

# Examen prácticas 1-3

- comando
- formato
- importante
- directorio

## • Sesión 1:

`$whoami` → pedimos el nombre del usuario actual  
`$su` → solicitamos cambio a modo root

Los usuarios disponen de `username`, `UID` (identificador de usuario) y el grupo o grupos a los que pertenece (`gid`). Todo usuario tiene un grupo principal que está en el archivo `/etc/passwd`. Puede tener grupos suplementarios en `/etc/group`.

Para crear nuevas cuentas usamos `adduser` o `useradd`. Se crea el usuario, su grupo principal y las entradas correspondientes en `/etc/passwd`, `/etc/shadow` y `/etc/group`. Se crea también el directorio de inicio (suele ser en `/home`).

`useradd -d home_dir -u UID -g GROUP username`

`usermod`: modifica una cuenta existente

`userdel`: borra una cuenta (no suele borrar `home`)

`newusers`: crea cuentas utilizando datos de un archivo de texto con formato `/etc/passwd`

`passwd <nombre usuario>`: asignar o cambiar contraseñas

Para establecer restricciones de envejecimiento de las contraseñas se hace en `/etc/shadow`

↳ `--mindays`: min número de días para cambiar la contraseña

↳ `--maxdays`: max número de días para cambiar la contraseña

Se usa la orden `chage`, por ejemplo: `chage --mindays num.días usuario`

Los grupos tienen `groupname`, `GID`, y el archivo de configuración `/etc/group` (`nombre:x:gid:lista de usuarios`)

`usermod -g` modifica el grupo del usuario

`id` devuelve el `UID` y el `GID`

`/bin`: programas de utilidad fundamentales para ser usados por cualquier usuario

`/boot`: archivos fundamentales para el programa Boot loader

`/dev`: archivos especiales de dispositivos

`/etc`: archivos de configuración de sistemas

`/home`: directorio de inicio de todos los usuarios menos root

`/mnt`: punto de montaje para sistemas de archivos montados temporalmente

`/proc`: hace de interfaz con el núcleo y los procesos

`/tmp`: archivos temporales que no se suelen mantener al reiniciar

`/proc/filesystem`: archivos soportados por el kernel

## • Sesión 2

Para hacer la partición, se usa un dispositivo simulado `/dev/loop`?

Para crearlos hacemos: `mknod /dev/loop0 b 7 0`  
`mknod /dev/loop1 b 7 1` } creamos dos `/dev/loop`

`dd if=/dev/zero of=/root/archivo-SA20 bs=2k count=10000`  
`dd if=/dev/zero of=/root/archivo-SA30 bs=3k count=10000` } creamos un archivo de 20MB y otro de 30MB

`losetup /dev/loop0 /root/archivo-SA20`  
`losetup /dev/loop1 /root/archivo-SA30` } Asociamos los archivos de dispositivo loop a los archivos que acabamos de crear

`fdisk -l /dev/loop0 /dev/loop1` ⇒ comprobamos la configuración

```
con: fdisk /dev/loop0
m
n
p
enter
enter
enter
w
```

creamos la tabla de particiones

`fdisk -l /dev/loop0` ⇒ comprobamos

`mke2fs -t ext? /dev/loop?`

`mke2fs -L LABEL-ext? /dev/loop?` → le damos un label a la partición

Para montar sistemas de archivos cambiamos `/etc/fstab` con la orden `vi /etc/fstab`

`/etc/fstab` tiene el siguiente formato:

<file system>	<mount point>	<type>	<options>	<dump>	<pass>
número que identifica al archivo	directorio que actúa como punto de montaje Ej: /mnt	tipo de sistema de archivos Ej: tmpfs, proc	opciones para el proceso de ensamblado	frecuencia copia seguridad	orden comprobaciones sobre los SAs

rw: lectura, escritura  
ro: sólo lectura  
auto/no-auto: montar autom. o no  
usrquota: cuota usuario  
defaults: rw, suid, dev, exec, auto, nouser, async.  
uidgid: propietario y grupo propietario del archivo

Añadimos la línea: `LABEL=LABEL-ext? /mnt ext? ro 0 0`

Luego hacemos `mount -L LABEL-ext?`

También se puede hacer: `mount -o options /dev/loop0 /mnt`

rpm y yum son gestores de paquetes softwares. Estos últimos son archivos que contienen conjuntos de programas complementarios.

`yum list`: lista de paquetes disponibles

`yum list installed`: paquetes instalados

`yum remove paquete`: elimina el paquete especificado

`yum install paquete`: instala el paquete especificado

`mount none /mnt -t hostfs -o /tmp/paquetes` ⇒ para montar los paquetes

`rpm -i nombre`: instala paquete

`rpm -e nombre`: elimina paquete

`rpm -qi nombre`: muestra info del paquete instalado

Las cuotas sirven para limitar el uso de disco por parte de los usuarios del sistema.

Hay que tener instalado el paquete `quota`.

Hay dos tipos de límites:

↳ Hard: no se puede sobrepasar, si se llega, el usuario no podrá crear más archivos

↳ Soft: se puede sobrepasar durante cierto tiempo. Pasado este tiempo, actuará como un límite hard.

Para activar el sistema de cuotas, editamos `/etc/fstab` y añadimos `usrquota` al usuario correspondiente

Volvemos a montar el dispositivo: `mount -o remount /mnt`

creamos el archivo que permite llevar el control de cuotas de usuarios para el SA: `quotacheck -nm /mnt/`

Activamos el sistema de control de cuotas: `quotaon -a`

Editamos la cuota para cada usuario: `edquota username`

• Session 3

uptime muestra la siguiente info: <hora actual><tiempo que lleva el sistema en marcha><n usuarios conectados>  
<carga media del sistema en los últimos 1, 5, 15 minutos>

w muestra la misma información y lo que está haciendo cada usuario conectado

time muestra el tiempo de ejecución de un programa:  $\langle \text{tiempo real} \rangle \langle \text{tiempo modo usuario} \rangle \langle \text{tiempo modo supervisor} \rangle$

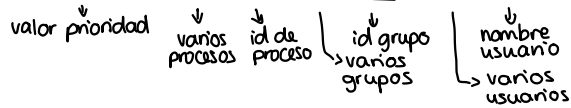
↳ poniendola delante de otra orden, mide su tiempo de ejecución

nice permite establecer valores de prioridad a los procesos (entre  $[-20, 19]$ )

↳ un valor negativo aumenta la posibilidad de ejecución de un proceso y sólo lo puede hacer un root.

renice permite hacer lo mismo para varios procesos o usuarios:

renice prioridad [-p] pid [-g] pgrp [-u] usuario



**ps-tree** muestra un árbol de procesos de ejecución

ps muestra información de los procesos en ejecución

↳ USER	PID	TTY
--------	-----	-----

usuario que lanzó el programa	identificador proceso	terminal asociada al proceso
root	1	1
root	2	1
root	3	1
root	4	1
root	5	1
root	6	1
root	7	1
root	8	1
root	9	1
root	10	1
root	11	1
root	12	1
root	13	1
root	14	1
root	15	1
root	16	1
root	17	1
root	18	1
root	19	1
root	20	1
root	21	1
root	22	1
root	23	1
root	24	1
root	25	1
root	26	1
root	27	1
root	28	1
root	29	1
root	30	1
root	31	1
root	32	1
root	33	1
root	34	1
root	35	1
root	36	1
root	37	1
root	38	1
root	39	1
root	40	1
root	41	1
root	42	1
root	43	1
root	44	1
root	45	1
root	46	1
root	47	1
root	48	1
root	49	1
root	50	1
root	51	1
root	52	1
root	53	1
root	54	1
root	55	1
root	56	1
root	57	1
root	58	1
root	59	1
root	60	1
root	61	1
root	62	1
root	63	1
root	64	1
root	65	1
root	66	1
root	67	1
root	68	1
root	69	1
root	70	1
root	71	1
root	72	1
root	73	1
root	74	1
root	75	1
root	76	1
root	77	1
root	78	1
root	79	1
root	80	1
root	81	1
root	82	1
root	83	1
root	84	1
root	85	1
root	86	1
root	87	1
root	88	1
root	89	1
root	90	1
root	91	1
root	92	1
root	93	1
root	94	1
root	95	1
root	96	1
root	97	1
root	98	1
root	99	1
root	100	1

si hacemos **ps-ef** se muestra más información

si en el campo TTY hay una ?, significa que no hay una terminal asociada al proceso

**top** muestra de forma continuada la actividad de un proceso en tiempo real

↳ SHR muestra la cantidad de memoria compartida

↳ si escribimos:  $r_i$ , podemos cambiar la prioridad de un proceso

$N$ , ordenarlos por PID

n, cambiar el número de procesos que se muestran

q, para salir

mostrar muestra las estadísticas del procesador junto con la media global de los datos mostrados.

mpstat intervalo número

- ↳ cada cuantos segundos debe mostrar los datos

**free** consume menos recursos (CPU y memoria) que la orden top. Es una orden muy ligera que sirve para visualizar el uso actual de memoria (RAM y swap)

**vmstat** muestra información de memoria, procesos, E/S y CPU.

vmstat segundos iteraciones

Columns:

- us: programas de usuario
- sy: tareas del sistema
- id: no haciendo nada
- r: cuantos procesos hay en cola de ejecución
- wa: intercambio

vmstat -S M 2 10 muestra 10 informes, cada 2 segundos, en megabytes.

**ls** muestra los archivos guardados en un directorio

↳ -l: long listing format

↳ -n: numeric user and group IDs

↳ -la: doesn't ignore hidden entries

d está asociado a directorios, b a archivos de bloque, c a archivos de caracteres...

**ls -l** : sort alphabetically by directory

-t : sort by modification time

-u : " " access time

-c : " " ctime

} para usar long format hacemos por ejemplo:  
ls -u -lt

**df** permite visualizar, para cada SA montado, información sobre su capacidad de almacenamiento.

**du** permite ver todo el subárbol de la jerarquía que comienza en un directorio, y cuanto espacio en el disco gasta.

Los enlaces pueden ser a los nombres de archivos (enlaces simbólicos) o bien a sus metadatos (enlaces duros)

Para crear enlaces sobre un archivo existente hacemos: **ln archivo nombre enlace**

Para **montar los paquetes** hacemos:

**mkdir paquetitos**

**mount none /root/paquetitos -t hostfs -o /home/lauralazarro/.../SO/paquetes**

**cd paquetitos**

**rpm -i libn...**

**rpm -i tcp-wrappers...**

**rpm -i quota...**

# ejercicios

## Ejercicio 1:

- (a) `useradd -d /home/laura -u 312 examen1`  
`useradd -d /home/laura/otro -u 313 examen2`  
`cat /etc/passwd`
- (b) `groupadd muchos`  
`usermod -g muchos examen1`  
`usermod -g muchos examen2`  
`groups examen1`  
`groups examen2`
- (c) `/etc/fstab` muestra los archivos que se pueden montar en el sistema y `/etc/mtab` los ya montados.
- (d) `/proc/filesystem`

## Ejercicio 2:

- (a) `mknod /dev/loop0 b 7 0`  
`dd if=/dev/zero of=/root/archivo-examen bs=35k count=1000`  
`losetup /dev/loop0 /root/archivo-examen`
- (b) `fdisk /dev/loop0`  
m  
n  
p  
↓  
w  
`mke2fs -t ext 4 /dev/loop0`
- (c) `mount -o rw -o sync /dev/loop0 /mnt`

## Ejercicio 3:

- (a) En la primera línea aparece la hora actual, el tiempo que lleva en marcha el sistema, los usuarios conectados y la carga media en el último minuto, últimos 5 y últimos 15.  
`top -N` ordena por PID
- (b) `vmstat -s M 2 10`  
programas o procesos de usuarios
- (c) `ln archivo-examen enlace`
- (d) `ls -lt`

## más ejercicios

### Ejercicio 1:

(a) `adduser examen3 -d /Escritorio -u 801`  
`adduser examen4 -u 803`  
`cat /etc/passwd`

(b) `passwd -x 10 -w 2 examen3`  
`passwd -x 14 -w 4 examen4`

(c) `groupadd muchos`  
`usermod -g muchos examen3`  
`usermod -g muchos examen4`  
`id examen3`

- (d) `diratime` supone que el registro de tiempos para los directorios está activado  
`dirsync` supone que todas las actualizaciones de directorios se deben hacer de manera síncrona.

### Ejercicio 2:

(a) `mknod /dev/loop0 b 7 1`  
`dd if=/dev/zero of=/archivo-examen bs=15k count=1000`  
`losetup /dev/loop0 archivo-examen`

(b) `fdisk /dev/loop0`  
`m`  
`n`  
`p`  
`↓`  
`w`  
`mke2fs -t /ext3 /dev/loop0`

(c) `mount -o ro -o noexec /dev/loop0 /mnt`

### Ejercicio 3:

(a) El número de tareas que se están ejecutando, totales, hibernando, detenidas y zombie.  
`n`  
`5`

(b) `vmstat -s K 2 10`  
`free`: memoria libre  
`r`: procesos en cola de ejecución

(c) `df -h`

(d) `ls -u -lt`