

Incluye, en cada ejercicio, las **capturas de pantalla** necesarias.

Gestión de procesos

1. Muestra la información de todos los procesos en formato largo.

```
luciano ~ ps -fe
UID      PID     PPID    C  STIME TTY          TIME CMD
root         1         0  0  10:08 ?        00:00:03 /sbin/init splash
root         2         0  0  10:08 ?        00:00:00 [kthreadd]
root         3         2  0  10:08 ?        00:00:00 [pool_workqueue_release]
root         4         2  0  10:08 ?        00:00:00 [kworker/R-rcu_g]
root         5         2  0  10:08 ?        00:00:00 [kworker/R-rcu_p]
root         6         2  0  10:08 ?        00:00:00 [kworker/R-slub_]
root         7         2  0  10:08 ?        00:00:00 [kworker/R-netns]
root         9         2  0  10:08 ?        00:00:00 [kworker/0:0H-events_highpri]
root        11         2  0  10:08 ?        00:00:00 [kworker/R-mm_pe]
root        12         2  0  10:08 ?        00:00:00 [rcu_tasks_kthread]
root        13         2  0  10:08 ?        00:00:00 [rcu_tasks_rude_kthread]
root        14         2  0  10:08 ?        00:00:00 [rcu_tasks_trace_kthread]
root        16         2  0  10:08 ?        00:00:00 [ksoftirqd/0]
root        17         2  0  10:08 ?        00:00:01 [rcu_preempt]
root        18         2  0  10:08 ?        00:00:00 [migration/0]
root        19         2  0  10:08 ?        00:00:00 [idle_inject/0]
root        20         2  0  10:08 ?        00:00:00 [cpuhp/0]
root        21         2  0  10:08 ?        00:00:00 [cpuhp/1]
root        22         2  0  10:08 ?        00:00:00 [idle_inject/1]
root        23         2  0  10:08 ?        00:00:00 [migration/1]
root        24         2  0  10:08 ?        00:00:00 [ksoftirqd/1]
```

2. Muestra el árbol de procesos actuales en el sistema.

```
luciano ~ pstree
systemd--ModemManager--3*[{ModemManager}]
systemd--NetworkManager--3*[{NetworkManager}]
systemd--accounts-daemon--3*[{accounts-daemon}]
systemd--acpid
systemd--avahi-daemon--avahi-daemon
systemd--bluetoothd
systemd--cron
systemd--cups-browsed--3*[{cups-browsed}]
systemd--cupsd
systemd--dbus-daemon
systemd--firewalld--{firewalld}
systemd--ibus-daemon--ibus-dconf--4*[{ibus-dconf}]
systemd--ibus-daemon--ibus-engine-sim--3*[{ibus-engine-sim}]
systemd--ibus-daemon--ibus-extension--6*[{ibus-extension-}]
systemd--ibus-daemon--ibus-ui-gtk3--7*[{ibus-ui-gtk3}]
systemd--ibus-daemon--3*[{ibus-daemon}]
systemd--ibus-x11--5*[{ibus-x11}]
systemd--irqbalance--{irqbalance}
systemd--lightdm--Xorg--7*[{Xorg}]
systemd--lightdm--lightdm--xfce4-session--xfce4-session--Thunar--3*[{Thunar}]
systemd--lightdm--lightdm--xfce4-session--xfce4-session--aplet.py
systemd--lightdm--lightdm--xfce4-session--xfce4-session--ayatana-indicat--3*[{ayatana-indicat}]
systemd--lightdm--lightdm--xfce4-session--xfce4-session--nm-applet--4*[{nm-applet}]
systemd--lightdm--lightdm--xfce4-session--xfce4-session--package-update--3*[{package-update-}]
systemd--lightdm--lightdm--xfce4-session--xfce4-session--polkit-gnome-au--3*[{polkit-gnome-au}]
systemd--lightdm--lightdm--xfce4-session--xfce4-session--xfce4-panel--panel-10-whiske--firefox-bin
```

3. ¿Cómo se llama el proceso padre de toda la jerarquía? Averigua cuál es su PID y su PPID.

Systemd

4. Visualiza sólo los procesos correspondientes a tu usuario.

```
luciano ~ ps -eu luciano
  PID TTY          TIME CMD
    1 ?            00:00:04 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 pool_workqueue_release
    4 ?            00:00:00 kworker/R-rcu_g
    5 ?            00:00:00 kworker/R-rcu_p
    6 ?            00:00:00 kworker/R-slub_
    7 ?            00:00:00 kworker/R-netns
    9 ?            00:00:00 kworker/0:0H-events_highpri
   11 ?            00:00:00 kworker/R-mm_pe
   12 ?            00:00:00 rcu_tasks_kthread
   13 ?            00:00:00 rcu_tasks_rude_kthread
   14 ?            00:00:00 rcu_tasks_trace_kthread
   16 ?            00:00:00 ksoftirqd/0
   17 ?            00:00:02 rcu_preempt
   18 ?            00:00:00 migration/0
```

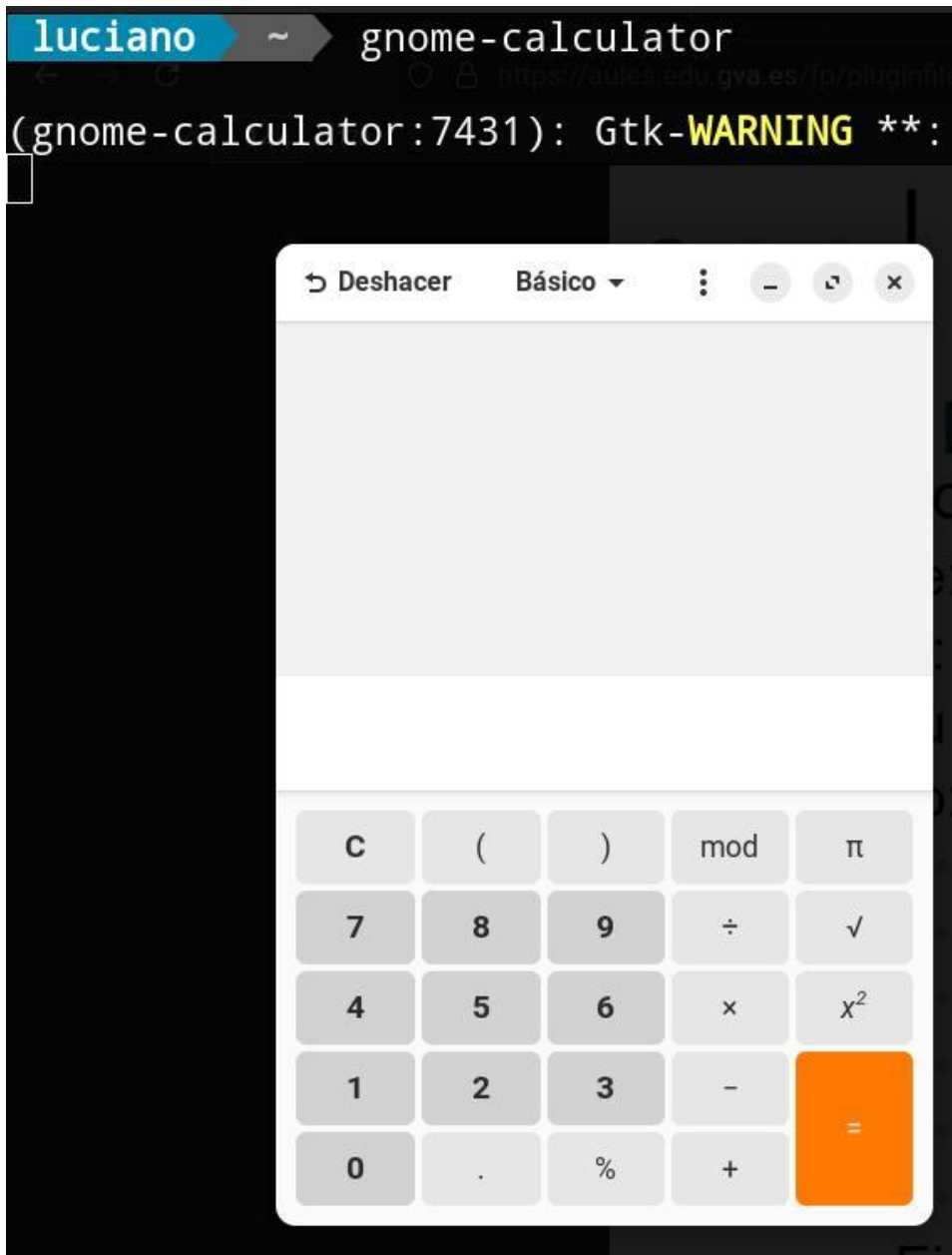
5. Visualiza sólo los procesos pertenecientes al superusuario.

```
luciano ~ ps -eu root
  PID TTY          TIME CMD
    1 ?            00:00:04 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 pool_workqueue_release
    4 ?            00:00:00 kworker/R-rcu_g
    5 ?            00:00:00 kworker/R-rcu_p
    6 ?            00:00:00 kworker/R-slub_
    7 ?            00:00:00 kworker/R-netns
    9 ?            00:00:00 kworker/0:0H-events_highpri
   11 ?            00:00:00 kworker/R-mm_pe
   12 ?            00:00:00 rcu_tasks_kthread
   13 ?            00:00:00 rcu_tasks_rude_kthread
   14 ?            00:00:00 rcu_tasks_trace_kthread
```

6. Muestra sólo los siguientes datos de todos los procesos: pid, estado, % uso de cpu, % uso de memoria y comando.

```
luciano ~ ps -efo pid,s,%cpu,%mem,comm
  PID S %CPU %MEM COMMAND
 3770 S  0.0  0.0 bash SSH_AUTH_SOCK=/tmp/ssh-1NQLuA1UvEBm/agent.2099 LANGUAGE=es:en LANG=es_ES.UTF-8 PAPERSIZE=letter XDG_CURRENT_DE
 7370 R 200  0.0 \_ ps SHELL=/bin/bash SESSION_MANAGER=local/luciano-Wyse-5470:@/tmp/.ICE-unix/2021,unix/luciano-Wyse-5470:/tmp/.IC
```

7. Abre, desde la consola, la calculadora (`gnome-calculator`).



8. ¿Puedes seguir tecleando comandos en la consola?

no

9. Abre ahora la calculadora en segundo plano.


```
luciano ~ gnome-calculator&
[1] 7598
luciano ~ 1
(gnome-calculator:7598): Gtk-WARNING:
/home/luciano/.cache/gnome-calculator/
^

** (gnome-calculator:7598): WARNING:
me-calculator/eurofxref-daily.xml
/home/luciano/.cache/gnome-calculator/
^

** (gnome-calculator:7598): WARNING:
calculator/fxtwdcad.xml
luciano ~ 1
```

10. Cierra la calculadora con un comando.

```
luciano ~ 1 kill 7598
[1]+ Terminado gnome-calculator
luciano ~ 1
```

11. Averigua el PID del proceso asociado a la ventana de terminal (**bash**).

```
luciano ~ ps
  PID TTY          TIME CMD
 3770 pts/0    00:00:00 bash
 7790 pts/0    00:00:00 ps
```

12. Intenta terminarlo enviando la señal por defecto. ¿Se cierra la consola?

Kill 3770, no se cierra.

13. Envía una señal adecuada para que se cierre la consola.

Kill -9 3770

14. Lanza el **gnome-text-editor** en segundo plano y termina el proceso por su nombre.



```
luciano ~ mousepad&
[1] 8267
luciano ~ 1 killall mousepad
luciano ~ 1
[1]+ Hecho
luciano ~ 1
luciano ~
```

The image shows a terminal window with a dark background. The prompt is 'luciano ~'. The first command entered is 'mousepad&', which runs in the background. The prompt changes to '[1] 8267'. The second command entered is 'killall mousepad'. The prompt changes to 'luciano ~ 1'. The third command entered is 'killall mousepad' again, and the prompt changes to 'luciano ~ 1'. The fourth command entered is 'killall mousepad' again, and the prompt changes to '[1]+ Hecho'. The fifth command entered is 'killall mousepad' again, and the prompt changes to 'luciano ~ 1'. The sixth command entered is 'killall mousepad' again, and the prompt changes to 'luciano ~'. On the right side of the terminal, there is a window titled 'mousepad' with three buttons.

15. Ejecuta el comando **gnome-calculator** y, a continuación:

- Pulsa las teclas adecuadas para detenerlo (suspenderlo).
Ctrl + z
- Envía el trabajo a ejecutarse en segundo plano.

```
luciano ~ gnome-calculator
(gnome-calculator:8552): Gtk-WARNING **: 11:45:02.03
/home/luciano/.cache/gnome-calculator/eurofxref-dail
^
** (gnome-calculator:8552): WARNING **: 11:45:02.143
me-calculator/eurofxref-daily.xml
/home/luciano/.cache/gnome-calculator/fxtwdcad.xml:1
^
** (gnome-calculator:8552): WARNING **: 11:45:02.151
calculator/fxtwdcad.xml
^Z
[1]+ Detenido
luciano ~ 1 SIGTSTP bg "%gnome-calculator"
[1]+ gnome-calculator &
luciano ~ 1
```

c. Muestra el estado de los trabajos.

```
luciano ~ 1 jobs
[1]+ Ejecutando      gnome-calculator &
```

d. Por último, termina el proceso.

```
luciano ~ 1 killall gnome-calculator
[1]+ Terminado      gnome-calculator
```

16. Ejecuta el comando `man ps &` y, a continuación:

a. Muestra el estado de los trabajos.

```
luciano ~ man ps &
[1] 8848
luciano ~ 1 jobs
[1]+ Detenido man ps
```

- b. Envía a ejecutarse en primer plano el trabajo asociado a **man**.

```
luciano ~ 1 1 fg "%man"
man ps
[1]+ Detenido man ps
```

- c. Pulsa las teclas CTRL+Z para detener el programa sin finalizarlo.
- d. Intenta terminar el proceso enviándole la señal por defecto (15). ¿Se puede terminar?
Kill 8848 no.
- e. Envía la señal adecuada para terminarlo.
Kill -9 8848

17. En tu HOME, crea los siguientes scripts (ficheros de texto creados con **nano**):

script1.sh

```
#!/bin/bash
echo "Ejecución del script1"
while true
do
    sleep 30
    echo "Ejecución del script1"
done
```

script2.sh

```
#!/bin/bash
echo "Ejecución del script2"
while true
do
    sleep 30
    echo "Ejecución del script2"
done
```

script3.sh

```
#!/bin/bash
echo "Ejecución del script3"
while true
do
    sleep 30
    echo "Ejecución del script3"
done
```

- a. Asigna permiso de ejecución a los 3 scripts.

```
luciano ~ scripts ls -l
total 12
-rw-rw-r-- 1 luciano luciano 106 Jan 30 11:55 script1.sh
-rw-rw-r-- 1 luciano luciano 106 Jan 30 11:55 script2.sh
-rw-rw-r-- 1 luciano luciano 106 Jan 30 11:56 script3.sh
luciano ~ scripts chmod 775 script1.sh script2.sh script3.sh
luciano ~ scripts ls -l
total 12
-rwxrwxr-x 1 luciano luciano 106 Jan 30 11:55 script1.sh
-rwxrwxr-x 1 luciano luciano 106 Jan 30 11:55 script2.sh
-rwxrwxr-x 1 luciano luciano 106 Jan 30 11:56 script3.sh
luciano ~ scripts
```


- b. Lanza el primer script (`./script1.sh`) y pulsa las teclas adecuadas para detenerlo (suspenderlo). A continuación, ejecuta `./script2.sh` y suspéndelo. Después ejecuta `./script3.sh` y lo suspendes también.

```
luciano ~ > scripts > ./script1.sh
Ejecución del script1
^Z
[1]+  Detenido                  ./script1.sh
luciano ~ > scripts > 1 > SIGTSTP > ./script2.sh
Ejecución del script2
^Z
[2]+  Detenido                  ./script2.sh
luciano ~ > scripts > 2 > SIGTSTP > ./script3.sh
Ejecución del script3
^Z
[3]+  Detenido                  ./script3.sh
luciano ~ > scripts > 3 > SIGTSTP >
```

- c. Muestra los trabajos de esta terminal.

```
luciano ~ > scripts > 3 > SIGTSTP > jobs
[1] Detenido                  ./script1.sh
[2]- Detenido                  ./script2.sh
[3]+ Detenido                  ./script3.sh
```

- d. Ejecuta el comando `jobs -l`. ¿Qué información adicional aparece?

```
luciano ~ > scripts > 3 > jobs -l
[1] 9478 Parado                ./script1.sh
[2]- 9490 Parado                ./script2.sh
[3]+ 9499 Parado                ./script3.sh
```


- e. Envía a primer plano el último trabajo detenido. A continuación, suspéndelo de nuevo.

```
luciano ~ > scripts 3 fg
./script3.sh
Ejecución del script3
^Z
[3]+ Detenido ./script3.sh
```

- f. Envía a primer plano la ejecución del **script1.sh**. A continuación, suspéndelo de nuevo.

```
luciano ~ > scripts 3 fg %1
./script1.sh
Ejecución del script1
^Z
[1]+ Detenido ./script1.sh
```

- g. Ejecuta el comando **bg**. Explica lo que ocurre.

[Envía a segundo plano el ultimo trabajo](#)

- h. Envía al **script2.sh** a ejecutarse en segundo plano.

[Bg 2](#)

- i. Envía al **script3.sh** a ejecutarse en segundo plano.

[Bg 3](#)

- j. Termina los 3 procesos a la vez.

```
kill 9478 9490 9499
```

Abre el **Monitor del sistema** y realiza las actividades indicadas a continuación.

18. Abre el Editor de textos (el proceso se llama *gnome-text-editor*).

19. Detén el proceso e intenta escribir algo en el editor. A continuación, reanuda la ejecución del proceso. ¿Qué ocurre?

[No deja escribir si se detiene.](#)

20. Disminuye la prioridad del proceso *gnome-text-editor*.

[Clic derecho y prioridad](#)

21. Añade alguna columna más para visualizar datos que no aparecen por defecto.

[Tres puntos y a preferencias](#)

22. Termina el proceso del editor de textos desde esta herramienta gráfica.

[Clic derecho matar](#)

Servicios

23. Muestra el estado de todos los servicios.

```
luciano ~$ systemctl list-units --type service --all
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
accounts-daemon.service	loaded	active	running	Accounts Service
acpid.service	loaded	active	running	ACPI event daemon
alsa-restore.service	loaded	active	exited	Save/Restore Sound Card State
alsa-state.service	loaded	inactive	dead	Manage Sound Card State (restore and s
apparmor.service	loaded	active	exited	Load AppArmor profiles
apt-daily-upgrade.service	loaded	inactive	dead	Daily apt upgrade and clean activities
apt-daily.service	loaded	inactive	dead	Daily apt download activities
apt-xapian-index.service	loaded	inactive	dead	Update APT-Xapian-Index
aptpatchll.service	loaded	inactive	dead	Patch /etc/apt/ ensure minimal Install
auditd.service	loaded	active	running	Security Auditing Service
avahi-daemon.service	loaded	active	running	Avahi mDNS/DNS-SD Stack
blk-availability.service	loaded	active	exited	Availability of block devices
blueman-mechanism.service	loaded	inactive	dead	Bluetooth management mechanism
bluetooth.service	loaded	active	running	Bluetooth service

24. Comprueba el estado del servicio auditd.service.

```
luciano ~$ systemctl status auditd.service
```

```
• auditd.service - Security Auditing Service
   Loaded: loaded (/usr/lib/systemd/system/auditd.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-01-30 12:35:29 CET; 1min 34s ago
     Docs: man:auditd(8)
           https://github.com/linux-audit/audit-documentation
   Process: 11338 ExecStart=/sbin/auditd (code=exited, status=0/SUCCESS)
   Process: 11348 ExecStartPost=/sbin/augenrules --load (code=exited, status=0/SUCCESS)
   Main PID: 11344 (auditd)
     Tasks: 2 (limit: 9197)
    Memory: 480.0K (peak: 1.7M)
       CPU: 52ms
    CGroup: /system.slice/auditd.service
           └─11344 /sbin/auditd
```

```
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: enabled 1
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: failure 1
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: pid 11344
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: rate_limit 0
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog_limit 8192
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: lost 0
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog 4
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog_wait_time 60000
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog_wait_time_actual 0
Jan 30 12:35:29 luciano-Wyse-5470 systemd[1]: Started auditd.service - Security Auditing Service.
```

25. Instala el paquete auditd y comprueba de nuevo el estado del servicio auditd.service.

```
luciano ~ systemctl status auditd.service
● auditd.service - Security Auditing Service
   Loaded: loaded (/usr/lib/systemd/system/auditd.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-01-30 12:35:29 CET; 1min 34s ago
     Docs: man:auditd(8)
           https://github.com/linux-audit/audit-documentation
   Process: 11338 ExecStart=/sbin/auditd (code=exited, status=0/SUCCESS)
   Process: 11348 ExecStartPost=/sbin/augenrules --load (code=exited, status=0/SUCCESS)
  Main PID: 11344 (auditd)
    Tasks: 2 (limit: 9197)
   Memory: 480.0K (peak: 1.7M)
      CPU: 52ms
   CGroup: /system.slice/auditd.service
           └─11344 /sbin/auditd

Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: enabled 1
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: failure 1
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: pid 11344
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: rate_limit 0
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog_limit 8192
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: lost 0
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog 4
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog_wait_time 60000
Jan 30 12:35:29 luciano-Wyse-5470 augenrules[11361]: backlog_wait_time_actual 0
Jan 30 12:35:29 luciano-Wyse-5470 systemd[1]: Started auditd.service - Security Auditing Service.
```

26. Reinicia el servicio `auditd.service`.

```
luciano ~ systemctl restart auditd.service
luciano ~
```

27. Si se modificara el fichero de configuración de este servicio, ¿qué se puede hacer para que se tengan en cuenta los cambios?

`Systemctl reload/rest`

28. Detén el servicio `cups.service`.

`Systemctl stop cups.service`

29. Comprueba el estado del servicio `cups.service`.

`Systemctl status cups.service`

```
luciano ~ systemctl status cups.service
● cups.service - CUPS Scheduler
   Loaded: loaded (/usr/lib/systemd/system/cups.service; enabled; preset: enabled)
   Active: inactive (dead) since Thu 2025-01-30 12:41:12 CET; 47s ago
 Duration: 2h 32min 22.924s
 TriggeredBy: ○ cups.path
               ○ cups.socket
     Docs: man:cupsd(8)
   Process: 1311 ExecStart=/usr/sbin/cupsd -l (code=exited, status=0/SUCCESS)
  Main PID: 1311 (code=exited, status=0/SUCCESS)
   Status: "Scheduler is running..."
      CPU: 63ms

Jan 30 10:08:49 luciano-Wyse-5470 systemd[1]: Starting cups.service - CUPS Scheduler...
Jan 30 10:08:49 luciano-Wyse-5470 systemd[1]: Started cups.service - CUPS Scheduler.
Jan 30 12:41:12 luciano-Wyse-5470 systemd[1]: Stopping cups.service - CUPS Scheduler...
Jan 30 12:41:12 luciano-Wyse-5470 systemd[1]: cups.service: Deactivated successfully.
Jan 30 12:41:12 luciano-Wyse-5470 systemd[1]: Stopped cups.service - CUPS Scheduler.
```

30. Inicia el servicio `cups.service`.

`Systemctl start cups.service`

Programación de tareas

31. Todos los viernes a las 9 y media de la mañana se ejecutará el script: `~/script1.sh`.

`30 9 * * 5 luciano ./home/luciano/script1.sh`

32. Todos los sábados cada 15 minutos se guardará, en un fichero de tu HOME, un listado con todos los procesos en ejecución de tu usuario.

`15 * * * * luciano ps -e`

33. A las 6:30 am todos los lunes, miércoles y viernes se creará, con el comando `tar`, una copia comprimida con `gzip` del directorio `/home` en el directorio `/media/backup`.

`30 6 * * 1,3,5 luciano tar -czf /media/backup/home_backup.tar.gz /home`

34. Todos los días durante el mes de agosto a las 12 de la noche se guardará en un fichero la salida del comando que muestra los usuarios que están conectados en ese momento.

`0 0 * 8 * luciano who > ~/usuarios_conectados.txt`

35. Los viernes de diciembre a las 16:00 se ejecutará el script: `~/script2.sh`.

`0 16 * 12 5 luciano ~/script2.sh`

36. El día 23 de noviembre a las 9:30 se copiará en tu HOME, el fichero que contiene los usuarios dados de alta en el sistema.

`30 9 23 11 * luciano cp /etc/passwd ~/usuarios.txt`

37. A las 12 del mediodía todos los días 1, 15 y 28 de cada mes se vaciará el directorio Descargas de tu HOME.

`0 12 1,15,28 * * luciano rm -rf ~/Descargas/*`

38. Todos los domingos a las 10:00 a.m, se ejecutará el comando que actualiza la información de los repositorios.

`0 10 * * 0 luciano sudo apt update`

39. Cada 8 horas ejecute un script que:

- Muestre la fecha y la hora y la guarde en un fichero (sin borrar el contenido).
- Muestre el espacio ocupado en el disco `/dev/sda` y lo guarde en el mismo fichero.

0 * / 8 * * * luciano (date >> ~/registro.txt; df -h /dev/sda >> ~/registro.txt)