

## Tarea para SI06

**Ejercicio 1.** Tras crear los usuarios y asignarlos a cada grupo, ejecutamos el comando `sudo cat /etc/passwd` y visualizamos el archivo con los usuarios del sistema. Al final del archivo encontramos los últimos usuarios que hemos creado.

```
alacreu:x:1000:1000:Alacreu,,,:/home/alacreu:/bin/bash
vboxadd:x:999:1::/var/run/vboxadd:/bin/false
alumno:x:1001:1001:alumno,,,:/home/alumno:/bin/bash
juana:x:1002:1002:,,,:/home/juana:/bin/bash
luis:x:1003:1003:,,,:/home/luis:/bin/bash
orena:x:1004:1003:,,,:/home/orena:/bin/bash
maria:x:1005:1004:,,,:/home/maria:/bin/bash
angel:x:1006:1004:,,,:/home/angel:/bin/bash
```

| Usuario | UID  | Grupo       | GID  |
|---------|------|-------------|------|
| juana   | 1002 | juana       | 1002 |
| luis    | 1003 | informatico | 1003 |
| orena   | 1004 | informatico | 1003 |
| maria   | 1005 | vendedor    | 1004 |
| angel   | 1006 | vendedor    | 1004 |

Iniciamos la sesión con el usuario juana y creamos los tres archivos



```
juana@SistemasUbuntu:~$ touch factura1
juana@SistemasUbuntu:~$ touch factura2
juana@SistemasUbuntu:~$ touch carta
juana@SistemasUbuntu:~$
```

accedemos como root y modificamos el grupo de juana a vendedor

```
juana@SistemasUbuntu:~$ su -  
Contraseña:  
root@SistemasUbuntu:~# usermod -g vendedor juana
```

movemos el archivo carta al home de luis

```
oot@SistemasUbuntu:~# mv /home/juana/carta /home/luis
```

modificamos el propietario del archivo carta

```
root@SistemasUbuntu:~# chown luis:informatico /home/luis/carta
```

cambiamos al usuario juana al grupo vendedor

```
root@SistemasUbuntu:~# chgrp -R vendedor /home/juana
```

eliminamos el grupo juana

```
root@SistemasUbuntu:~# groupdel juana
```

añadimos al usuario luis al grupo sudo

```
root@SistemasUbuntu:~# gpasswd -a luis sudo  
Añadiendo al usuario luis al grupo sudo
```

verificamos que se ha añadido correctamente

```
root@SistemasUbuntu:~# cat /etc/group | grep sudo  
sudo:x:27:alacreu,luis
```

## Ejercicio 2.

Introducimos el disco de windows en la maquina virtual.

Podemos ver que se encuentra montado en la siguiente carpeta: /media/juana

```
juana@SistemasUbuntu:~$ ls /media/juana/  
CCCOMA_X64FRE_ES-ES_DV9
```

El archivo de dispositivo es /dev/sr0.

Mostramos el listado de archivos en la raiz del cd

```
juana@SistemasUbuntu:~$ ls /media/juana/CCCOMA_X64FRE_ES-ES_DV9  
autorun.inf boot bootmgr bootmgr.efi efi setup.exe sources support
```

Por último mostramos el contenido del archivo autorun.inf

```
juana@SistemasUbuntu:~$ cat /media/juana/CCCOMA_X64FRE_ES-ES_DV9/autorun.inf  
[AutoRun.Amd64]  
open=setup.exe  
icon=setup.exe,0  
  
[AutoRun]  
open=sources\SetupError.exe x64  
icon=sources\SetupError.exe,0
```

### Ejercicio 3.

Mostramos la información de todas las particiones de los discos duros

```
root@SistemasUbuntu:~# fdisk -l /dev/sda
Disco /dev/sda: 100 GiB, 107374182400 bytes, 209715200 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xaf9b1a64

Dispositivo Inicio Comienzo      Final Sectores Tamaño Id Tipo
/dev/sda1      *          2048   97656831 97654784   46,6G 83 Linux
/dev/sda2          97656832 105656319  7999488    3,8G 82 Linux swap / Solaris
```

Creamos la particion nueva

```
root@SistemasUbuntu:~# fdisk /dev/sda
Bienvenido a fdisk (util-linux 2.31.1).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (2 primaria(s), 0 extendida(s), 2 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): e
Número de partición (3,4, valor predeterminado 3): 3
Primer sector (105656320-209715199, valor predeterminado 105656320):
Último sector, +sectores o +tamaño{K,M,G,T,P} (105656320-209715199, valor predeterminado 209715199):

Crea una nueva partición 3 de tipo 'Extended' y de tamaño 49,6 GiB.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 5
Primer sector (105658368-209715199, valor predeterminado 105658368):
Último sector, +sectores o +tamaño{K,M,G,T,P} (105658368-209715199, valor predeterminado 209715199): +10G

Crea una nueva partición 5 de tipo 'Linux' y de tamaño 10 GiB.

Orden (m para obtener ayuda): w
```

Tras reiniciar, nos aseguramos de que la particion esté creada y la formateamos con el sistema de archivos ext4

```
root@SistemasUbuntu:~# fdisk -l /dev/sda
Disco /dev/sda: 100 GiB, 107374182400 bytes, 209715200 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xaf9b1a64

Dispositivo Inicio Comienzo Final Sectores Tamaño Id Tipo
/dev/sda1 * 2048 97656831 97654784 46,6G 83 Linux
/dev/sda2 97656832 105656319 7999488 3,8G 82 Linux swap / Solaris
/dev/sda3 105656320 209715199 104058880 49,6G 5 Extendida
/dev/sda5 105658368 126629887 20971520 10G 83 Linux
root@SistemasUbuntu:~# mkfs -t ext4 /dev/sda5
mke2fs 1.44.1 (24-Mar-2018)
Se está creando un sistema de ficheros con 2621440 bloques de 4k y 655360 nodos-i
UUID del sistema de ficheros: ff096554-12dc-41fd-a976-774ff6a60a2d
Respalos del superbloque guardados en los bloques:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (16384 bloques): hecho
Escribiendo superbloques y la información contable del sistema de archivos: 0/8hecho
```

creamos la carpeta /mnt/Datos y editamos el archivo /etc/fstab para que la particion se monte al arrancar el sistema

```
root@SistemasUbuntu:~# sudo mkdir /mnt/Datos
root@SistemasUbuntu:~# sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=343e302b-3b64-424f-9f07-095aa874cd81 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda2 during installation
UUID=1649c8cc-7466-47a4-9ffd-1347b1748ec0 none swap sw 0 0
/dev/sda5 /mnt/Datos ext4 rw,user,auto 0 0
```

#### Ejercicio 4.

Iniciamos sesion con el usuario Luis y creamos el archivo con el contenido indicado

```
#!/bin/bash
clear
touch otroArchivo.txt
ls -l
```

vemos las propiedades del archivo

```
luis@SistemasUbuntu:~$ ls -l archivo
-rw-r--r-- 1 luis informatico 46 feb 18 19:46 archivo
```

El usuario propietario es luis y tiene permisos de lectura y escritura.

El grupo propietario es informatico y tienen permisos de lectura.

El resto de usuarios tienen permisos de lectura.

Modificamos los permisos a

|     |     |     |
|-----|-----|-----|
| rwX | rw- | r-- |
| 111 | 110 | 100 |
| 7   | 6   | 4   |

```
luis@SistemasUbuntu:~$ ls -l archivo
-rwxrw-r-- 1 luis informatico 46 feb 18 19:46 archivo
```

Luis es el único que puede ejecutar el archivo.

Todos los usuarios del grupo informatico pueden modificar el archivo.

Todos los usuarios pueden leer el archivo.

Ejecutamos el archivo

```
luis@SistemasUbuntu:~$ ./archivo

total 48
-rwxrw-r-- 1 luis informatico 46 feb 18 19:46 archivo
-rw-r--r-- 1 luis informatico 0 feb 15 18:29 carta
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Descargas
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Documentos
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Escritorio
-rw-r--r-- 1 luis informatico 8980 feb 15 18:06 examples.desktop
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Imágenes
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Música
-rw-r--r-- 1 luis informatico 0 feb 18 19:59 otroArchivo.txt
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Plantillas
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Público
drwxr-xr-x 2 luis informatico 4096 feb 18 19:40 Vídeos
```

Modificamos los permisos para que todos los usuarios puedan ejecutar el archivo

|     |     |     |
|-----|-----|-----|
| rwX | rwX | r-X |
| 111 | 111 | 101 |
| 7   | 7   | 5   |

```
luis@SistemasUbuntu:~$ chmod 775 archivo
luis@SistemasUbuntu:~$ ls -l archivo
-rwxrwxr-x 1 luis informatico 46 feb 18 19:46 archivo
```

## Ejercicio 5

Ejecutamos el comando sleep 100 e identificamos el proceso desde otro terminal para terminarlo

```
root      2394      1    0 19:42 ?          00:00:00 /usr/lib/fwupd/fwupd
luis      2493    1797    0 19:42 ?          00:00:02 /usr/lib/gnome-terminal/gnome-te
luis      2502    2493    0 19:42 pts/0      00:00:00 bash
luis      2525    1826    0 19:42 tty2       00:00:00 /usr/lib/deja-dup/deja-dup-monit
root      2576      2    0 19:52 ?          00:00:00 [kworker/u2:2-ev]
luis      2610    2502    0 20:04 pts/0      00:00:00 sleep 100
luis      2613    2493    0 20:04 pts/1      00:00:00 bash
luis      2621    2613    0 20:04 pts/1      00:00:00 ps -ef
luis@SistemasUbuntu:~$ kill -9 2610
```

En el primer terminal podemos observar que se ha terminado el proceso

```
luis@SistemasUbuntu:~$ sleep 100
Terminado (killed)
```

Creamos el script infinito.sh

```
#!/bin/bash
while true
do sleep 5
echo Han pasado 5 segundos
done
```

y lo ejecutamos

```
luis@SistemasUbuntu:~$ sh infinito.sh
Han pasado 5 segundos
Han pasado 5 segundos
Han pasado 5 segundos
```

vemos el listado de procesos

```
luis      2630    2502    0 20:12 pts/0      00:00:00 sh infinito.sh
luis      2640    2630    0 20:12 pts/0      00:00:00 sleep 5
luis      2641    2613    0 20:13 pts/1      00:00:00 ps -ef
```



El PID de sleep es 2640 y el proceso padre tiene como PID 2630 y es el comando de ejecucion del script.

El comando infinito.sh no termina hasta que no lo hagamos nosotros con el comando kill -9 2630

```
Han pasado 5 segundos
Han pasado 5 segundos
Terminado (killed)
```

Ejecutamos el comando yes hola > archivo.txt y vemos el consumo del procesador para este proceso, que en este caso varía entre el 15% y el 30%

```
top - 20:20:28 up 40 min, 1 user, load average: 1,65, 0,55, 0,20
Tareas: 219 total, 1 ejecutar, 183 hibernar, 0 detener, 0 zombie
%Cpu(s): 0,9 usuario, 9,3 sist, 0,0 adecuado, 0,0 inact, 76,6 en espera, 0,0 hardw int, 13,1 softw int, 0,0 robar tiempo
KiB Mem : 2035512 total, 75684 libre, 1125372 usado, 834456 búfer/caché
KiB Intercambio: 3999740 total, 3998960 libre, 780 usado, 701956 dispon Mem
```

| PID  | USUARIO  | PR | NI  | VIRT    | RES    | SHR    | S | %CPU | %MEM | HORA+   | ORDEN           |
|------|----------|----|-----|---------|--------|--------|---|------|------|---------|-----------------|
| 2698 | luis     | 20 | 0   | 14572   | 792    | 728    | D | 30,1 | 0,0  | 0:18.93 | yes             |
| 81   | root     | 20 | 0   | 0       | 0      | 0      | S | 4,6  | 0,0  | 0:02.91 | kswapd0         |
| 2697 | root     | 20 | 0   | 0       | 0      | 0      | D | 4,3  | 0,0  | 0:02.77 | kworker/u2:1+fl |
| 2016 | luis     | 20 | 0   | 3044636 | 322956 | 122920 | S | 0,7  | 15,9 | 0:26.95 | gnome-shell     |
| 2699 | luis     | 20 | 0   | 49124   | 3752   | 3116   | R | 0,7  | 0,2  | 0:00.22 | top             |
| 10   | root     | 20 | 0   | 0       | 0      | 0      | S | 0,3  | 0,0  | 0:00.16 | ksoftirqd/0     |
| 652  | message+ | 20 | 0   | 51660   | 6216   | 4004   | S | 0,3  | 0,3  | 0:00.81 | dbus-daemon     |
| 763  | root     | 20 | 0   | 0       | 0      | 0      | I | 0,3  | 0,0  | 0:01.54 | kworker/0:4-eve |
| 1817 | luis     | 20 | 0   | 485852  | 134032 | 69788  | S | 0,3  | 6,6  | 0:09.91 | Xorg            |
| 2037 | luis     | 20 | 0   | 377864  | 10184  | 8572   | S | 0,3  | 0,5  | 0:01.69 | ibus-daemon     |
| 2493 | luis     | 20 | 0   | 874040  | 40240  | 28784  | S | 0,3  | 2,0  | 0:06.06 | gnome-terminal- |
| 1    | root     | 20 | 0   | 160028  | 9152   | 6624   | S | 0,0  | 0,4  | 0:01.33 | systemd         |
| 2    | root     | 20 | 0   | 0       | 0      | 0      | S | 0,0  | 0,0  | 0:00.00 | kthreadd        |
| 3    | root     | 0  | -20 | 0       | 0      | 0      | I | 0,0  | 0,0  | 0:00.00 | rcu_gp          |
| 4    | root     | 0  | -20 | 0       | 0      | 0      | I | 0,0  | 0,0  | 0:00.00 | rcu_par_gp      |
| 5    | root     | 20 | 0   | 0       | 0      | 0      | I | 0,0  | 0,0  | 0:00.00 | kworker/0:0-eve |
| 6    | root     | 0  | -20 | 0       | 0      | 0      | I | 0,0  | 0,0  | 0:00.00 | kworker/0:0H-kb |
| 9    | root     | 0  | -20 | 0       | 0      | 0      | I | 0,0  | 0,0  | 0:00.00 | mm_percpu_wq    |
| 11   | root     | 20 | 0   | 0       | 0      | 0      | I | 0,0  | 0,0  | 0:00.22 | rcu_sched       |

Tras revisar el tamaño de archivo.txt observamos que tiene un tamaño de 37Gb. Lo borramos

```
luis@SistemasUbuntu:~$ du -sh archivo.txt
37G    archivo.txt
luis@SistemasUbuntu:~$ rm archivo.txt
```

Al iniciar un proceso con prioridad como root, no tenemos ninguna limitación. Si lo hacemos con otros usuarios, la máxima prioridad que podemos usar es 0.

Ejecutamos el comando yes con el usuario juana. Con el comando ps -efl vemos que el proceso yes tiene una prioridad de 0.

```
0 S juana      2743   2721   0  80    0 -  7431 wait    05:53 pts/1    00:00:00 bash
0 R juana      2753   2731  24  80    0 -  3643 -      05:53 pts/0    00:00:01 yes
0 R juana      2754   2743   0  80    0 - 11138 -      05:53 pts/1    00:00:00 ps -e
```

Ejecutamos de nuevo el comando `yes` y cambiamos su prioridad a 10

```
1 I root      2771      2  1  80  0 -    0 -    05:57 ?    00:00:01 [kwo
0 R juana     2772    2731 23  80  0 -   3643 -    05:59 pts/0    00:00:01 yes
0 R juana     2773    2743  0  80  0 -  11138 -    05:59 pts/1    00:00:00 ps -e
juana@SistemasUbuntu:~$ renice 10 -p 2772
2772 (process ID) prioridad anterior 0, nueva prioridad 10
```

Para volver a aumentar la prioridad del proceso, necesitamos hacerlo como root

```
jvana@SistemasUbuntu:~$ su -
Contraseña:
root@SistemasUbuntu:~# renice 0 -p 2772
2772 (process ID) prioridad anterior 10, nueva prioridad 0
```

## Ejercicio 6

La version del kernel es 5.4.0

```
juana@SistemasUbuntu:~$ uname -r
5.4.0-65-generic
```

Con el comando `lscpu` mostramos la información de nuestro procesador

```
juaana@SistemasUbuntu:~$ lscpu
Arquitectura: x86_64
modo(s) de operación de las CPUs: 32-bit, 64-bit
Orden de los bytes: Little Endian
CPU(s): 1
Lista de la(s) CPU(s) en línea: 0
Hilo(s) de procesamiento por núcleo: 1
Núcleo(s) por «socket»: 1
«Socket(s)»: 1
Modo(s) NUMA: 1
ID de fabricante: GenuineIntel
Familia de CPU: 6
Modelo: 158
Nombre del modelo: Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz
Revisión: 10
CPU MHz: 2207.996
BogoMIPS: 4415.99
Fabricante del hipervisor: KVM
Tipo de virtualización: lleno
Caché L1d: 32K
Caché L1i: 32K
Caché L2: 256K
Caché L3: 9216K
CPU(s) del nodo NUMA 0: 0
```



Accedemos como root y mostramos las últimas 10 líneas del archivo de registro

```
juana@SistemasUbuntu:~$ su -
Contraseña:
root@SistemasUbuntu:~# tail -10 /var/log/syslog
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Starting D-Bus User Message Bus Socket.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Started Pending report trigger for Ubuntu Report.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Reached target Paths.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Listening on GnuPG cryptographic agent and passphrase cache.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Listening on D-Bus User Message Bus Socket.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Reached target Sockets.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Reached target Basic System.
Feb 19 06:12:00 SistemasUbuntu systemd[1]: Started User Manager for UID 0.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Reached target Default.
Feb 19 06:12:00 SistemasUbuntu systemd[2820]: Startup finished in 109ms.
```

El \$HOME del usuario Juana ocupa 6,1Mb

```
juana@SistemasUbuntu:~$ du -sh /home/juana
6,1M    /home/juana
```

## Ejercicio 7

Creamos el script con los siguientes comandos

```
#!/bin/bash

date >> resultado7.txt
df >> resultado7.txt
top >> resultado7.txt
```

Lo ejecutamos y comprobamos que se va añadiendo la información al archivo resultado7.txt cada vez que ejecutamos el script.

Editamos el archivo /etc/crontab para que se ejecute el script todos los días, cada hora en punto

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
# juana /home/juana/7.sh
```