

WUOLAH



postdata9

www.wuolah.com/student/postdata9



32569

sesion2.pdf

Módulo I (actualizado)



2º Sistemas Operativos



Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.





**KEEP
CALM
AND
ESTUDIA
UN POQUITO**

Sesión 2:

Herramientas de administración del sistema de archivos

Índice:

1. Partición de dispositivos de almacenamiento secundario

1.1 ¿Cuántas particiones hago en mi dispositivo?

1.2 ¿Qué directorios de primer nivel de FHS deberían estar soportados por una partición independiente?

1.2.1 Preparación previa a la partición mediante un archivo especial de dispositivo

1.2.2 Preparación previa a la partición mediante un pen drive

2. Asignación de un SA a una partición (formateo lógico)

3. Ajuste de parámetros configurables de un SA y comprobación de errores

4. Montaje y desmontaje de SA

5. Administración de software

5.1 Gestores de paquetes

5.2 APT y YUM

5.3 RPM

6. Administración de cuotas

7. Preguntas de repaso

1. Partición de dispositivos de almacenamiento secundario

Para poder utilizar un dispositivo de almacenamiento secundario, hay que establecer secciones dentro del dispositivo físico para alojar un SA concreto y que sean identificables.

Todos los dispositivos de almacenamiento secundario pueden agruparse en 3 tipos:

- discos duros;
- memorias flash;
- dispositivos RAID.

Un **sector** es la mínima unidad de almacenamiento de información en un dispositivo de almacenamiento secundario. Cuando creamos una partición, debemos asociarle una **etiqueta**; esta etiqueta se almacena mediante un código numérico que determina el tipo de la partición. Un ejemplo de etiqueta puede ser **0x83**, que indica tipo **ext2**; ó **0x83** que indica tipo **swap**.

Para poder consultar la información de las particiones, tenemos las **tablas de particiones**, en la que cada entrada se corresponde con la información de una partición:

- dónde comienza y finaliza la partición;
- el tipo;
- si es partición de arranque.

En la primera tabla que se hizo, esta información se almacenaba como parte el sector de arranque maestro y tenía espacio solo para 4 entradas. Actualmente, dichas 4 particiones se conocen como **particiones primarias**.

Una partición primaria puede subdividirse, y cada una de estas subdivisiones se conoce como **particiones lógicas**. La partición primaria que aloja las particiones lógicas se conoce como **partición extendida** y tiene su propio tipo de partición: **0x05**. El número de particiones está limitado y depende del sistema operativo.

1.1 ¿Cuántas particiones hago en mi dispositivo?

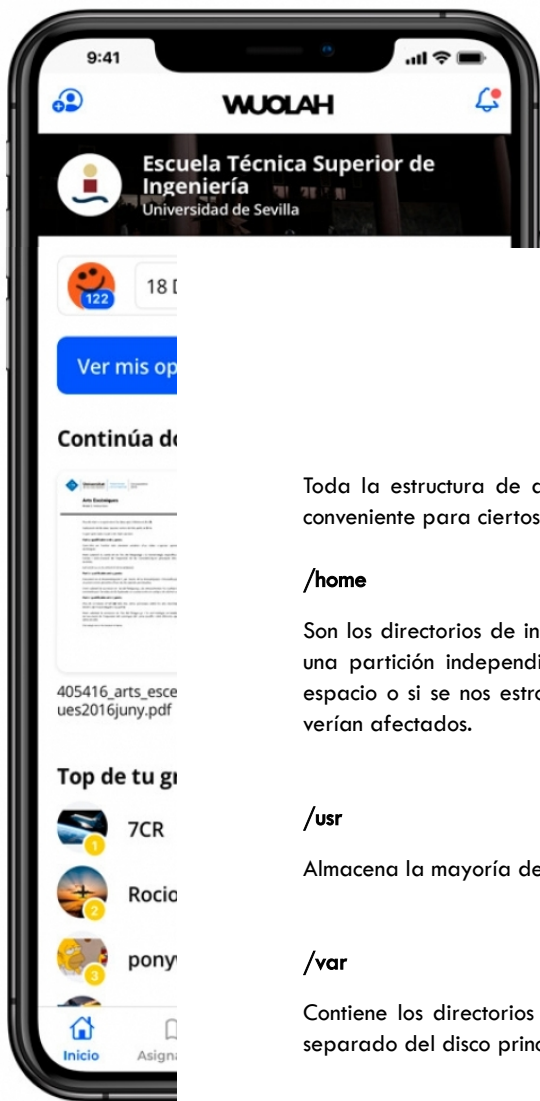
Esto depende de si nuestro dispositivo actuará como **dispositivo de arranque** (queremos que nuestro SO arranque desde el dispositivo, con lo que lo primero que utilizará será la BIOS) o no.

Si queremos que nuestro SO arranque desde el dispositivo, podríamos hacer las siguientes particiones:

- una partición primaria;
- una o más particiones swap;
- ninguna o las que queramos particiones primarias y lógicas.

Si queremos que nuestro dispositivo almacene información, sin actuar como dispositivo de arranque, las particiones serían las siguientes:

- una o más particiones primarias o lógicas;
- ninguna o las que queramos particiones swap.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



1.2 ¿Qué directorios de primer nivel del FHS deberían estar soportados por una partición independiente?

Toda la estructura de directorios del FHS puede estar soportada en una única partición, sin embargo, es conveniente para ciertos directorios almacenarlos en particiones independientes. Estos directorios son:

/home

Son los directorios de inicio de todos los usuarios con cuenta en el sistema. Es recomendable almacenarlo en una partición independiente por si queremos actualizar a una nueva versión el SO, si nos quedamos sin espacio o si se nos estropea el disco, ya que al estar en una partición independiente, nuestros datos no se verían afectados.

/usr

Almacena la mayoría de los ejecutables binarios del sistema, así como el árbol de las fuentes del kernel.

/var

Contiene los directorios de SPOOL y todos los logs y colas del sistema. Esto sería necesario que estuviera separado del disco principal.

Actividad 2.1 Partición de un dispositivo: "USB pen drive" ó "memory stick"

Vamos a establecer dos particiones primarias en un dispositivo USB en el que la primera de ellas albergará un SA ext3 y la segunda, un ext4; ambas siendo particiones de tipo Linux 0x83. El tamaño de cada una de las particiones debe ser mayor a 512MB.

[Seguir el guión](#)

2. Asignación de un SA a una partición (formateo lógico)

Una vez tenemos nuestro dispositivo de almacenamiento secundario con las particiones, debemos asignarle a cada una de ellas el sistema de archivos adecuado. En Linux, además de swap (para las particiones dedicadas a intercambio), tenemos 3 sistemas de archivos (ext viene de *extended file system*):

- **ext2:** SA de disco de alto rendimiento usado en Linux para discos duros, memorias flash y medios extraíbles. Este es una extensión de ext y ofrece un mejor rendimiento en cuanto a velocidad de transferencia de E/S y uso de CPU;
- **ext3:** es una versión de ext2 que incluye **journaling**; esto es un SA que mantiene un archivo especial, **journal** (registro diario), en el que se almacena la información necesaria que se utiliza para reparar cualquier inconsistencia cuando se produzca un apagado incorrecto del sistema;
- **ext4:** es el estándar actual de las distribuciones Linux, tiene unas estructuras similares a las del ext3 y presenta las siguientes mejoras:
 - **extents:** los extents han sido introducidos para reemplazar el esquema de bloques usado por ext2/3. Un extent es un conjunto de bloques físicos contiguos que mejoran el rendimiento al trabajar con ficheros de gran tamaño y reducen la fragmentación;
 - **asignación retardada de espacio en el disco (allocate-on-flush):** antes se reservaban los bloques necesarios antes de escribir, sin embargo en este SA la reserva se realiza cuando se va a escribir en el disco. Con esto, se mejora el rendimiento y se reduce la fragmentación, ya que se hace la reserva en base al tamaño real del fichero.

Actividad 2.2. Creación de sistemas de archivos

El objetivo es simplemente formatear lógicamente las particiones creadas con anterioridad de forma consistente con el tipo de SA que se estableció que iba a ser alojado. En la primera partición crearemos un SA de tipo ext3 y en la segunda un ext4.

La orden que permite establecer un SA de los reconocidos dentro del sistema Linux sobre una partición de disco es `mke2fs` (consulta el manual en línea para familiarizarte con sus opciones). El resultado de la ejecución de esta orden es el formateo lógico de la partición escogida utilizando el SA que se ha seleccionado.

Utiliza el manual en línea para conocer cómo ejecutar la orden de creación de SA. `mke2fs` es la orden genérica para creación de sistemas de archivos. Como requisito es necesario que establezcas dos etiquetas de volumen para los SAs: `LABEL_ext3` para la primera partición y `LABEL_ext4` para la segunda. Debería aparecer un listado en pantalla similar al siguiente.

```
$ mkfs.ext3 -L 'LABEL_ext3' /dev/loop0
```

```
$ mkfs.ext4 -L 'LABEL_ext4' /dev/loop1
```

3. Ajuste de parámetros configurables de un SA y comprobación de errores

Con la orden **tune2fs**, podemos ajustar los parámetros de nuestros SAs. Tenemos varias opciones interesantes con esta orden:

- **-l**: muestra el contenido del superbloque del SA;
- **-c max-mount-counts <dispositivo>**: para establecer el número máximo de montajes que se pueden realizar sin realizar una comprobación de consistencia del SA;
- **-L label <dispositivo>**: le asigna una etiqueta al dispositivo.
 - Para realizar esto, también tenemos la orden **elabel2**.

Las estructuras de metainformación del SA puede llegar a corromperse, lo que puede provocar degradación en el rendimiento del sistema y pérdidas de información. Linux, para solucionarlo, automatiza el proceso de comprobación de la consistencia según el número de montajes que se han realizado; para automatizar esto, Linux proporciona la orden **fsck**, que actúa sobre las estructuras de la metainformación, no sobre el contenido.

Hay varias situaciones de inconsistencias de metadatos del SA:

- bloques asignados simultáneamente a varios archivos;
- bloques asignados a un archivo que están marcados como libres;
- bloques libres que están marcados como asignados;
- inconsistencia del número de enlaces a un determinado archivo;
- inodos marcados como ocupados pero no están asociados a ningún archivo.

Actividad 2.3. Personalización de los metadatos del SA

Consultando el manual en línea para la orden **tune2fs** responde a las siguientes preguntas:

- a) ¿Cómo podrías conseguir que en el siguiente arranque del sistema se ejecutara automáticamente **e2fsck** sin que se haya alcanzado el máximo número de montajes?

Con la opción **-C** y estableciendo un número de veces.

- b) ¿Cómo podrías conseguir reservar para uso exclusivo de un usuario **username** un número de bloques del sistema de archivos?

\$ tune2fs /dev/loop0 -r [numero_bloques] -u username

4. Montaje y desmontaje de un SA

Cuando disponemos de nuestros nuevos SAs, solo nos queda ponerlos a disposición de los usuarios. Para poder crear archivos y directorios en ellos, es necesario hacerlos visibles en el espacio de nombres. Para ello, utilizamos **mount**.

Al realizar **mount** (montaje de un SA), estamos añadiendo una nueva rama al espacio de nombres actualmente en uso, lo que hará al dispositivo visible y preparado para ser utilizado. Para montar un SA, debemos indicar el directorio en el que se montará (**punto de montaje**), que es dónde crecerá la nueva rama del espacio de nombres) y el nombre del archivo especial de dispositivo.

Cuando no es necesario acceder a la información de dicho SA, debemos desmontar el SA (**umount**), con lo que el SA deja de estar disponible en el espacio de nombres. No es posible desmontar un SA si está siendo utilizado.

Actividad 2.4. Montaje de sistemas de archivos

Utiliza el manual en línea para descubrir la forma de montar nuestros SAs de manera que cumplas los siguientes requisitos:

- El SA etiquetado como LABEL_ext3 debe estar montado en el directorio /mnt/SA_ext3 y en modo de solo lectura.

```
$ mount /dev/loop0 -r /mnt/SA_ext3
```

- El SA etiquetado como LABEL_ext4 debe estar montado en el directorio /mnt/LABEL_ext4 y debe tener sincronizadas sus operaciones de E/S de modificación de directorios.

```
$ mount /dev/loop1 -o dirsyc /mnt/SA_ext4
```

Actividad 2.5. Automontaje de Sistemas de Archivos

Escribe las dos líneas necesarias en el archivo /etc/fstab para que se monten automáticamente nuestros dos SA en el arranque del sistema con los mismos requisitos que se han pedido en la Actividad 2.4.

```
$ /dev/loop0 /mnt/SA_ext3 ext3 auto,ro 0 0
```

```
$ /dev/loop1 /mnt/SA_ext4 ext4 auto,dirsyc 0 0
```



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



5. Administración de software

En Linux, se estructura el software a instalar en paquetes. Un **paquete software** es un archivo con un conjunto de programas complementarios con dependencias con otros paquetes. Los metadatos del paquete software incluyen la especificación de las dependencias del propio paquete, scripts de ayuda, atributos de archivo y otra información descriptiva.

Las distribuciones de sistemas operativos incluyen gestores de paquetes, para la administración de los paquetes software.

5.1 Gestores de paquetes

Existe un gran número de gestores de paquetes con características diferentes y su disponibilidad depende del SO y/o distribución específica. Los paquetes más conocidos son los basados en **Debian**, con la extensión **.deb** precedido del nombre del archivo y el número de versión; y los basados en **Red Hat**, con la extensión **.rpm** (**Redhat Package Manager**) precedido por el nombre y un identificador acerca del procesador donde fueron compilados los programas incluidos en el paquete.

Clasificación de los principales gestores de paquetes:

	Formato <i>.deb</i> (Debian, Ubuntu)	Formato RPM (Red Hat)
Modo Línea de Órdenes	dpkg; apt-get*; aptitude* (interfaz texto para apt)	rpm; YUM; apt-get; aptitude (interfaz texto para apt)
Modo Gráfico	dselect; Synaptic*; Adept (basado en apt-get); Kpackage (parte de kdeadmin)	Synaptic; pup, pirut y yumex (basados en YUM); Adept (basado en apt-get); gpk-application (parte de gnome-packagekit); ; Kpackage (parte de kdeadmin)

Dentro de los gestores de paquetes en modo línea de órdenes, se distinguen dos niveles:

- **alto nivel:** como **apt-get** y **YUM**, que a su vez suelen proporcionar interfaces gráficas como PackageKit y gnome-packageKit sobre YUM. Suelen ser más interactivos y pueden realizar búsquedas, actualizaciones automáticas, incluyendo análisis de dependencias;
- **bajo nivel:** como **dpkg** y **rpm**. Suelen ser más precisos y potentes.

5.2 APT y YUM

APT (Advanced Packaging Tool) fue desarrollado por Debian Linux y modificado para usarlo también con paquetes RPM. APT y su interfaz gráfico Synaptic son fáciles de utilizar.

Sin embargo, hay desarrolladores que prefieren **YUM** (Yellow dog Updater Modified) para gestionar paquetes RPM, ya que parece que APT contiene código innecesario que realmente se utiliza para los paquetes .deb. YUM proporciona una interfaz para la línea de órdenes fácil y cómodo de usar.

Algunas órdenes útiles de YUM:

- **yum --help | more:** para obtener una lista completa de las órdenes y opciones disponibles de YUM;
- **yum list:** lista los paquetes disponibles en los repositorios para su instalación;

- **yum list installed:** lista los paquetes actualmente instalados;
- **yum list updates:** muestra todos los paquetes con actualizaciones disponibles en los repositorios para su instalación;
- **yum install <nombre_paquete>:** instala el paquete nombre_paquete;
- **yum update:** se actualizan todos los paquetes instalados;
- **yum remove <nombre_paquete>:** elimina el paquete nombre_paquete y los paquetes que dependen de él.

Actividad 2.7. Trabajo con el gestor de paquetes YUM

Encuentra los archivos de configuración de YUM y explora las distintas órdenes disponibles en YUM ejecutándolas. En concreto, lista todos los paquetes instalados y disponibles, elimina el paquete instalado que te indique el profesor de prácticas, y a continuación vuelve a instalar el mismo paquete haciendo uso de los paquetes que se encuentran disponibles en /fenix/depar/lsi/so/paquetes. Para obtener acceso a este directorio del sistema de archivos anfitrión ejecute la siguiente orden de montaje una vez lanzado el sistema operativo User Mode Linux (UML):

```
#> mount none /<directorio-punto-montaje> -t hostfs -o /fenix/depar/lsi/so/paquetes
```

5.3 RPM

RPM descende del primer software de gestión de paquetes Linux, comprueba dependencias e incluye opciones como la verificación de la revisión y las firmas de seguridad de privacidad GNU (los paquetes se distribuyen libres de virus). RPM utiliza el directorio /var/lib/rpm.

A continuación una clasificación de tipos de operaciones en el gestor RPM y órdenes más comunes:

Tipo de función	Órdenes más comunes	Descripción
Instalación de nuevos paquetes	rpm -i <nombre>	No muestra mensaje si se ha instalado correctamente.
Eliminación de paquetes instalados	rpm -e <nombre>	No muestra mensaje si se ha eliminado correctamente.
Actualización de paquetes instalados	rpm -U <nombre>	Con -U incluye la eliminación automática de la versión del paquete previamente instalada.
	rpm -F <nombre/servidor>	Se busca el paquete en el servidor designado en Internet.
Obtención de información sobre paquetes software	rpm -qa grep <nombre> sort	Busca paquetes instalados por su nombre o parte de este.
	rpm -qi <nombre>	Muestra información precisa del paquete instalado
Verificación e integridad de la instalación	rpm -V <nombre>	Verifica la instalación de un paquete. Si se ha realizado correctamente, no se produce salida.

Actividad 2.8. Trabajo con el gestor de paquetes rpm

En primer lugar deseamos mostrar cierta metainformación acerca de uno o más paquetes ya instalados. Para ello, debes utilizar la orden rpm con las opciones adecuadas. Utiliza el manual en línea si no sabes ya las opciones que debes utilizar.

1. Muestra la información general (nombre, versión, arquitectura, grupo, descripción, etc.) y lista los archivos que contiene un paquete ya instalado haciendo uso de la orden rpm y un único conjunto de opciones.

```
$ rpm -qli <paquete>
```

2. Ídem que el anterior pero mostrando únicamente los archivos de configuración que contiene el paquete.

```
$ rpm -q -c -i <paquete>
```

3. Escribe una orden que muestre los paquetes requeridos por un paquete determinado que se encuentre instalado en el sistema. Escriba la orden que devuelva el mismo resultado pero para un paquete no instalado en el sistema.

```
$ rpm -q --whatrequires <paquete>
```

4. Instala el paquete quota que encontrarás en el directorio de software de la asignatura (directorio que ya has montado en la Actividad 2.7).

```
$ rpm -i /fenix/depar/lisi/so/paquetes/quota-3.17-13.fc14.i686.rpm
```

5. Instala y desinstala el paquete sysstat mostrando en pantalla también la máxima información posible acerca del propio proceso de eliminación del paquete.

Para instalarlo, mostrando la mayor información posible:

```
$ rpm -ivh /fenix/depar/lisi/so/paquetes/sysstat-9.0.6-3.fc13.i686.rpm
```

Para desinstalarlo, mostrando la mayor información posible:

```
$ rpm -e --nodeps -v sysstat
```

6. Administración de cuotas

Mediante un sistema de **cuotas** podemos limitar el número de recursos de un SA que puede utilizar un usuario, como los bloques de disco y los inodos. Para poder trabajar con ello, es necesario tener instalado el paquete **quota**, que no suele venir por defecto.

El sistema de cuotas establece dos tipos de límites y se pueden establecer para usuarios y/o grupos y para bloques y/o inodos:

- **Límite hard:** el usuario no puede sobrepasar el número de bloques ni número de inodos, y una vez sobrepasados, no puede crear unos nuevos ni ampliar los ya creados;
- **Límite soft:** es un número de recursos inferior al hard, y puede sobrepasarse durante cierto tiempo (periodo de gracia) que es estipulado por el administrador, sin superar el límite hard. Una vez transcurre este tiempo, actúa como un límite hard.

Algunas órdenes útiles para el trabajo con cuotas son:

- **quota nombre_usuario:** para la asignación de cuotas para un usuario;
- **repquota <SA>:** estadísticas de las cuotas para todos los usuarios.

Actividad 2.9. Sistema de cuotas para el sistema de archivos tipo ext3

En esta actividad se van a presentar los pasos que necesitas llevar a cabo para establecer el sistema de cuotas de disco en Linux. El objetivo será activar el sistema de cuotas sobre el sistema de archivos tipo ext3 que has creado con anterioridad.

1. Editar el archivo `/etc/fstab` y activar el sistema de cuotas de usuario para el SA tipo ext3. Busca cómo se especifica esta opción en el manual en línea. Una ayuda para la búsqueda es que la realices sobre la orden `mount` y recuerdes que las opciones de montaje vienen especificadas en los apartados: **FILESYSTEM INDEPENDENT MOUNT OPTIONS** y **FILESYSTEM SPECIFIC MOUNT OPTIONS**.

```
$ /dev/loop0 /mnt/SA_ext3 ext3 auto,ro,usrquota 0 0
```

2. Montar de nuevo el SA en el espacio de nombres para que se active la opción previamente establecida. Usa la siguiente orden:

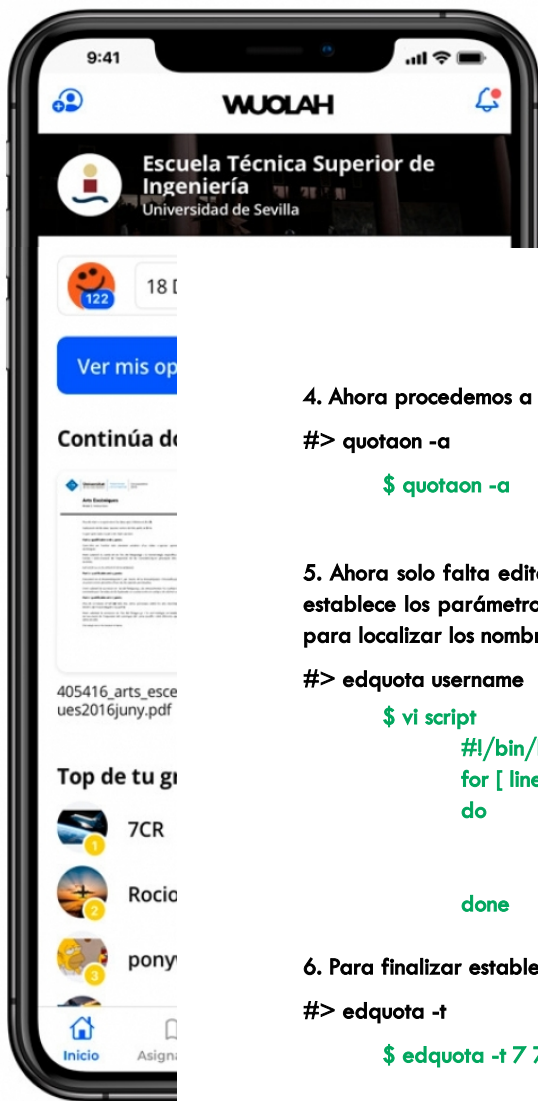
```
#> mount -o remount <directorio_punto_de_montaje>
```

```
$ mount -o remount /mnt/SA_ext3
```

3. Crear el archivo que permite llevar el control de cuotas de usuario para el SA. El nombre de este archivo es `aquota.user`. Para ello utiliza la siguiente orden:

```
#> quotacheck -nm <directorio_punto_de_montaje>
```

```
$ quotacheck -nm /mnt/SA_ext3
```



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



4. Ahora procedemos a activar el sistema de control de cuotas de usuario. Para ello ejecuta la orden:

```
#> quotaon -a
```

```
$ quotaon -a
```

5. Ahora solo falta editar la cuota para cada usuario del sistema mediante la siguiente orden. En este caso, establece los parámetros para cada usuario existente. Puede ser buena idea utilizar el archivo `/etc/passwd` para localizar los nombres.

```
#> edquota username
```

```
$ vi script
#!/bin/bash
for [ linea in `cat /etc/passwd` ]
do
    usuario = `cut --d : -f1`
    setquota -u $usuario
done
```

6. Para finalizar estableceremos el periodo de gracia para el límite soft.

```
#> edquota -t
```

```
$ edquota -t 7 7
```

Actividad 2.10. Establecer límites sobre recursos de un SA

Establece los límites de bloques e i-nodos para un par de usuarios del sistema UML sobre el que trabajas en el laboratorio.

```
$ setquota -u usuario -b nombre 5 9 5 9
```

7. Preguntas de repaso

1. ¿Cómo podemos ver la lista de tipos de particiones con los que nos permite trabajar nuestro SO?

```
$ /sbin/sfdisk -T
```

2. ¿Cómo podemos ver cuál es el dispositivo que acabamos de insertar en el sistema?

```
$ lsblk
```

```
ó $ fdisk -l
```

3. ¿Cómo podemos ver la lista de dispositivos que están montados?

```
$ mount
```

4. ¿Cómo podríamos saber los dispositivos loop que están usándose?

\$ fdisk -l | grep loop muestra los dispositivos loop que hay, pero no dice si están usándose o no
\$ losetup -a

5. Si no sabemos qué dispositivo loop está libre, ¿cómo podemos asignar un fichero al primer dispositivo loop que esté libre?

\$ losetup -f archivo_a_asignar (la opción -f muestra los loop libres)

6. ¿Con qué orden se determina el número de bloque de comienzo y de fin de una partición?

\$ fdisk

7. ¿Con qué orden se determina el tipo de un Sistema de Archivos?

\$ mke2fs

8. [Como usuario root] Supongamos que hemos ejecutado con éxito la orden: losetup /dev/loop0 /archivo1

Asocia el archivo de dispositivo /dev/loop0 con /archivo1

9. Elija cuál de las siguientes afirmaciones es la correcta:

1. ¿Con qué orden se asigna el tipo de un Sistema de Archivos (SA)?

\$ mke2fs

2. ¿Qué orden se asigna a un número de montajes tras los cuales se ejecutará la utilidad de chequeo de consistencia de un SA?

\$ tune2fs

3. ¿Con qué orden se asigna el tamaño de un inodo?

\$ tune2fs

4. ¿Con qué orden se asigna el directorio que asociamos a un Sistema de Archivos?

\$ mount

5. ¿Con qué orden se asigna el número de bloque de comienzo y de fin de una partición?

\$ fdisk

10. /etc/mtab se diferencia de /etc/fstab en que:

/etc/mtab muestra todos los sistemas de archivos ya montados, mientras que /etc/fstab muestra los que se podrían montar