

WUOLAH



David97

www.wuolah.com/student/David97



11661

SOModllesion2.pdf

Sesión 2 - Módulo I - SO



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 SA

Apuntes

1 Introducción

Tipos de archivos en los sistemas UNIX:

- **Archivos regulares.** Archivos de programa y datos.
- **Directorios.** Archivos diseñados para soportar la estructura jerárquica de organización de información en un SA. Esta estructura implementa el *espacio de nombres de archivo*.
- **Enlaces simbólicos.** Archivos que permiten referenciar a otros archivos desde distintas ubicaciones en el espacio de nombres (distintos directorios).
- **Archivos especiales de dispositivo.** Representan dispositivos, siendo una abstracción de estos. UNIX/Linux distinguen dos subtipos: orientados a dispositivos de **bloques** y orientados a dispositivos de **caracteres**.
- Archivos que permiten comunicar procesos:
 - **FIFO.**
 - **Socket.**

MBR: *Master Boot Record* → registro de arranque maestro

En el MBR está la información clave del disco. Es el primer sector de un dispositivo de almacenamiento de datos.

1. Sistema de ficheros
2. Administración de dispositivos de almacenamiento secundario y SAs.
 - 2.1. Formateo a bajo nivel
 - 2.2. Particiones (fdisk)
 - 2.3. Formateo a alto nivel (lógico) → asigna sistema de archivos a una partición (mkfs*)
 - 2.4. Personalización de metadatos de un SA (tune2fs)
 - 2.5. Reparación de inconsistencias (fsck)
 - 2.6. Montaje (mount)
3. Instalación de software

2 Partición de dispositivos de almacenamiento secundario

Las **particiones** dentro del dispositivo físico permiten alojar un sistema de archivos concreto. Los dispositivos de almacenamiento (y sus particiones) están constituidos por **sectores**, que forman los **discos lógicos**. Un sector es la mínima unidad de almacenamiento de información.

Al crear una partición es necesario asignarle una etiqueta que indica el tipo de SA que va a alojar tras el formateo. Los tipos de particiones (código) se pueden consultar con `/sbin/sfdisk -T`.

Una **partición primaria** de disco es cada una de las cuatro particiones que se pueden establecer en un dispositivo. Estas pueden subdividirse en **particiones lógicas**. La partición primaria utilizada para almacenar particiones lógicas se denomina **partición extendida**.

2.1 Número de particiones en dispositivo

Dispositivo de arranque (*Boot drive*)

Es el dispositivo que utiliza la BIOS para cargar en memoria el programa SO o un programa cargador de SO. Para establecer este arranque desde el dispositivo a particionar hacen falta:

- Una partición primaria.
- Una o más particiones **swap**.
- Ninguna o tantas particiones primarias y lógicas como se deseen (dentro de los límites)

Dispositivo de almacenamiento secundario que no se utilice como arranque

- Una o más particiones primarias o lógicas.
- Ninguna o varias particiones **swap**.

2.2 Directorios de primer nivel en el estándar FHS en una partición

/home	Almacena los directorios de inicio de todos los usuarios. Por tanto, podemos establecer una partición independiente y limitar de esta forma el espacio de almacenamiento permitido para los usuarios.
/usr	Almacena la mayoría de los ejecutables binarios del sistema, así como sus demás archivos adicionales, incluyendo los de documentación. Las fuentes del kernel también se almacenan aquí.
/var	Contiene los directorios de SPOOL, como por ejemplo los de impresión y email. Normalmente todos los computadores en los que Linux actúa como servidor asignan a /var una partición independiente de la partición raíz /.

- **mknod**: crea un bloque o archivo especial de caracteres
 - **b** → crear un archivo especial de bloque (con búfer)
 - **7** → tipo de archivo
 - **0** → n.º de archivos de este tipo creados
- **dd**: copia un fichero, convirtiendo y formateando de acuerdo a los operandos
 - **if=FILE** → lee de FILE en lugar del *stdin*
 - **of=FILE** → escribe en FILE en lugar de *stdout*
 - **bs=B** → lee y escribe hasta B bytes a la vez (default: 512 B); anula *ibs* y *obs*
 - **count=N** → copia sólo N bloques de entrada
- **losetup**: Configuración y control de dispositivos loop

3 Asignación de un SA a una partición (formateo lógico)

Tipos de particiones que normalmente se utilizan en Linux (aparte de *swap*):

- **ext2**. SA de disco de alto rendimiento para DD, memorias flash y medios extraíbles.
- **ext3**. Versión del **ext2** que incluye “registro por diario” (*jounaling*), mecanismo por el cual un sistema informático puede implementar transacciones.
- **ext4**. Estándar de facto actual de Linux. Tiene estructuras similares a **ext3** pero, además, presenta mejoras en cuanto a extensiones y *allocate-on-flush*

mke2fs: Crea un SA de tipo **ext2**, **ext3** o **ext4**

- **-t** → tipo de archivo a asignar
- **-L** → etiqueta a establecer

4 Ajuste de parámetros configurables de un SA y comprobación de errores

Con la orden `tune2fs` se pueden ajustar determinados parámetros a los nuevos SAs.

- `-l <dispositivo>` → muestra el contenido del superbloque del SA.
- `-c max-mount-counts <dispositivos>` → n.º máximo de montajes a realizar sin comprobación de consistencia del SA.
- `-L label <dispositivos>` → pone una etiqueta al SA.

Linux automatiza el proceso de comprobación de la consistencia del SA. Cuando es el administrador del sistema de archivos quien tiene que ejecutar manualmente las comprobaciones y reparar las posibles inconsistencias, Linux proporciona la herramienta **fsck**, que actúa sobre las estructuras de metainformación del SA (no sobre los propios datos del archivo).

5 Montaje y desmontaje de Sistemas de Archivos

Una vez se dispones de los SA queda ponerlos a disposición de los usuarios haciendo que sean accesibles dentro de la jerarquía de directorios mediante la orden **mount**. Cuando no es necesario acceder a la información del SA montado se puede desmontar, de forma que deja de estar disponible (**umount**); no es posible desmontarlo si está siendo utilizado (*busy*).

6 Administración de software

Un **paquete software** es un **archivo** que contiene un conjunto de programas complementarios y que, en la mayoría de los casos, tienen dependencias con otros paquetes. Los paquetes pueden incluir archivos con programas en código binario o bien código fuente junto con las instrucciones necesarias para generar el código binario. En Debian la extensión típica es “.deb”, mientras que en Red Hat los paquetes se identifican por “.rpm”.

Los **gestores de paquetes** se encargan de administrar paquetes software. Los principales son:

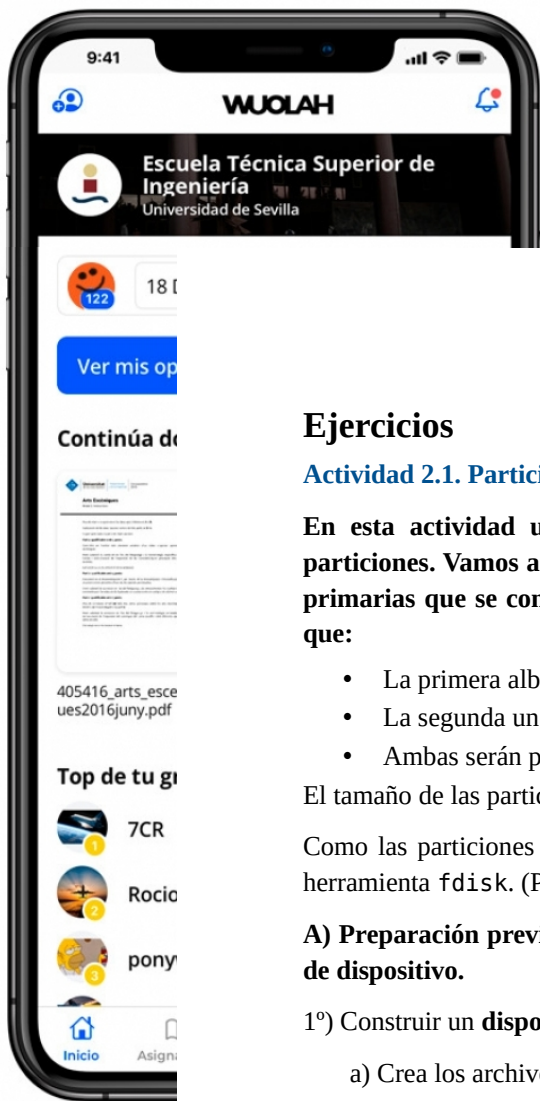
- Alto nivel (más interactivos): apt-get, YUM
- Bajo nivel (más precisos y potentes): dpkg, rpm,

7 Administración de cuotas

Las **cuotas** permiten **limitar el número de recursos** de un SA que va a poder utilizar un usuario. El sistema de cuotas permite establecer límites sobre grupos.

El sistema de cuotas establece dos tipos de límites para restringir el uso de bloques e i-nodos, los cuales se aplican sobre usuarios y/o grupos.

- **Límite *hard***. El usuario no puede sobrepasarlo. Si lo sobrepasase el sistema no le permitiría usar más recursos (bloques e i-nodos).
- **Límite *soft***. Debe configurarse como un n.º de recursos inferior al límite *hard* y se puede sobrepasar durante cierto tiempo (estipulado por el administrador), denominado *período de gracia*, pero sin sobrepasar el límite *hard*. Si transcurre el período de gracia el sistema de cuotas actúa como si se hubiera superado el límite *hard*.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



David Carrasco Chicharro

Ejercicios

Actividad 2.1. Partición de un dispositivo: “USB pen drive” o “memory stick”

En esta actividad utilizaremos un dispositivo USB de tipo “pen drive” para establecer particiones. Vamos a crear una tabla de particiones en las que se van a definir dos particiones primarias que se configuraran para albergar dos sistemas de archivos tipo linux, de manera que:

- La primera albergará un SA **ext3**
- La segunda un **ext4**.
- Ambas serán particiones tipo Linux **0x83**.

El tamaño de las particiones es libre, pero **por lo menos** deben tener **512 MB**. ← **está mal**

Como las particiones que vamos a hacer no van a ser excesivamente grandes vamos a utilizar la herramienta **fdisk**. (Para tamaños mayores, usar la orden **parted** de GNU) .

A) Preparación previa a la partición de un dispositivo simulado mediante un archivo especial de dispositivo.

1º) Construir un **dispositivo simulado** mediante un archivo **/dev/loop?**

a) Crea los archivos **/dev/loop0** y **/dev/loop1**, si no se encuentran en el sistema:

```
# ls /dev/loop*    ← no existen
# mknod /dev/loop0 b 7 0
# mknod /dev/loop1 b 7 1
```

b) **Crea un archivo** de 20 MB y otro de 30 MB en tu sistema de archivos:

```
# dd if=/dev/zero of=/root/archivo_SA20 bs=2k count=10000
10000+0 records in
10000+0 records out
20480000 bytes (20 MB) copied, 0.272752 s, 75.1 MB/s
# dd if=/dev/zero of=/root/archivo_SA30 bs=3k count=10000
10000+0 records in
10000+0 records out
30720000 bytes (31 MB) copied, 0.291487 s, 105 MB/s
```

c) Ahora vamos a **asociar un archivo de dispositivo loop** (a) a cada uno de los archivos creados (b). De esta forma el “disco virtual” que representa el archivo pasará a estar asociado al archivo de dispositivo **/dev/loop0** y **/dev/loop1**:

```
# losetup /dev/loop0 /root/archivo_SA20
# losetup /dev/loop1 /root/archivo_SA30
```

d) Comprobamos la configuración de los “discos virtuales”:

```
# fdisk -l /dev/loop0 /dev/loop1
Disk /dev/loop0: 20 MB, 20480000 bytes
255 heads, 63 sectors/track, 2 cylinders, total 40000 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/loop0 doesn't contain a valid partition table
```

```

Disk /dev/loop1: 30 MB, 30720000 bytes
255 heads, 63 sectors/track, 3 cylinders, total 60000 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/loop1 doesn't contain a valid partition table

```

B) Preparación previa a la partición de un pen drive

```
cat /proc/mounts
```

```
umount
```

CREAR TABLA DE PARTICIONES

- **Op. 1:** En los discos virtuales /dev/loop0 y /dev/loop1.
- **Op. 2:** En el pen drive

Opción 1

```
# fdisk /dev/loop0
```

```
Command: n
```

```
Command action: (primary partition 1-4) p
```

```
Partition number (1-4, default 1): 1
```

```
First sector (2048-39999, default 2048): 2048 (o dejar vacío)
```

```
Last sector, +sectors or +size{K,M,G} (2048-59999, default 59999): +20M
```

(si no deja, porque diga “**Value out of range**”, bajar el tamaño a +18M aprox. Para evitar esto, reservar más espacio del necesario).

```
Command: w
```

```
# fdisk /dev/loop1
```

```
Command: n
```

```
Command action: (primary partition 1-4) p
```

```
Partition number (1-4, default 1): 1
```

```
First sector (2048-59999, default 2048): 2048
```

```
Last sector, +sectors or +size{K,M,G} (2560-39999, default 39999): +30M
```

```
Command: w
```

Actividad 2.2. Creación de sistemas de archivos

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.

```
# mke2fs -L LABEL_ext3 -t ext3 /dev/loop0
```

```
# mke2fs -L LABEL_ext4 -t ext4 /dev/loop1
```

Ajuste de parámetros configurables de un SA y comprobación de errores:

```
# tune2fs -l /dev/loop0
```

```
# tune2fs -l /dev/loop1
```

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 el *flag* `-c`, que indica el `max-mount-counts`, con el valor 0 ó -1.

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

`-u {username|<UID>} -r <nº bloques>`

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:

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

```
# mkdir /mnt/SA_ext3
# mount -r -L LABEL_ext3 /mnt/SA_ext3
```

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

```
# mkdir /mnt/LABEL_ext4
# mount -L LABEL_ext4 -o sync,dirsync /mnt/LABEL_ext4
```

Actividad 2.5 Automontaje de Sistemas de Archivos

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

```
LABEL=LABEL_ext3 /mnt/SA_ext3      ext3      ro,auto      0 0
LABEL=LABEL_ext4 /mnt/LABEL_ext4    ext4      dirsync,sync,auto 0 0
```

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/lisi/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 /<dir-punto-montaje> -t hostfs -o /fenix/depar/lisi/so/paquetes
```


En mi caso, que estoy trabajando con mi ordenador personal y tengo descargados dichos paquetes, la orden utilizada es (previa creación del usuario u1 y el directorio *paquetes*):

```
# mount none /home/u1/paquetes -t hostfs -o /home/david/Universidad/S0/paquetes

# yum list installed → lista los paquetes instalados
# yum remove libnl.i686 → borra el paquete libnl.i685
```

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.

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.

Consultamos en primer lugar los paquetes instalados con: `# rpm -qa`

```
# rpm -qi passwd-0.78-1.fc14.i686
```

```
Name       : passwd                               Relocations: (not relocatable)
Version    : 0.78                                Vendor: Fedora Project
Release    : 1.fc14                              Build Date: Fri Jul 16 06:46:39 2010
Install Date: Wed Nov  3 05:12:38 2010          Build Host: x86-12.phx2.fedoraproject.org
Group      : System Environment/Base             Source RPM: passwd-0.78-1.fc14.src.rpm
Size       : 378157                               License: BSD or GPLv2+
Signature  : RSA/SHA256, Wed Jul 28 16:48:59 2010, Key ID 421caddb97a1071f
Packager   : Fedora Project
URL        : http://fedorahosted.org/passwd
Summary    : An utility for setting or changing passwords using PAM
Description:
This package contains a system utility (passwd) which sets or changes passwords,
using PAM (Pluggable Authentication Modules) library.
```

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

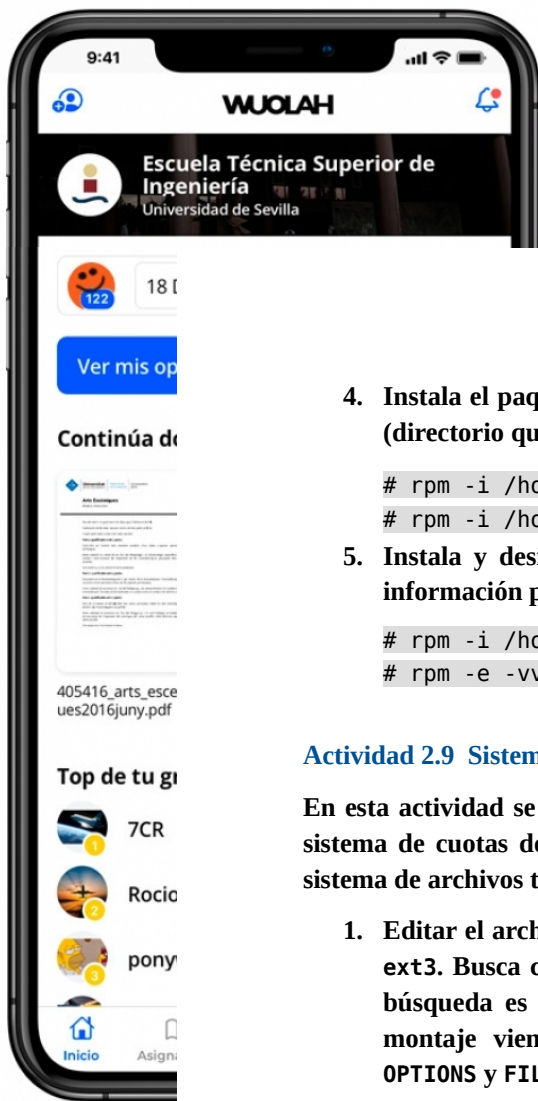
```
# rpm -qc passwd-0.78-1.fc14.i686
/etc/pam.d/passwd
```

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 -qR passwd-0.78-1.fc14.i686
/etc/pam.d/system-auth
audit-libs >= 1.0.14
config(passwd) = 0.78-1.fc14
libaudit.so.1
libc.so.6
...
rpmLib(PayloadIsXz) <= 5.2-1
```

Para un paquete que **no** se encuentra instalado, los paquetes requeridos se pueden listar con:

```
# rpm -qR passwd-0.78-1.fc14.i686
```



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



David Carrasco Chicharro

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 /home/u1/paquetes/libnl-1.1-12.fc14.i686.rpm ← (necesario para quota)
# rpm -i /home/u1/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.

```
# rpm -i /home/u1/paquetes/sysstat-9.0.6-3.fc13.i686.rpm
# rpm -e -vv sysstat-9.0.6-3.fc13.i686
```

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**.

```
LABEL=LABEL_ext3    /mnt/SA_ext3    ext3    usrquota    0 0
```

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

```
# mount -o remount /mnt/SA_ext3
13025.410000] EXT3-fs (loop0): using internal journal
```

3. Crear el archivo que permite llevar el control de cuotas de usuario para el SA. El nombre de este archivo es `aquota.user`.

```
# quotacheck -nm /mnt/SA_ext3
```

-n → Si los archivos de cuota se corrompen, es posible que existan entradas duplicadas para un único ID de usuario o grupo. Normalmente, en este caso, `quotacheck` sale o pide al usuario que introduzca sus datos. Cuando esta opción está activada, siempre se utiliza la primera entrada encontrada.

-m → no trata de remontar el SA de sólo lectura

4. Ahora procedemos a activar el sistema de control de cuotas de usuario.

```
# 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 u1
```

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

```
# edquota -t
```

Actividad 2.10 Establecer límites sobre recursos de un SA

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

Uso: `# setquota -u <user> softlimit_block hardlimit_block \`
`softlimit_inode hardlimit_inode -a <filesystem>`

```
# setquota -u u1 100 200 10 15 -a /dev/loop0
```

Con `# edquota u1` comprobamos que se ha realizado correctamente:

Disk quotas for user u1 (uid 500):

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/loop0	0	100	200	0	10	15