red. Si ese **nombre de usuario y contraseña coinciden** con un **nombre de usuario y contraseña locales** de la propia máquina, supone que es el usuario indicado y le deja pasar.

En caso de que el punto 1 no se cumpla, comprueba si en la máquina que intenta entrar esta **activa la cuenta de invitado** y el sistema está configurado para permitir que se use la cuenta Invitado para red, y los permisos de red y la seguridad del recurso también lo permiten.

Si no se cumplen el punto 1 ni 2, aparece una pantalla como la indicada arriba que pregunta directamente un nombre de usuario y contraseña para entrar en el equipo.

Un punto importante que hay que conocer: Una vez que un equipo nos deja acceder al mismo nos concede una credencial, esta credencial no se borra inmediatamente, sino que es recordada en el equipo durante un tiempo. Esto es importante saberlo por que podremos encontrarnos con un equipo que nos concede el paso, borramos la cuenta que permite ese paso, y sin embargo al volver a probar resulta que seguimos entrando sin problemas. Simplemente ocurre que las credenciales siguen estando activas, aunque la situación que las genero haya desaparecido. Para actualizar estas credenciales la forma más segura es reiniciando el equipo.

### 7.1.1 RECURSOS COMPARTIDOS MEDIANTE CUENTA LOCAL.

Bien, creemos una carpeta y compartámosla en red, para ir viendo punto por punto como se realiza esta acción.

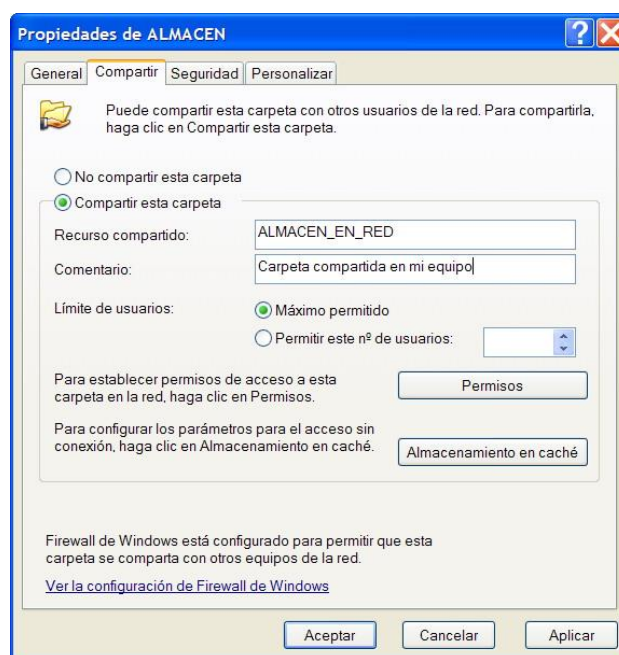
Creamos un usuario en nuestro sistema con el nombre RINO y contraseña 1234.

Cread una carpeta ALMACEN en la raíz de vuestro volumen.

Accedemos a sus propiedades y a la pestaña Compartir. (Si no se ve esta pestaña, es que tenéis activado la opción de uso compartido simple de archivos y habrá que desactivarla (Mi PC – Herramientas – Opciones de Carpeta – Ver – Habilitar el uso compartido simple de archivos).

Allí indicamos que queremos compartir el archivo, le ponemos un nombre y un comentario, y entramos en la opción de Permisos. Esta opción será la que nos indique que usuarios pueden entrar desde la red a dicho recurso.

Agregamos al usuario RINO en Permisos, con control total.



Accedemos a la pestaña Seguridad de ALMACEN. Añadimos al usuario RINO con todos los permisos (**No es suficiente con añadir al usuario en Permisos, también hay que añadirlo en Seguridad**).

Ahora que le hemos concedido al usuario RINO el derecho a entrar en la carpeta ALMACEN, nos situamos en el segundo ordenador.

Creamos la cuenta RINO con contraseña 1234.

Abrimos sesión con dicha cuenta RINO en la segunda máquina.

Para hacer lo mismo desde el explorador, escribid en la barra de direcciones del explorador \\Nombre\_Maquina\_Almacen

Hacemos clic con el botón derecho en la carpeta ALMACEN\_EN\_RED y escogemos la opción conectar a unidad de red, con lo que crearemos otro volumen para dicho recurso en nuestro sistema.

Si escribimos directamente \\Nombre\_Maquina\_Almacen\ALMACEN\_EN\_RED accederemos al recurso, sin crear ninguna letra de volumen.

Veremos que accedemos usando la cuenta del usuario RINO, ya que le hemos dado los permisos necesarios, tanto en permisos como en seguridad.

## 7.1.2 RECURSOS COMPARTIDOS Y ACCESO ANÓNIMO.

Hasta aquí hemos visto como acceder usando una cuenta de usuario común en los dos equipos. Veamos ahora como podemos activar el acceso de usuarios anónimos usando la cuenta Invitado. Esto nos permitirá ofrecer recursos compartidos en red para que pueda usarlo cualquier maquina sin preocuparnos de las cuentas locales que dicha maquina tenga.

Para ello, realizamos lo siguiente:

Compartimos un recurso en nuestra máquina.

Nos aseguramos de que la cuenta Invitado esta activa.

Accedemos a **secpol.msc** para tocar algunas directivas de grupo que deben ser cambiadas para permitir el acceso anónimo de usuarios:

Desde directivas locales – Asignación de derechos de usuario:

Denegar el acceso desde la red a este equipo. Nos aseguramos que la cuenta Invitado no está incluida en esta directiva.

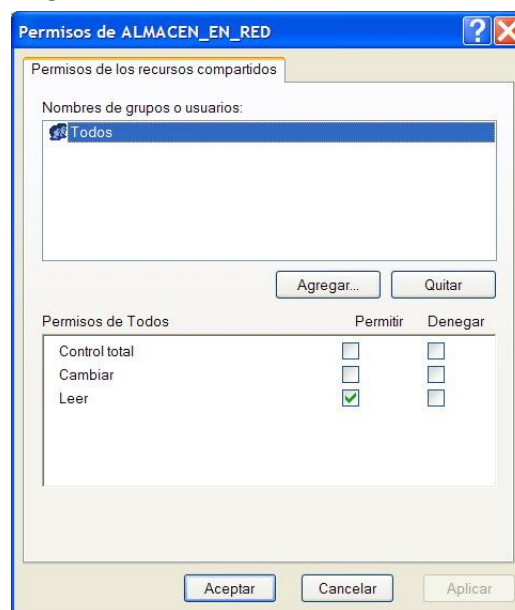
Tener acceso a este equipo desde la red. Nos aseguramos que la cuenta Invitado está incluida en esta directiva.

Desde directivas locales – Opciones de seguridad.

Acceso de red: deja los permisos de Todos para aplicarse a usuarios anónimos. Si la habilitamos, basta con que en permisos al compartir agreguemos el grupo Todos. Si esta deshabilitado el grupo Todos no incluye la cuenta Invitado, por lo que habrá que agregar específicamente la cuenta Invitado.

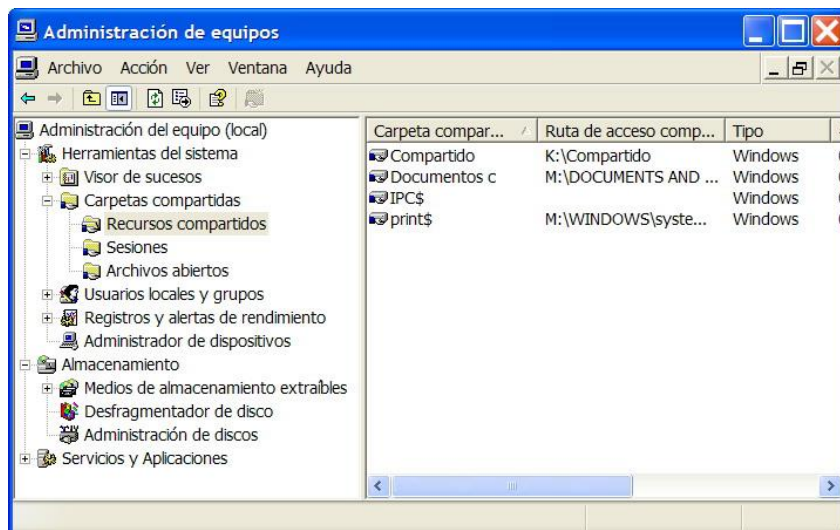
Con estos pasos, conseguiremos un sistema en el cual, si compartimos un recurso, en sus propiedades en permisos indicamos que puede ser usada por Todos y en su seguridad incluimos el usuario Invitado, cualquier usuario de la red podrá entrar en el recurso con los permisos que indiquemos para la cuenta Invitado. Esto se conoce como acceso anónimo.

Modificando otras directivas de grupo, es posible crear otros esquemas de compartición y trabajo en red, pero este es uno de los más útiles y simples, ya que permite recursos nominales y recursos anónimos.



La forma de crear un recurso compartido oculto en Windows es muy simple, basta con terminar el nombre de dicho recurso con el símbolo \$. Así, si creamos una carpeta VIDEOS y la compartimos en red con AVI\$, dicha carpeta no será visible ni con Net View ni desde el explorador, sin embargo dicho recurso se podrá usar sin ningún problema indicando su nombre completo \\Equipo\Avi\$.

De hecho, Windows comparte de forma predeterminada TODOS nuestros volúmenes completos, con los nombres C\$, D\$, etc. Estos recursos compartidos se usan desde la Administración de sistemas, y no deben ser desactivados. Esta forma de trabajar es muy cómoda, puesto que un Administrador siempre podrá acceder a sus equipos desde red, sin tener que compartir implícitamente ningún recurso.



Si queremos ver nuestros recursos compartidos, incluidos algunos ocultos, lo podemos hacer desde la consola **COMPMGMT.MSC**

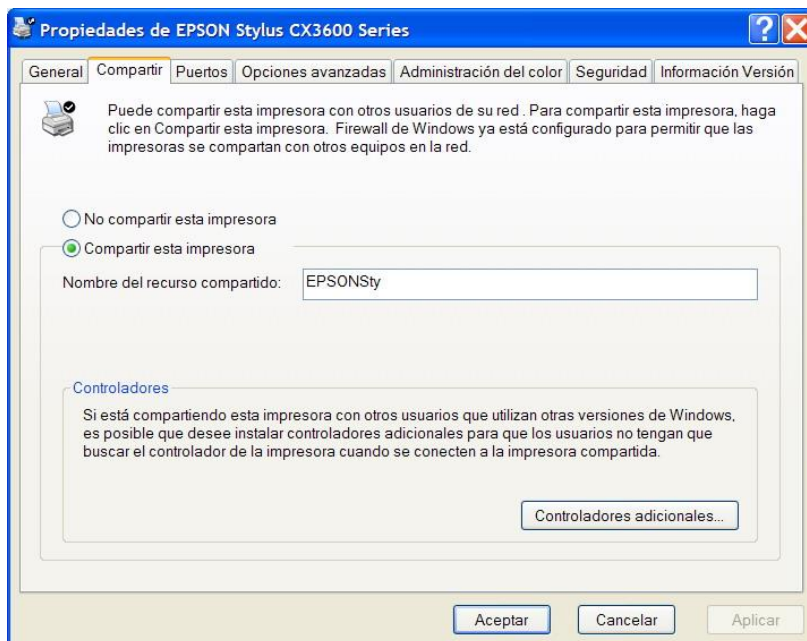
Desde esta consola también podemos ver las sesiones abiertas por usuarios remotos en nuestro equipo (e incluso desconectar dichas sesiones y también podemos ver que archivos están abiertos por dichas sesiones).

El recurso compartido IPC\$ es un recurso utilizado para funciones avanzadas de Administración de sistemas.

### 7.1.3 RECURSOS COMPARTIDOS. IMPRESORAS.

Compartir una impresora es prácticamente igual que compartir una carpeta. Basta con que accedamos a nuestra impresora (*Inicio – Impresoras y Faxes, o bien accedemos desde Panel de Control*), y en sus propiedades indiquemos compartir.

Desde esta opción podemos compartir la impresora, indicar un nombre de recurso compartido, y podemos instalar en nuestros sistemas controladores adicionales. Estos controladores permiten que cuando una máquina remota quiera acceder a nuestra impresora no tenga que instalar ningún driver (controlador) adicional, sino que se los bajara automáticamente desde nuestra propia máquina.

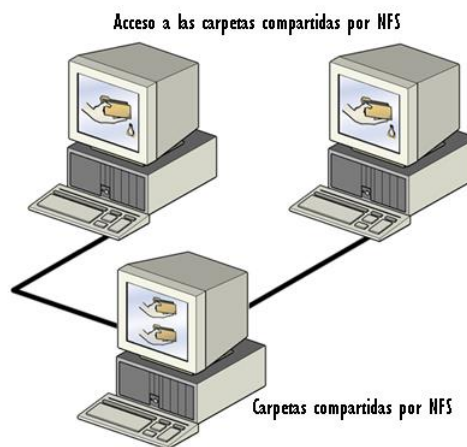


## 7.2. LINUX

### 7.2.1 ¿QUÉ ES NFS?

NFS (Network File System - Sistema de Archivos en Red) es el sistema nativo utilizado por Linux para compartir carpetas en una red. Mediante NFS, un servidor puede compartir sus carpetas en la red. Desde los PCs de los usuarios se puede acceder a dichas carpetas compartidas y el resultado es el mismo que si estuvieran en su propio disco duro.

Básicamente, NFS permite a PCs que utilizan Linux, compartir y conectarse a carpetas compartidas entre sí. Existen otras alternativas para compartir carpetas en una red como samba, ssh o ftp, pero el sistema recomendado para compartir carpetas entre sistemas Linux es NFS.



### 7.2.2 INSTALACIÓN DE NFS

Para poder disfrutar del servicio de compartir carpetas en la red mediante NFS, en el PC servidor es necesario instalar el paquete del **servidor NFS**. Lo normal es que todos los PCs dispongan del paquete servidor de NFS ya que en cualquier momento puede existir la necesidad de tener que compartir una carpeta desde cualquier PC, aunque lo habitual es que el único que comparta sea el servidor. Que un PC de un usuario tenga instalado el paquete del servidor NFS, no significa que automáticamente esté compartiendo su sistema de archivos en la red. Para ello es necesario configurar y arrancar el servicio.

Si deseamos instalar la última versión disponible, podemos hacerlo con apt-get desde una consola de root:

```
# apt-get install nfs-common nfs-kernel-server
```

### 7.2.3 CONFIGURACIÓN DEL SERVIDOR NFS

Antes de arrancar el servicio NFS, es necesario indicar qué carpetas deseamos compartir y si queremos que los usuarios accedan con **permisos de solo lectura o de lectura y escritura**. También existe la posibilidad de establecer desde qué PCs es posible conectarse. Estas opciones se configuran en el archivo **/etc/exports**

En cada línea del archivo de configuración del servidor NFS **/etc/exports**, se puede especificar:

- La carpeta que se quiere compartir
- El modo en que se comparte (solo lectura 'ro' o lectura y escritura 'rw' )
- Desde qué PC o PCs se permite el acceso (nombre o IP del PC o rango de IPs)

A continuación mostramos un sencillo archivo **/etc/exports** para configurar algunas carpetas compartidas

```
// Ejemplo de archivo /etc/exports de configuración del servidor NFS:
# Compartir la carpeta home del servidor en modo
# lectura y escritura y accesible desde la red 192.168.0.0/24
/home 192.168.0.0/255.255.255.0(rw)

# Compartir carpeta tmp a todos como 'solo-lectura'
/tmp *(ro)

# Compartir carpeta /var/log a un PC como 'solo-lectura'
/var/log 192.168.0.211(ro)
```



Cuando se comparte por NFS, se recomienda restringir al máximo los permisos. Si los usuarios no tienen la necesidad de escribir, debemos compartir con permiso de 'solo lectura'. Si los usuarios solo se conectan desde nuestra red 192.168.0.0/24, debemos permitir el acceso solo desde dicha red.

**Nota:** Los permisos de compartición por NFS no excluyen a los permisos del sistema unix sino que **prevalecen los más restrictivos**. Si una carpeta está compartida con permiso NFS de lectura y escritura pero en los permisos del sistema solo disponemos de permiso de lectura, no podremos escribir. Si una carpeta está compartida con permisos NFS de lectura y disponemos de permisos de lectura y escritura en el sistema, tampoco podremos escribir. Para poder escribir necesitaremos disponer permiso de lectura y escritura tanto en los permisos del sistema como en los permisos de compartición NFS. De igual forma, si compartimos la carpeta /home con permisos de lectura y escritura pero el usuario pepe solo tiene acceso a la carpeta /home/pepe, no podrá acceder a ninguna otra carpeta dentro de /home ya que los permisos del sistema se lo impedirán.

### **ARRANQUE Y PARADA MANUAL DE NFS**

Para que el servidor NFS funcione, es necesario que esté arrancado el servicio **portmap**, por lo tanto, la primera acción será iniciar portmap por si no estuviera arrancado:

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

Para poner en marcha el servicio NFS, o cada vez que modifiquemos el archivo /etc/exports, debemos reiniciar el servidor NFS, mediante el comando:

```
sudo /etc/init.d/nfs-kernel-server restart
```

Si deseamos detener el servidor NFS, debemos ejecutar:

```
sudo /etc/init.d/nfs-kernel-server stop
```

### **ACCESO A CARPETAS COMPARTIDAS POR NFS**

Para poder acceder desde un PC a una carpeta compartida por NFS en un servidor, lo primero que **tenemos** que hacer es instalar los paquetes **portmap** y **nfs-common** que nos permitirán acceder como clientes:

```
// Instalar portmap y nfs-common y reiniciar portmap
sudo apt-get install portmap nfs-common
sudo /etc/init.d/portmap restart
```

Ahora ya estaremos en condiciones de **montar** la carpeta compartida en nuestro sistema de archivos. De esta manera, el acceso a la carpeta compartida es exactamente igual que el acceso a cualquier otra carpeta de nuestro disco duro.

Por ejemplo, supongamos que un servidor comparte por NFS una carpeta llamada /fotos. En el PC cliente podemos crear una carpeta llamada /fotos-servidor y montar sobre ella la carpeta compartida en el servidor. Para ello, en el cliente y como root ejecutaríamos el siguiente comando:

```
// Montar carpeta compartida por NFS
sudo mount -t nfs ip-del-servidor:/fotos /fotos-servidor
```

A partir de este momento, podemos comprobar que nuestra carpeta /fotos-servidor contiene la información de la carpeta /fotos del servidor. Si disponemos de permisos de lectura y escritura, podemos incluso crear o modificar los archivos dentro de nuestra carpeta /fotos-servidor y los cambios se estarán guardando realmente en la carpeta /fotos del servidor.

Para realizar el montaje, debemos hacerlo sobre una carpeta existente en nuestro sistema. Si dicha carpeta de nuestro sistema contiene archivos, éstos no estarán accesibles ya que la carpeta nos mostrará los archivos remotos.

Si al intentar montar la carpeta NFS no funciona suele ser por una de estas tres razones: por un problema en la red, un problema en el servidor o un problema en el cliente. Para averiguar si el problema es del servidor o no, podemos intentar montar por NFS la carpeta en el propio servidor, usando la IP 127.0.0.1. Si funciona entonces el problema estará en la red o en el cliente. Si hacemos

ping del servidor al cliente y no hay cortafuegos, el problema será en el cliente. Podemos intentar hacer una reinstalación del cliente igual que la instalación en el servidor. Ejecuta en el cliente los siguientes comandos: `apt-get install nfs-common nfs-kernel-server`, luego `/etc/init.d/portmap restart`, después `/etc/init.d/nfs-kernel-server restart` y finalmente intentar montar la carpeta. `mount` para nada.

Para desmontar una carpeta, tan solo debemos ejecutar el comando **umount** seguido del nombre de la carpeta en la que está montada, ejemplo:

```
sudo umount /fotos-servidor
```

Si deseamos que nuestro PC monte siempre de forma automática una carpeta compartida por NFS cuando iniciemos nuestro Linux, existe la posibilidad de añadir en el archivo `/etc/fstab` una línea como por ejemplo:

```
# Montaje automático al iniciar el PC
# Añadir en /etc/fstab
ip-del-servidor:/fotos /fotos-servidor nfs
```

De esta manera, cuando arranquemos nuestro PC, la carpeta `/fotos` del servidor quedará automáticamente montada sobre nuestra carpeta `/fotos-servidor` y no tendremos que ejecutar el comando `mount` para nada.

**Nota:** Si al intentar montar la carpeta NFS no funciona suele ser por una de estas tres razones: por un **problema en la red**, un **problema en el servidor** o un **problema en el cliente**. Para averiguar si el problema es del servidor o no, podemos intentar montar por NFS la carpeta en el propio servidor, usando la IP 127.0.0.1. Si funciona entonces el problema estará en la red o en el cliente. Si entre el cliente y el servidor hay conectividad y no hay ningún cortafuegos que impida las comunicaciones, el problema estará en el cliente. Podemos intentar hacer una reinstalación del cliente igual que la instalación en el servidor. Ejecuta en el cliente los siguientes comandos: `apt-get install nfs-common nfs-kernel-server`, luego `/etc/init.d/portmap restart`, después `/etc/init.d/nfs-kernel-server restart` y finalmente intentar montar la carpeta.