

IES AGUADULCE

# ACTIVIDAD 8

PARTE II: LINUX

Francisco Jesús Díaz Martínez  
12-5-2023

# INDICE

## ACTIVIDAD 1

Pág.1

## ACTIVIDAD 2

Pág.5

## ACTIVIDAD 3

Pág.12

## ACTIVIDAD 4

Pág.14

## ACTIVIDAD 5

Pág.16

## ACTIVIDAD 6

Pág.18

(Resultados completos de ACL en la pág.19)

## Actividad 1.- Cuentas de usuarios y grupos locales.

En esta primera actividad se nos solicitan varios puntos de creación y listado mediante comandos en el terminal de Linux:

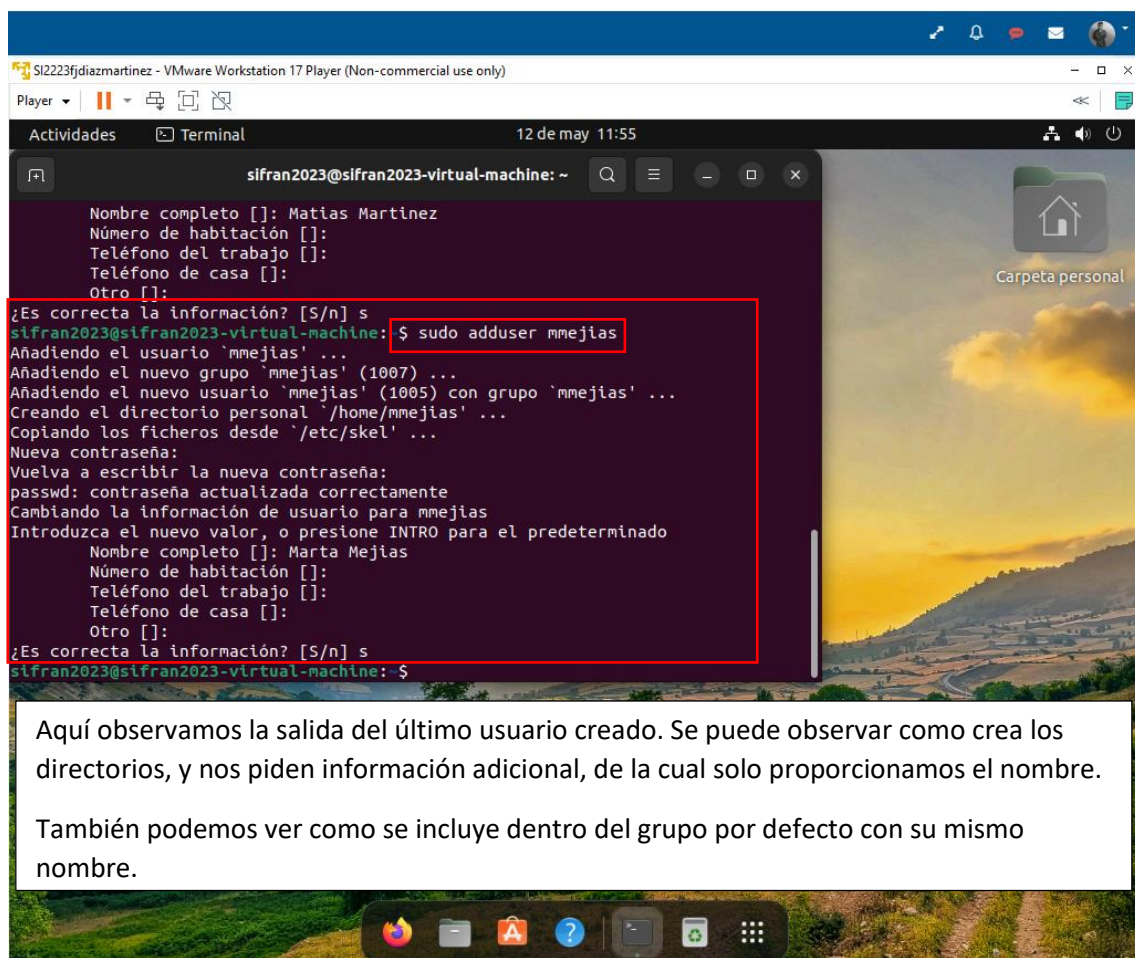
- Creación de usuarios
- Creación de grupos
- Listado de usuarios
- Listado de inclusión de los nuevos usuarios en sus correspondientes grupos

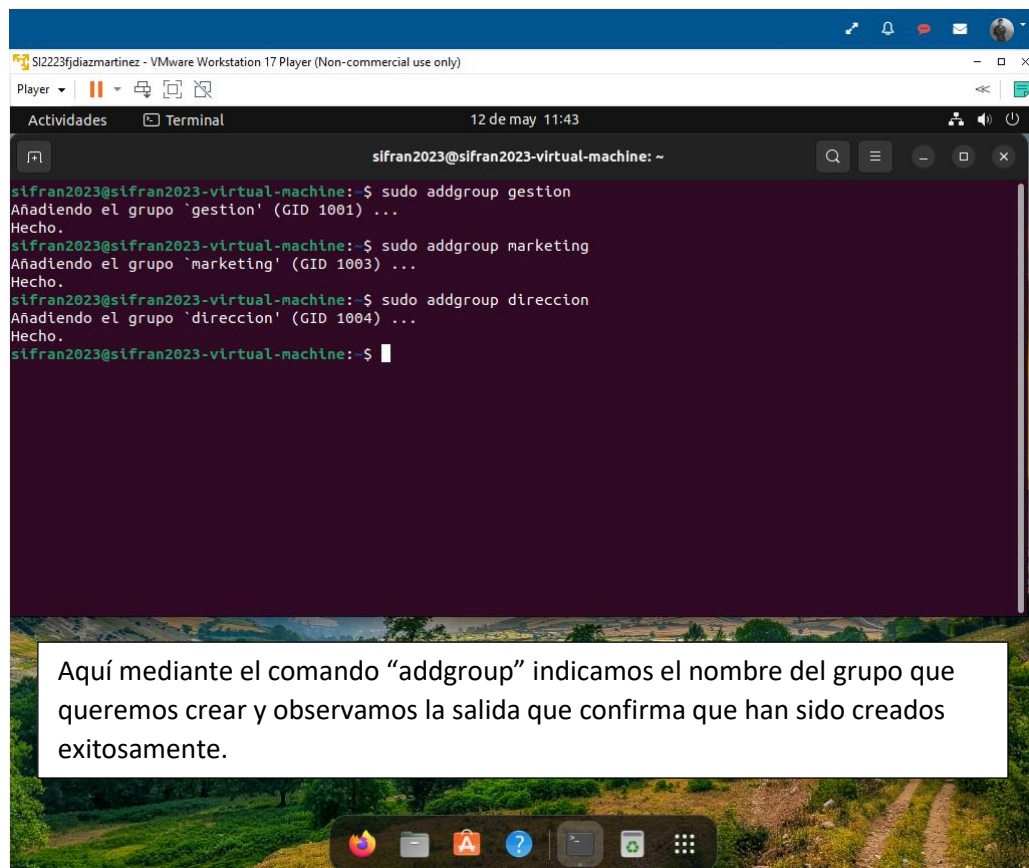
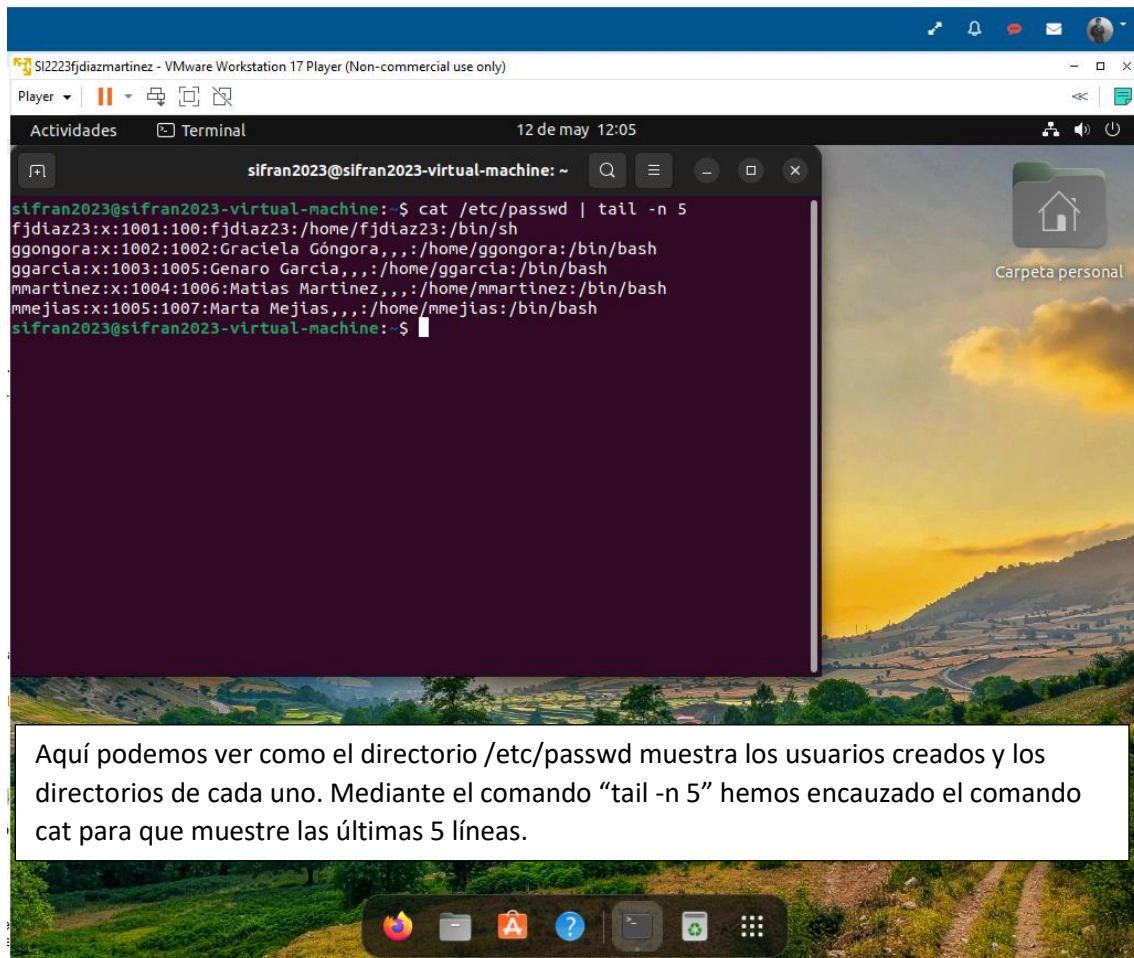
Las capturas que mostraremos a continuación atestiguan este proceso, pero primero, para hacer más sencillo el proceso, voy a recopilar los comandos usados y sus correspondientes funciones.

### Comandos de terminal:

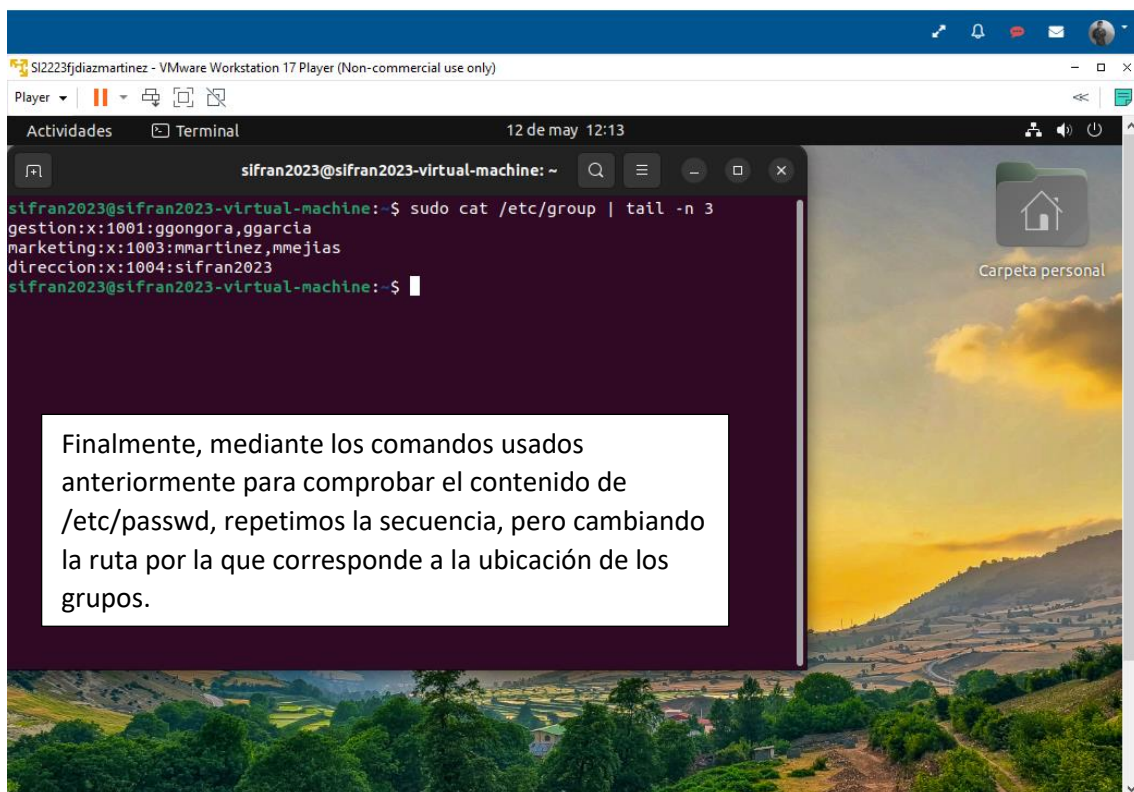
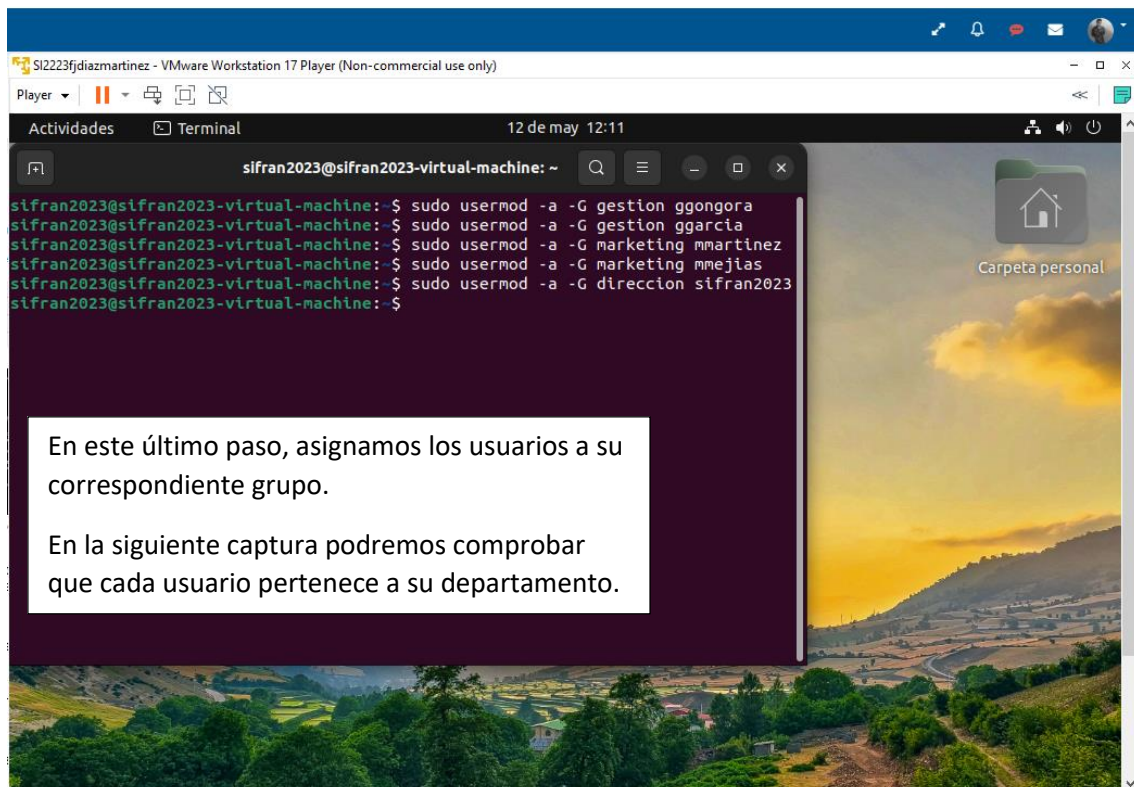
- **Creación de usuarios:** sudo adduser (nombre\_usuario)
- **Creación de grupos:** sudo addgroup (nombre\_grupo)
- **Listado de grupos:** sudo cat (ruta\_grupos)
- **Listado de usuarios:** sudo cat (ruta\_usuarios)
- **Comando adicional:** tail -n (número de líneas que se desean mostrar)

A continuación, las capturas con la ejecución de los comandos y la salida proporcionada.





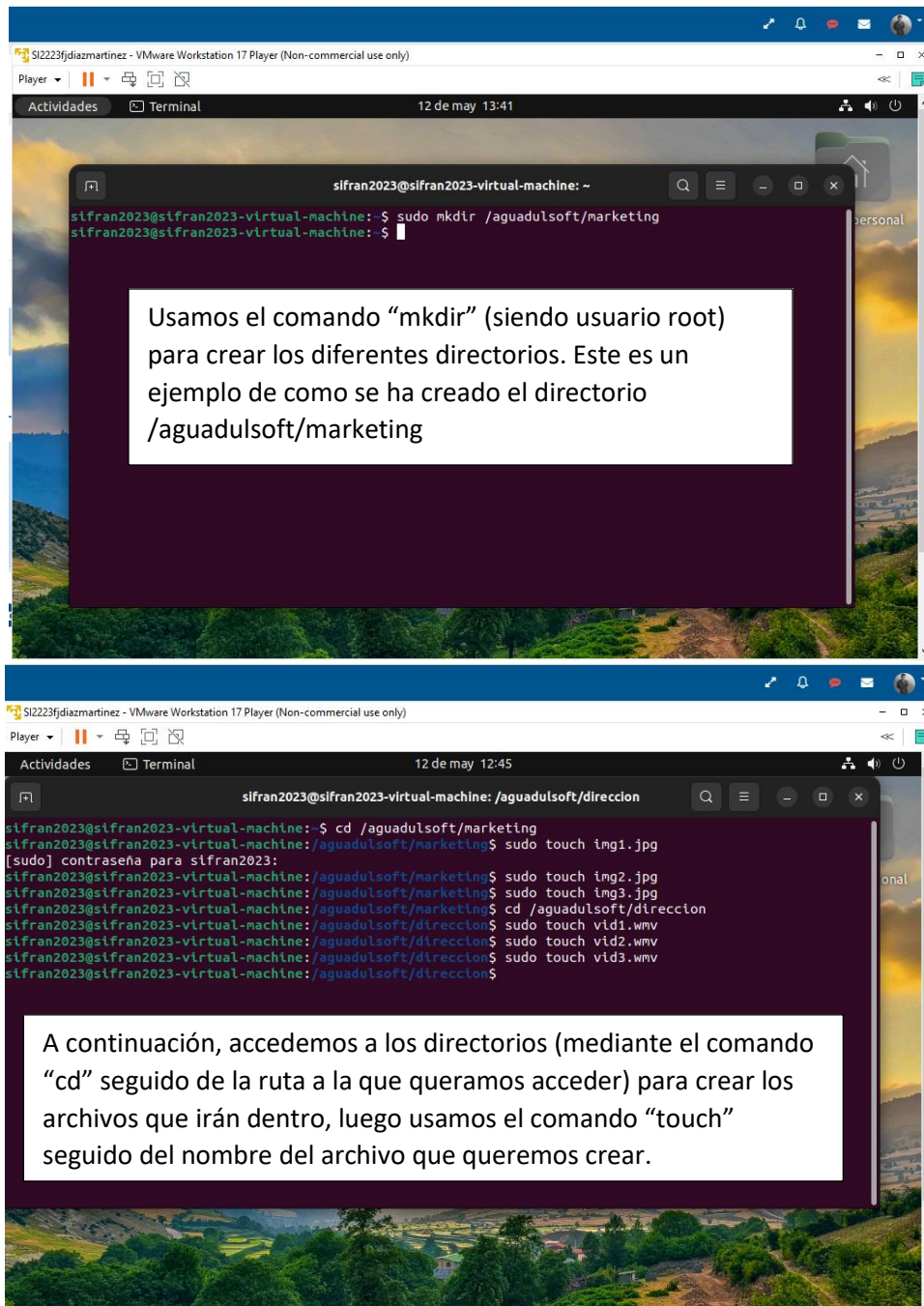




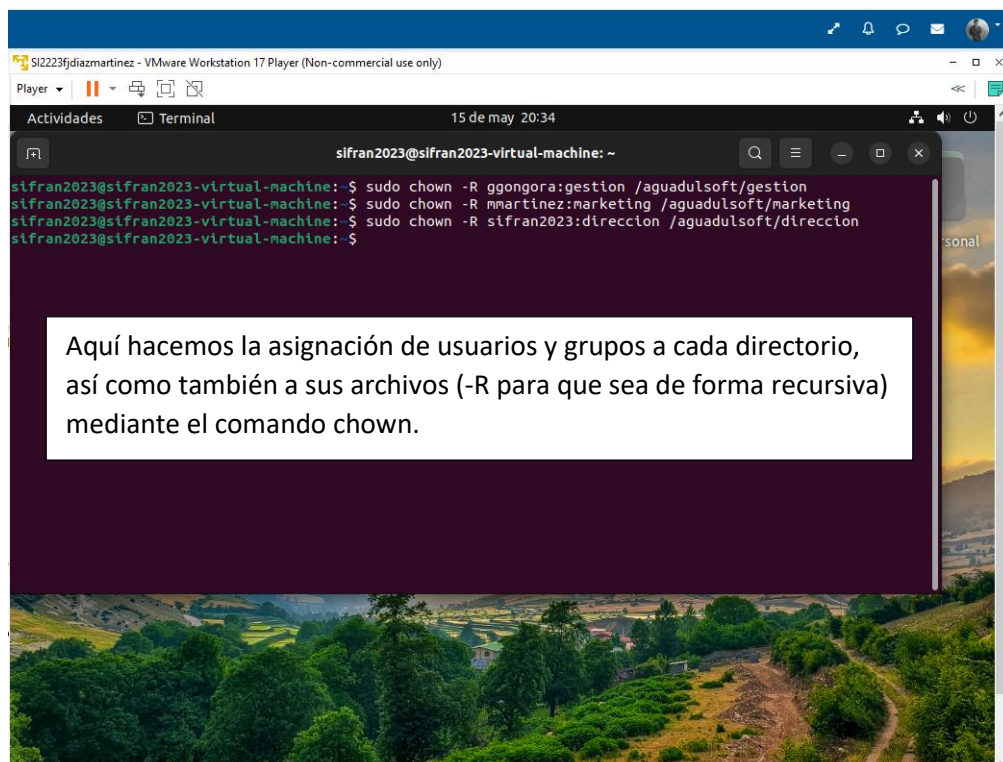
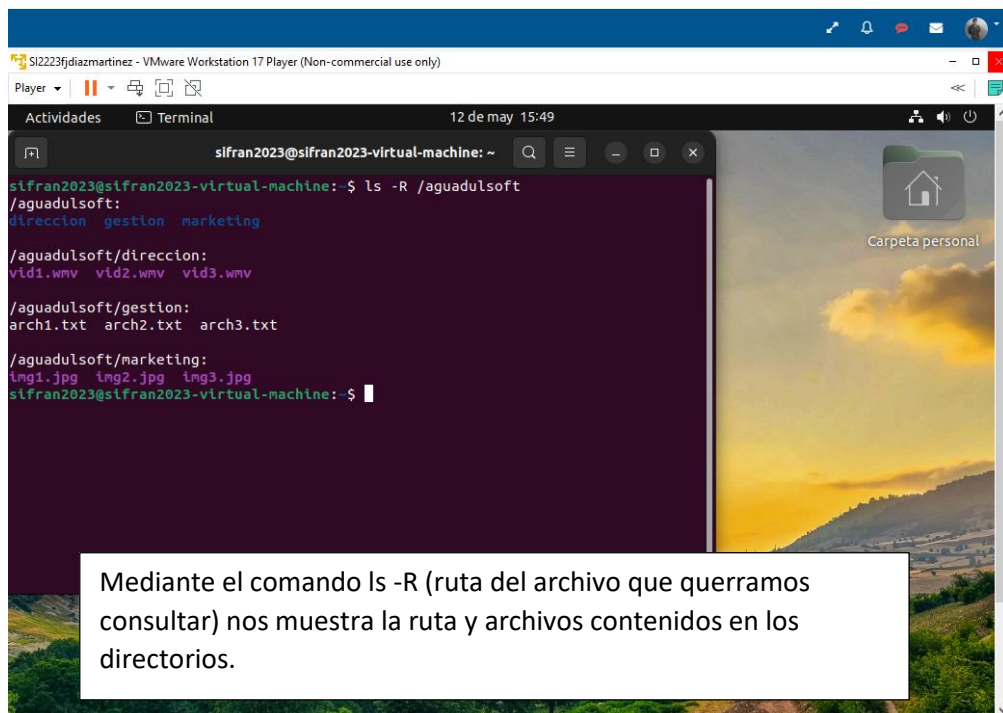
## Actividad 2.- Estructura departamental y permisos de acceso.

En esta actividad se nos pide la creación de un directorio (colgado del directorio raíz “/”) llamado aguadulsoft con tres departamentos y tres archivos dentro de cada uno. Después de ello deberemos de asignar diferentes permisos a cada usuario en relación a su departamento y los demás. No usaremos ni mayúsculas ni caracteres especiales.

Primero crearemos los directorios y los archivos correspondientes, usando el comando “touch” y “mkdir”.

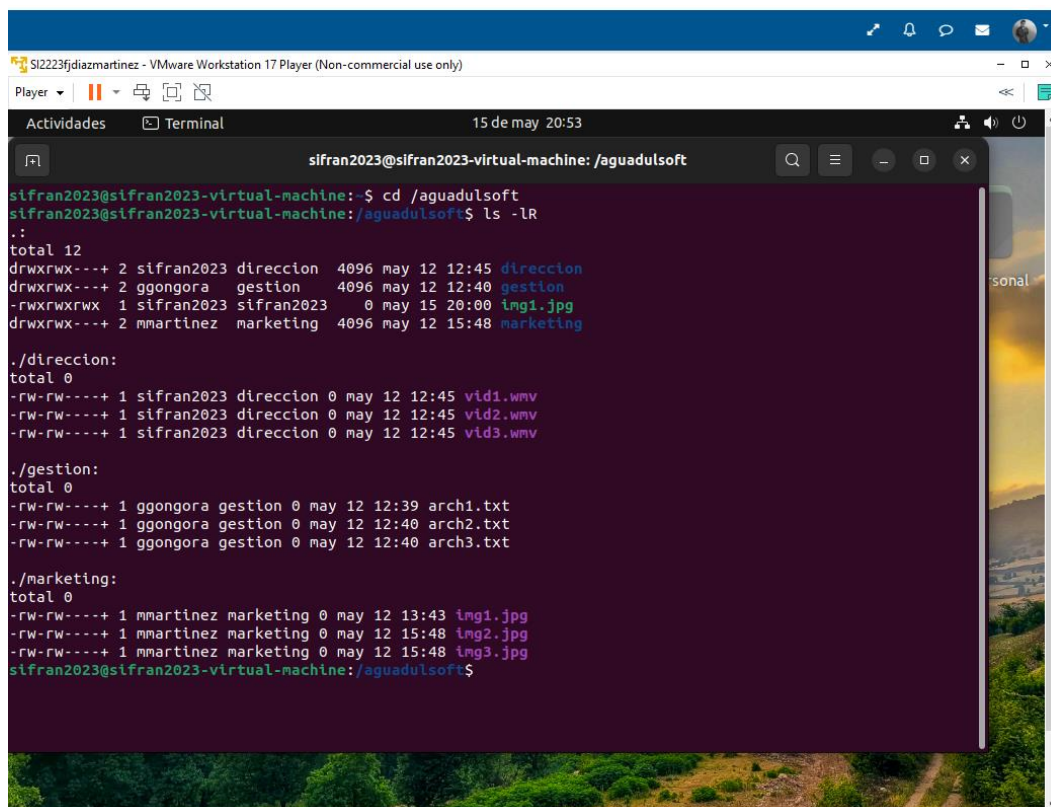
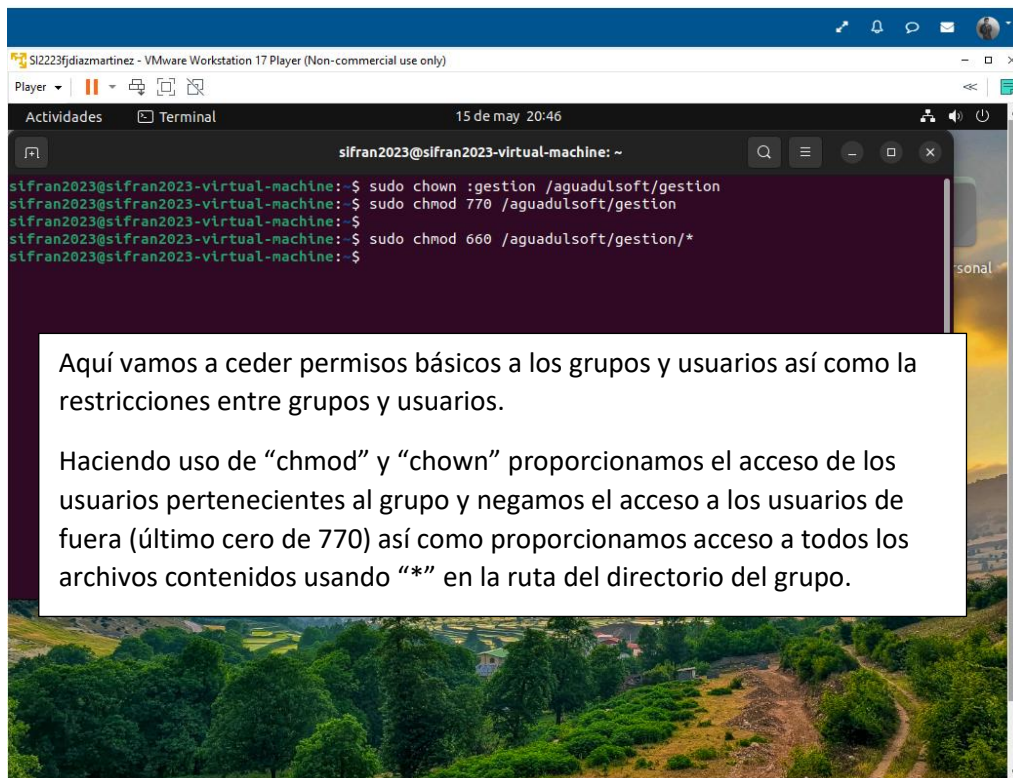


Posteriormente presentamos la estructura departamental que comprueba que todo el proceso descrito anteriormente ha surtido efecto.



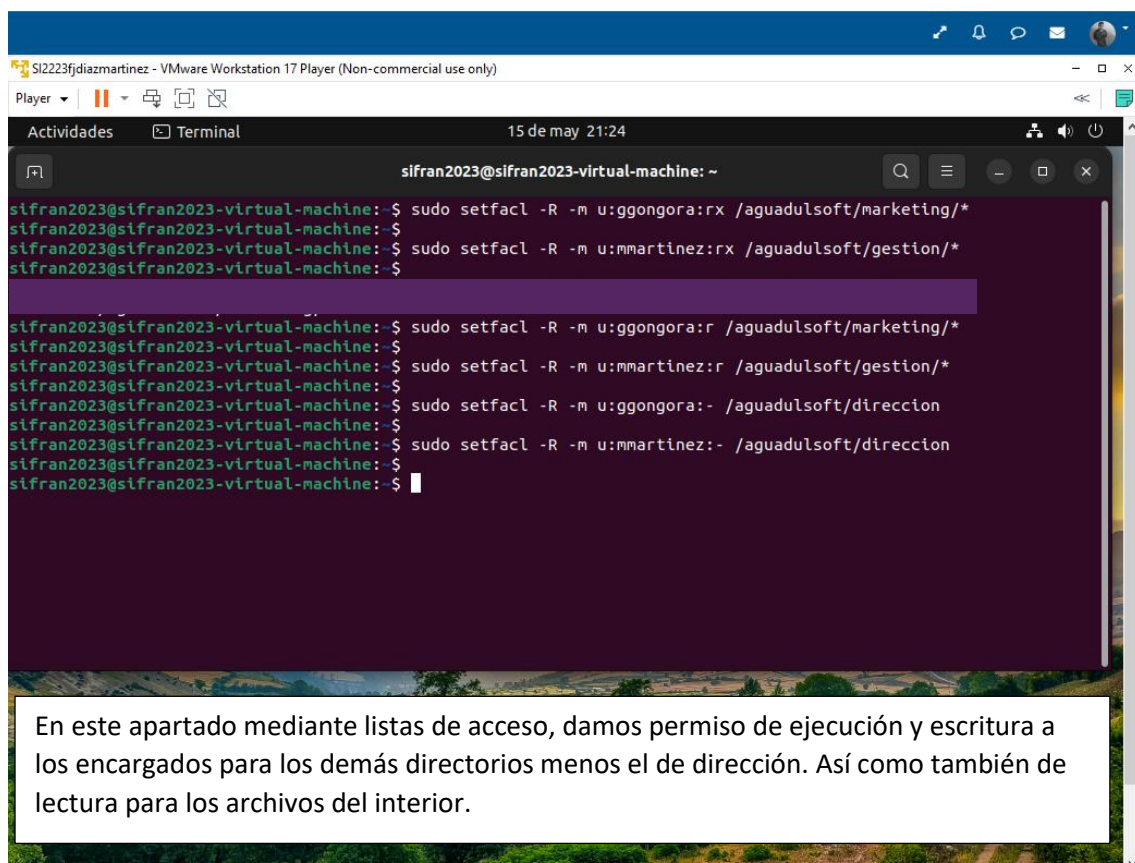
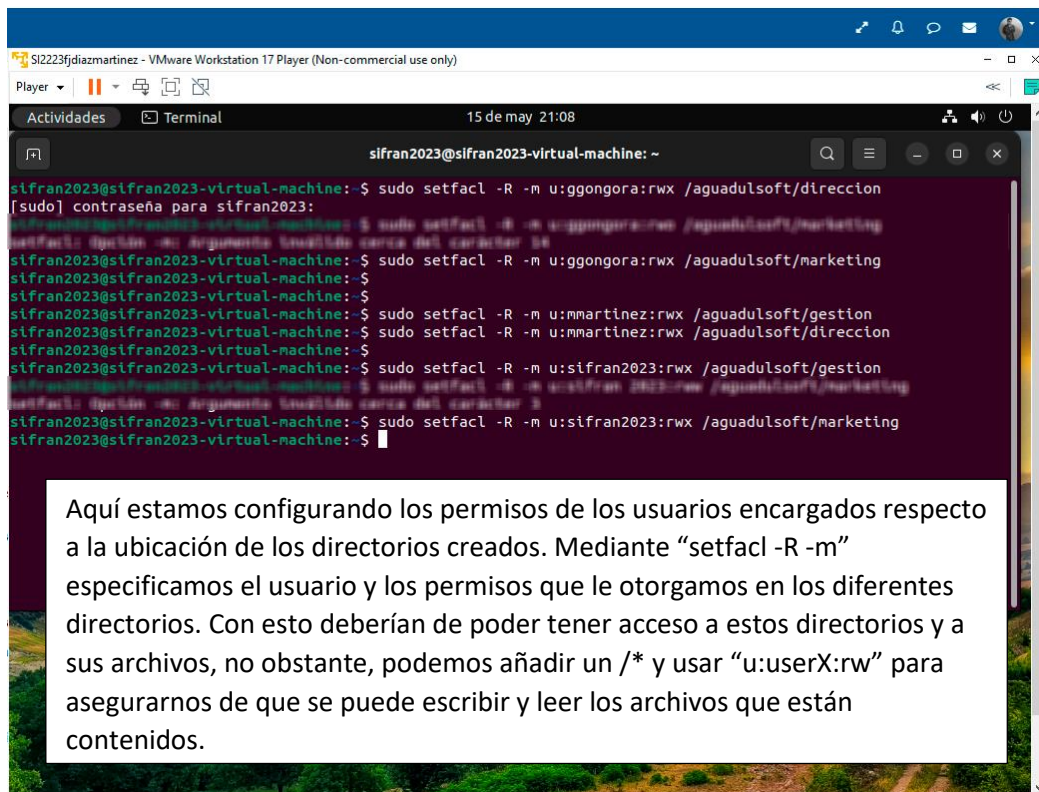
Todas las ejecuciones las hacemos como sudo, actualmente estamos usando el usuario root.



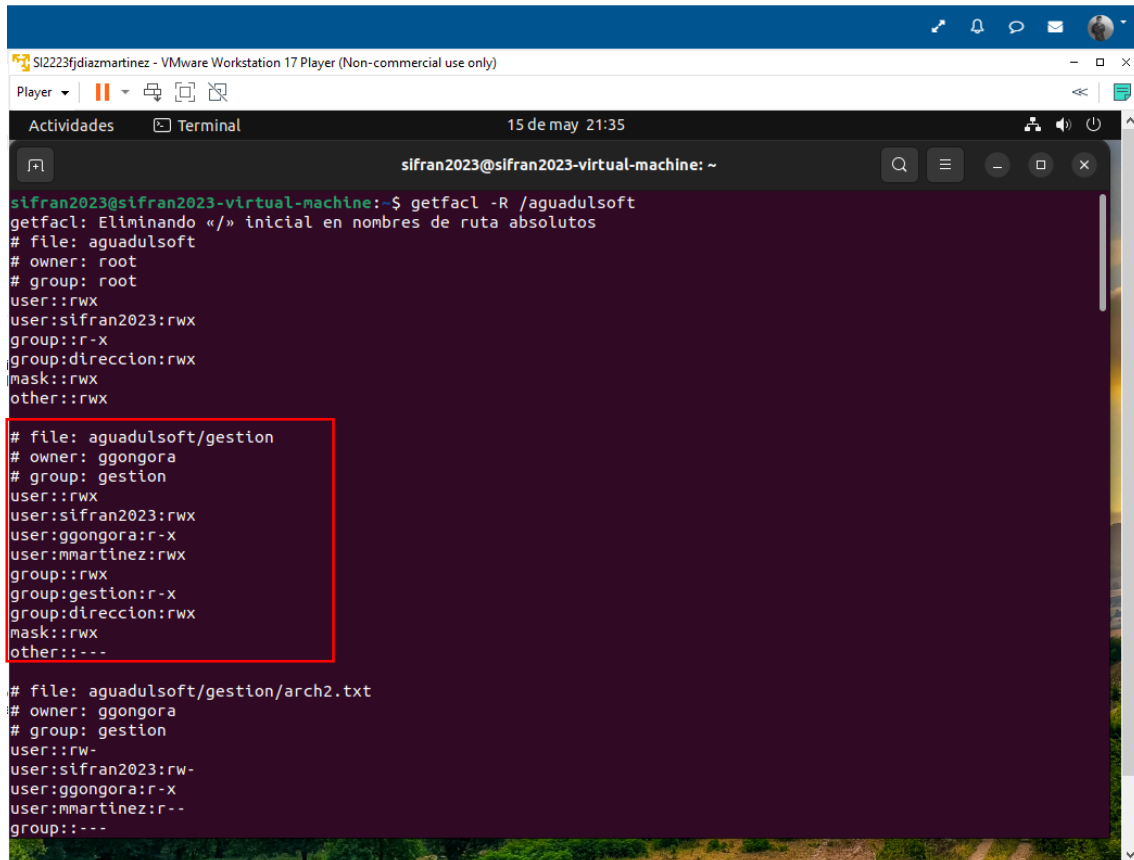


Usando el comando “ls -lR” en la ruta del directorio encontramos como ya todos los directorios tienen asignados sus permisos correspondientes con encargados y grupos (r,w,x)





En este apartado voy a extenderme un poco para poder explicar de forma correcta lo que se indentificar. Además, añadiré un .txt con el contenido completo para que se puedan identificar los demás elementos si se desean.



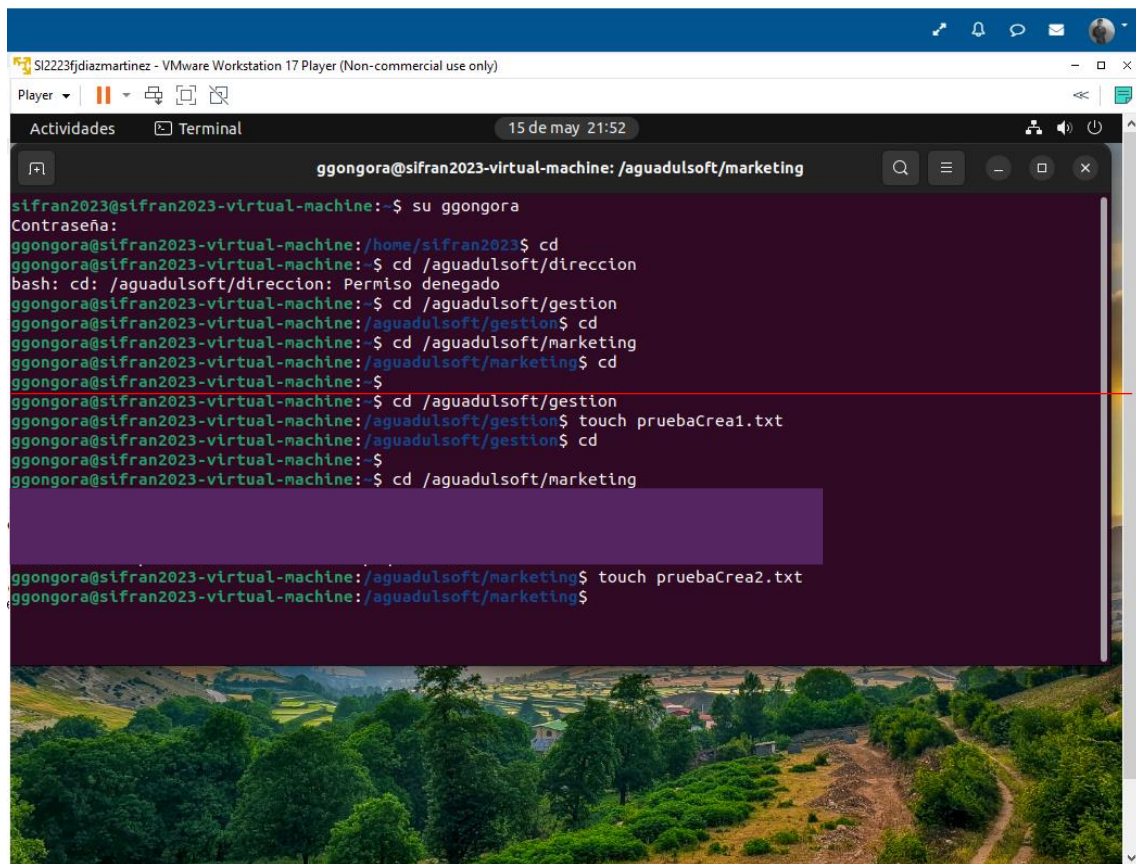
```
sifran2023@sifran2023-virtual-machine: ~  
sifran2023@sifran2023-virtual-machine:~$ getfacl -R /aguadulsoft  
getfacl: Eliminando «/» inicial en nombres de ruta absolutos  
# file: aguadulsoft  
# owner: root  
# group: root  
user::rwx  
user:sifran2023:rwx  
group::r-x  
group:direccion:rwx  
mask::rwx  
other::rwx  
  
# file: aguadulsoft/gestion  
# owner: ggongora  
# group: gestion  
user::rwx  
user:sifran2023:rwx  
user:ggongora:r-x  
user:mmartinez:rwx  
group::rwx  
group:gestion:r-x  
group:direccion:rwx  
mask::rwx  
other::---  
  
# file: aguadulsoft/gestion/arch2.txt  
# owner: ggongora  
# group: gestion  
user::rw-  
user:sifran2023:rw-  
user:ggongora:r-x  
user:mmartinez:r--  
group::---
```

Si analizamos el recuadro rojo veremos como todas las peticiones de la actividad se cumplen.

Nos encontramos en el directorio de gestión, del cual es encargada “ggongora”, podemos observar:

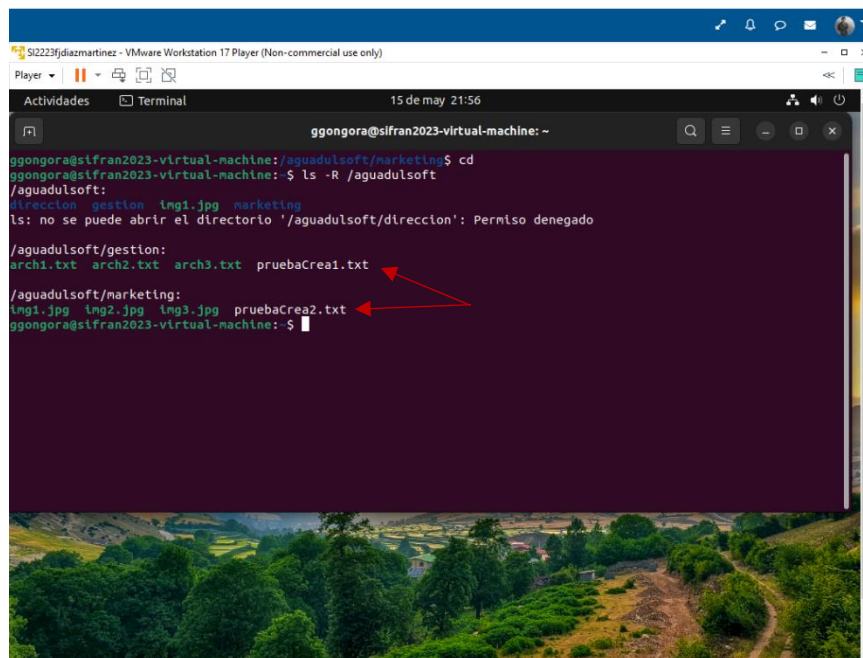
- El grupo pertenece a gestión
- El administrador (sifran2023) tiene todos los permisos
- El encargado tiene permisos de lectura y escritura de los archivos
- El encargado del grupo de marketing tiene permisos RWX del directorio
- El grupo gestión tiene permisos de RW sobre los archivos
- El grupo dirección tiene todos los permisos

También, como se podrá comprobar en el archivo de texto adjunto, sobre el grupo de dirección solo tienen permisos los usuarios que pertenecen a dirección. Así como también no se tienen permisos sobre sus archivos.



```
ggongora@sifran2023-virtual-machine: /aguadulsoft/marketing
sifran2023@sifran2023-virtual-machine:~$ su ggongora
Contraseña:
ggongora@sifran2023-virtual-machine:/home/sifran2023$ cd
ggongora@sifran2023-virtual-machine:~$ cd /aguadulsoft/direccion
bash: cd: /aguadulsoft/direccion: Permiso denegado
ggongora@sifran2023-virtual-machine:~$ cd /aguadulsoft/gestion
ggongora@sifran2023-virtual-machine:/aguadulsoft/gestion$ cd
ggongora@sifran2023-virtual-machine:~$ cd /aguadulsoft/marketing
ggongora@sifran2023-virtual-machine:/aguadulsoft/marketing$ cd
ggongora@sifran2023-virtual-machine:~$ cd /aguadulsoft/gestion
ggongora@sifran2023-virtual-machine:/aguadulsoft/gestion$ touch pruebaCrea1.txt
ggongora@sifran2023-virtual-machine:/aguadulsoft/gestion$ cd
ggongora@sifran2023-virtual-machine:~$ cd /aguadulsoft/marketing
ggongora@sifran2023-virtual-machine:/aguadulsoft/marketing$ touch pruebaCrea2.txt
ggongora@sifran2023-virtual-machine:/aguadulsoft/marketing$
```

Aquí deberemos de separar los dos pasos que hemos seguido. Antes de la línea roja hemos intentado acceder a todos los directorios, y nos ha permitido ingresar a todos menos al de dirección, como era de esperar. Por otro lado, posteriormente hemos ingresado a los directorios en los que los permisos otorgan la posibilidad de crear archivos y los hemos creado.

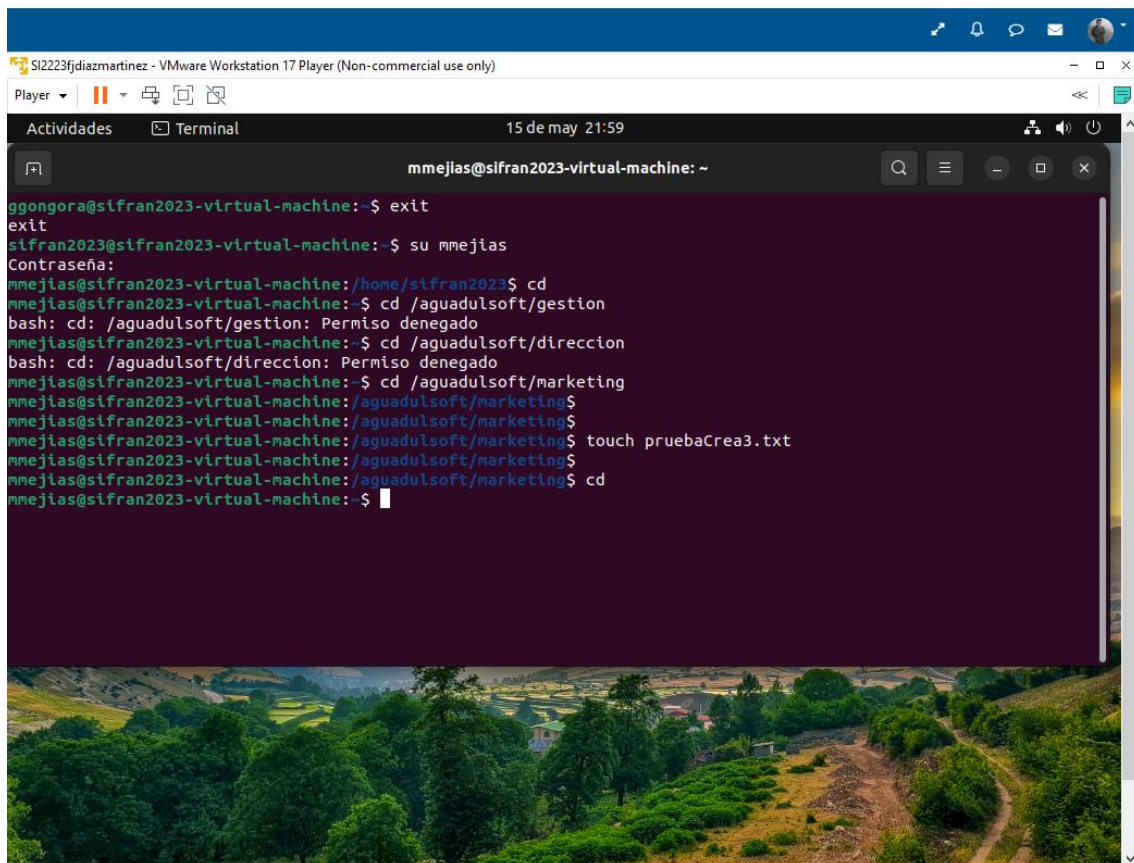


```
ggongora@sifran2023-virtual-machine:~$ cd
ggongora@sifran2023-virtual-machine:/aguadulsoft/marketing$ cd
ggongora@sifran2023-virtual-machine:~$ ls -R /aguadulsoft
/aguadulsoft:
direccion  gestion  img1.jpg  marketing
ls: no se puede abrir el directorio '/aguadulsoft/direccion': Permiso denegado

/aguadulsoft/gestion:
arch1.txt  arch2.txt  arch3.txt  pruebaCrea1.txt

/aguadulsoft/marketing:
img1.jpg  img2.jpg  img3.jpg  pruebaCrea2.txt
ggongora@sifran2023-virtual-machine:~$
```





Aquí repetimos el proceso anterior con un usuario que no es encargado. No tenemos permiso de acceso a los directorios de gestión o dirección (en su caso), por lo tanto no podemos intentar crear un archivo mediante el comando touch, sin embargo si podemos entrar en su departamento y crear un archivo.

```
mmejias@sifran2023-virtual-machine:~$ ls -R /aguadulsoft
/aguadulsoft:
direccion gestion img1.jpg marketing
ls: no se puede abrir el directorio '/aguadulsoft/direccion': Permiso denegado
ls: no se puede abrir el directorio '/aguadulsoft/gestion': Permiso denegado

/aguadulsoft/marketing:
img1.jpg img2.jpg img3.jpg pruebaCrea2.txt pruebaCrea3.txt
mmejias@sifran2023-virtual-machine:~$
```

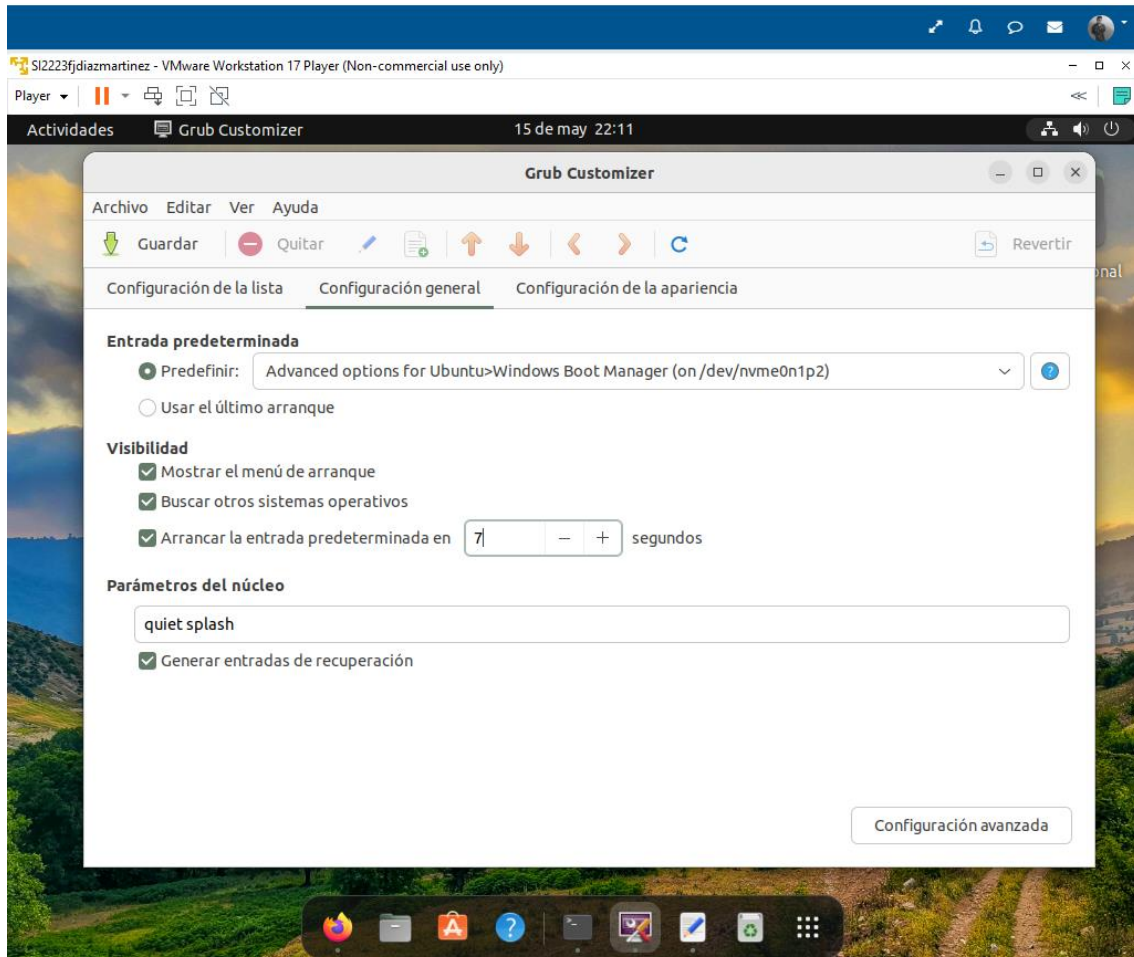
Como este usuario pertenece al grupo de marketing, no tiene acceso a los otros departamentos, por lo tanto no puede ver el contenido, y como se puede observar se ha creado el archivo, pues al estar dentro de su propio departamento tiene permiso de lectura y escritura.



### Actividad 3.- Personalización del gestor de arranque.

Igual que hicimos en trabajos anteriores, en Ubuntu tenemos la posibilidad de configurar el arranque, se puede hacer mediante comandos en el terminal, pero también tenemos una opción más intuitiva mediante una interfaz gráfica con el software “grub-customizer”.

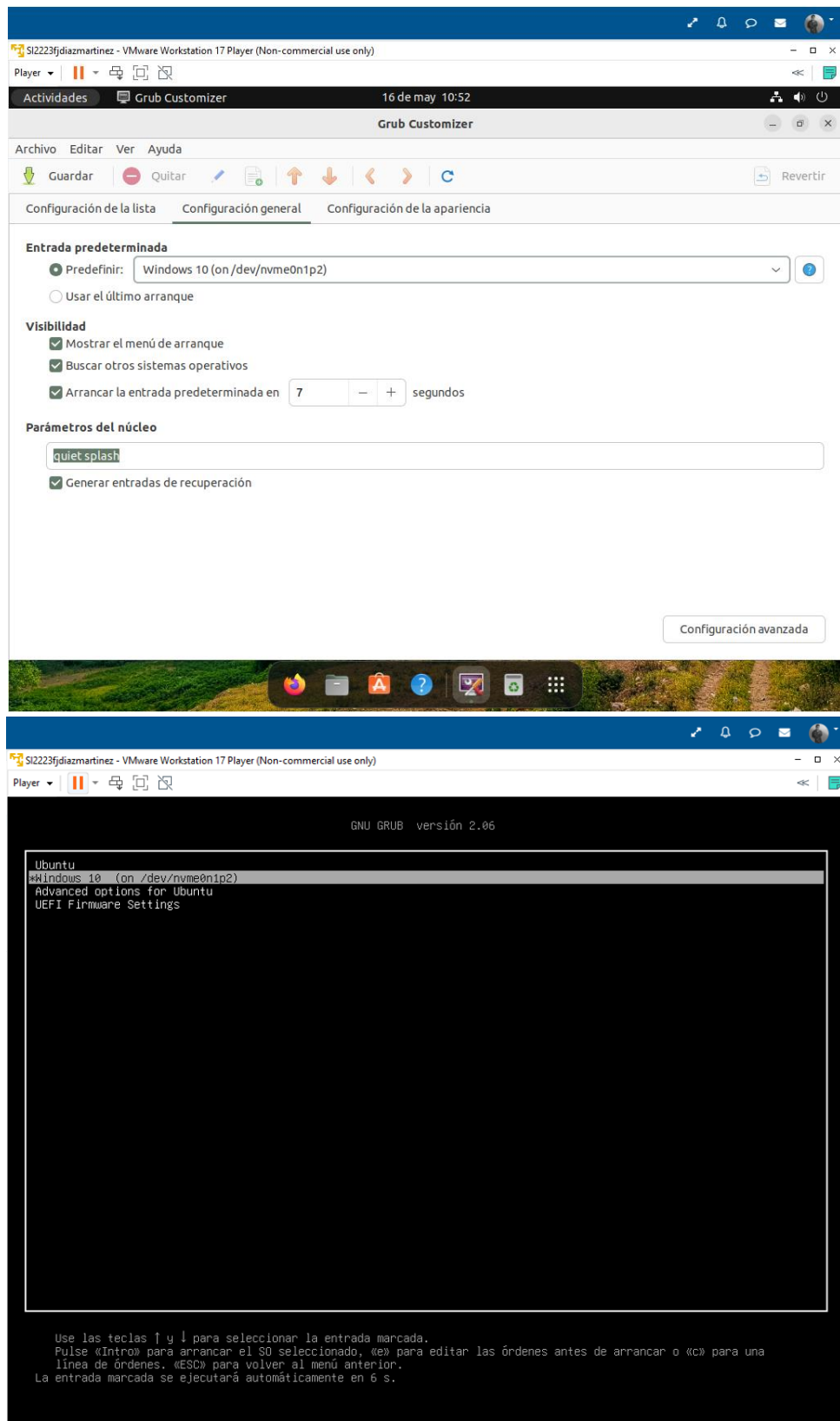
La instalación de este paquete se hace mediante comandos, solo deberemos de actualizar los repositorios para asegurarnos de que contamos con el paquete, para después instalarlo.



En esta primera captura vemos como la configuración de inicio por defecto se puede elegir mediante un desplegable, así como también el tiempo de arranque predeterminado.

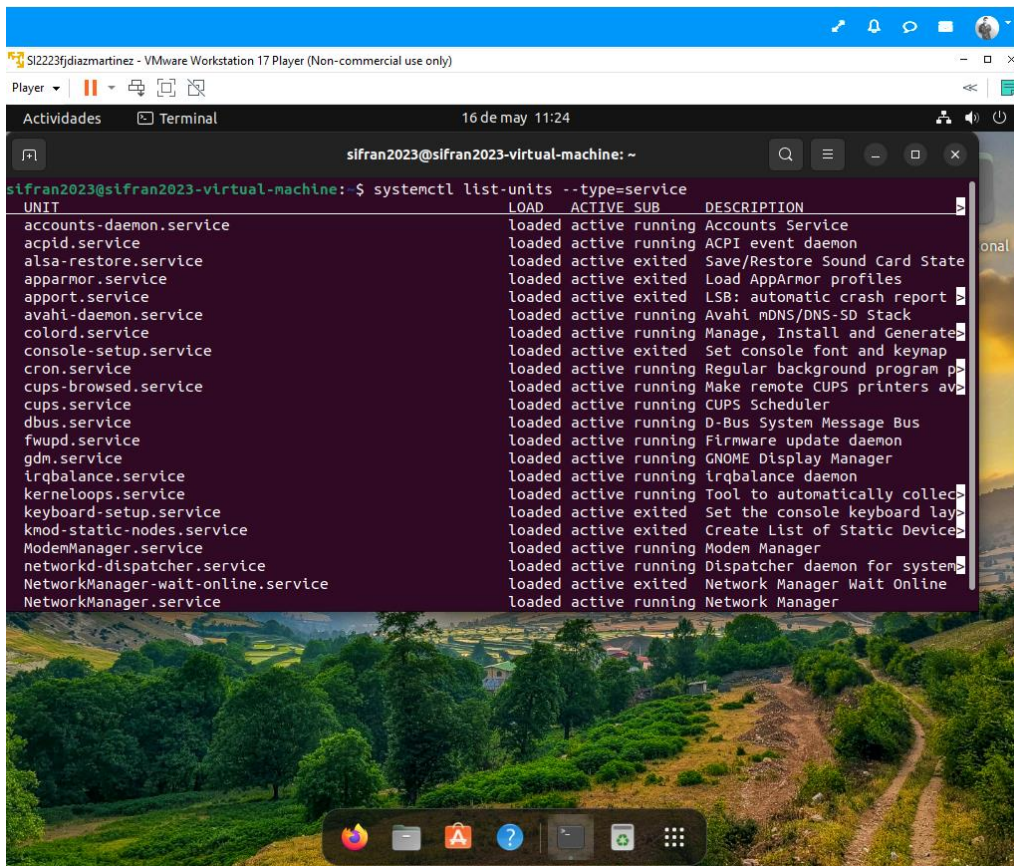
En mi caso, el boot manager de Windows se encuentra dentro del menú contextual “Advanced options for Ubuntu”. Es decir, en el menú de arranque del sistema deberíamos de dirigirnos a esta opción para que accedamos a Windows.

Sin embargo, en el enunciado nos piden que el arranque por defecto sea Windows 10, y que el tiempo de arranque predeterminado sea de 7 segundos si el usuario no escoge otra opción.



Aquí podemos observar los cambios, en el que nos muestra los dos sistemas operativos, Windows 10 es la opción por defecto, y abajo tenemos el contador que nos indica que se ejecutara la entrada en 7 segundos.

### Actividad 4.- Servicios del sistema.



```
sifran2023@sifran2023-virtual-machine: ~  
$ systemctl list-units --type=service  
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  
apparmor.service                   loaded active exited Load AppArmor profiles  
apport.service                     loaded active exited LSB: automatic crash report  
avahi-daemon.service               loaded active running Avahi MDNS/DNS-SD Stack  
colord.service                     loaded active running Manage, Install and Generate  
console-setup.service              loaded active exited Set console font and keymap  
cron.service                       loaded active running Regular background program p  
cups-browsed.service               loaded active running Make remote CUPS printers av  
cups.service                       loaded active running CUPS Scheduler  
dbus.service                       loaded active running D-Bus System Message Bus  
fwupd.service                     loaded active running Firmware update daemon  
gdm.service                        loaded active running GNOME Display Manager  
irqbalance.service                loaded active running irqbalance daemon  
kerneloops.service                loaded active running Tool to automatically collec  
keyboard-setup.service             loaded active exited Set the console keyboard lay  
kmod-static-nodes.service           loaded active exited Create List of Static Device  
ModemManager.service              loaded active running Modem Manager  
networkd-dispatcher.service         loaded active running Dispatcher daemon for system  
NetworkManager-wait-online.service loaded active exited Network Manager Wait Online  
NetworkManager.service             loaded active running Network Manager
```

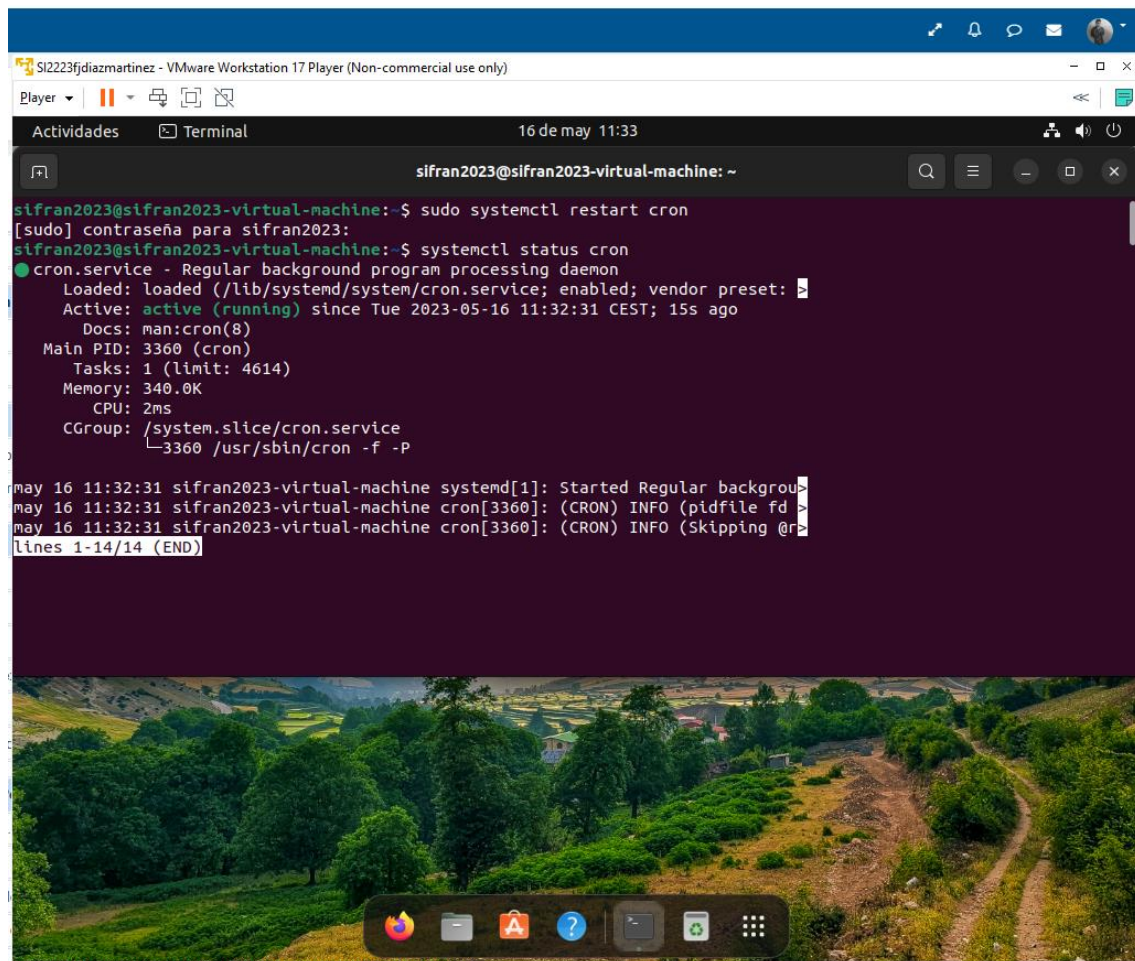
En esta actividad necesitamos encontrar un proceso mediante el terminal e identificar el estado en el que se encuentra para después poder cambiarlo.

En la unidad nos muestran el comando “top” que muestra los procesos del sistema, pero es de forma estática y no nos proporcionan lo que necesitamos.

Es por ello que investigando he llegado a la conclusión de que el mejor comando será el “control de sistema + listado de unidades + filtro por servicio”.

En la captura también podemos ver el servicio que buscamos “cron” y su estado “activado” y ejecutándose.

A continuación, nos piden que, si el proceso se encuentra activo, lo reiniciemos. En mi caso es así, por lo que lo reiniciaré y después mostraré el estado en el que se encuentra el proceso en concreto encauzando mediante terminal.



The screenshot shows a VMware Workstation 17 Player window titled "SI2223fjdíazmartínez - VMware Workstation 17 Player (Non-commercial use only)". Inside the player, there is a terminal window titled "sifran2023@sifran2023-virtual-machine: ~" with the date and time "16 de may 11:33". The terminal displays the following commands and output:

```
sifran2023@sifran2023-virtual-machine:~$ sudo systemctl restart cron
[sudo] contraseña para sifran2023:
sifran2023@sifran2023-virtual-machine:~$ systemctl status cron
● cron.service - Regular background program processing daemon
   Loaded: loaded (/lib/systemd/system/cron.service; enabled; vendor preset: 
   Active: active (running) since Tue 2023-05-16 11:32:31 CEST; 15s ago
     Docs: man:cron(8)
    Main PID: 3360 (cron)
      Tasks: 1 (limit: 4614)
     Memory: 340.0K
        CPU: 2ms
    CGroup: /system.slice/cron.service
            └─3360 /usr/sbin/cron -f -P

may 16 11:32:31 sifran2023-virtual-machine systemd[1]: Started Regular backgrou
may 16 11:32:31 sifran2023-virtual-machine cron[3360]: (CRON) INFO (pidfile fd >
may 16 11:32:31 sifran2023-virtual-machine cron[3360]: (CRON) INFO (Skipping @r
lines 1-14/14 (END)
```

The desktop background shows a scenic view of a valley with green fields, trees, and a dirt road. The bottom of the screen features a dock with icons for Firefox, Files, Applications, Help, Settings, and a Dash icon.

Hemos hecho uso de:

- “Sudo systemctl restart cron” para apuntar al proceso con el que estamos trabajando y reiniciarlo
- “Systemctl status cron” para comprobar el estado en el que se encuentra el proceso actualmente.

Se observa que está activo y en ejecución. Se muestra información adicional, como su descripción, su ubicación y el tiempo que lleva ejecutándose.

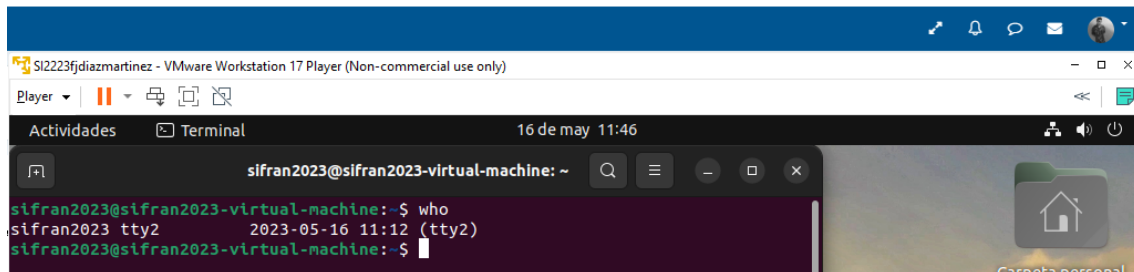


### Actividad 5.- Monitorización del sistema.

Para la monitorización del sistema contamos con multitud de comandos, yo voy a analizar tres de los más significativos en relación con lo que hemos trabajado.

- **who**

Este comando nos muestra los usuarios que están actualmente conectados con el equipo.



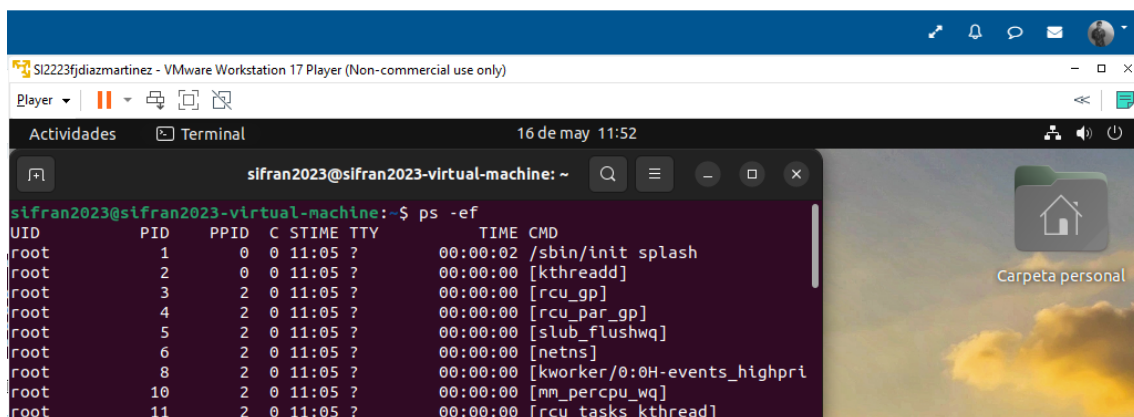
```
sifran2023@sifran2023-virtual-machine: ~  
sifran2023@sifran2023-virtual-machine:~$ who  
sifran2023 tty2          2023-05-16 11:12 (tty2)  
sifran2023@sifran2023-virtual-machine:~$
```

Las diferentes columnas muestran lo siguiente si las interpretamos de izquierda a derecha:

- nombre del usuario
- la consola asignada
- la fecha de autenticación
- la hora de inicio de sesión

- **ps**

Por otro lado, este comando nos muestra los procesos que se han lanzado y qué usuario ha sido el responsable, la información mostrada es dinámica, así que se actualiza según se lanzan nuevos procesos. Para ello debemos de añadir -ef a la secuencia.

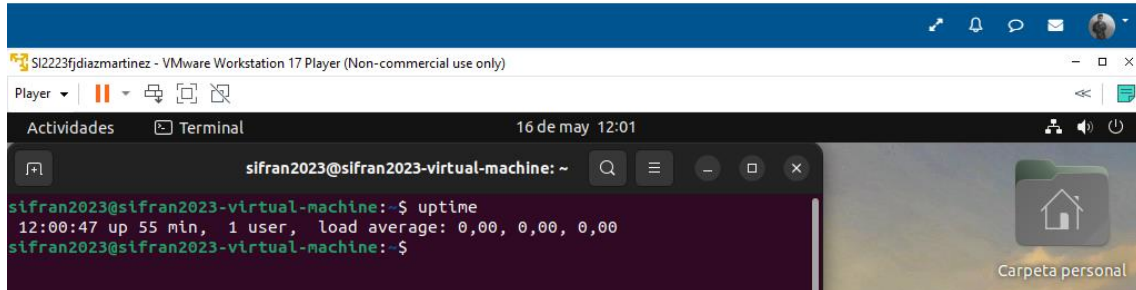


```
sifran2023@sifran2023-virtual-machine:~$ ps -ef  
UID          PID    PPID  C   STIME TTY          TIME CMD  
root           1        0  0  11:05 ?        00:00:02 /sbin/init splash  
root           2        0  0  11:05 ?        00:00:00 [kthreadd]  
root           3        2  0  11:05 ?        00:00:00 [rcu_gp]  
root           4        2  0  11:05 ?        00:00:00 [rcu_par_gp]  
root           5        2  0  11:05 ?        00:00:00 [slub_flushwq]  
root           6        2  0  11:05 ?        00:00:00 [netns]  
root           8        2  0  11:05 ?        00:00:00 [kworker/0:0H-events_highpri]  
root          10        2  0  11:05 ?        00:00:00 [mm_percpu_wq]  
root          11        2  0  11:05 ?        00:00:00 [rcu_tasks_kthread]
```

Observamos quien lanza el programa, las ID del proceso padre, tiempo en ejecución, nombre del terminal asociado al proceso y comando.

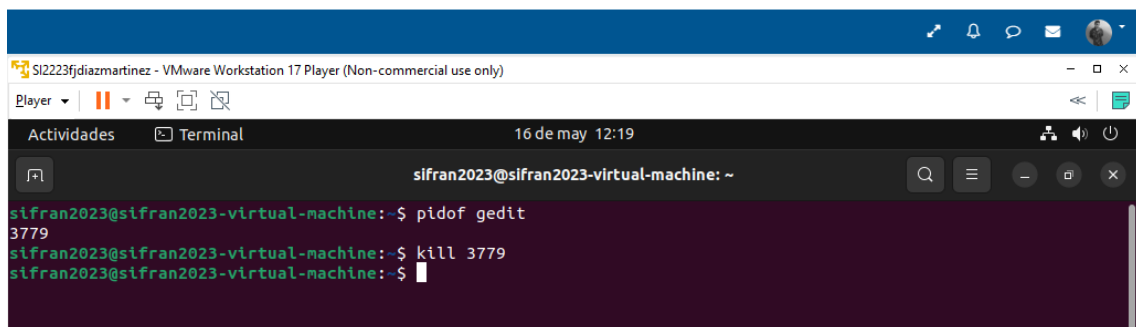
- **uptime**

Este comando muestra información sobre la actividad y carga del sistema. Tiene elementos como la hora del sistema, el tiempo que ha estado operativo, el número de usuarios que están conectados así como una media de la carga en 3 lapsos de tiempo (1,5,10 últimos minutos).



```
sifran2023@si2223fdiazmartinez - VMware Workstation 17 Player (Non-commercial use only)
Actividades Terminal 16 de may 12:01
sifran2023@si2223-virtual-machine: ~
sifran2023@si2223-virtual-machine:~$ uptime
12:00:47 up 55 min, 1 user, load average: 0,00, 0,00, 0,00
sifran2023@si2223-virtual-machine:~$
```

Para terminar, nos solicitan que abramos un editor de textos, localizamos su PID y posteriormente finalizamos el proceso, todo ello usando el terminal.



```
sifran2023@si2223-virtual-machine: ~
sifran2023@si2223-virtual-machine:~$ pidof gedit
3779
sifran2023@si2223-virtual-machine:~$ kill 3779
sifran2023@si2223-virtual-machine:~$
```

Hay un comando muy apropiado para este propósito, el del “pidof” que nos muestra el PID del proceso que indiquemos. Ç

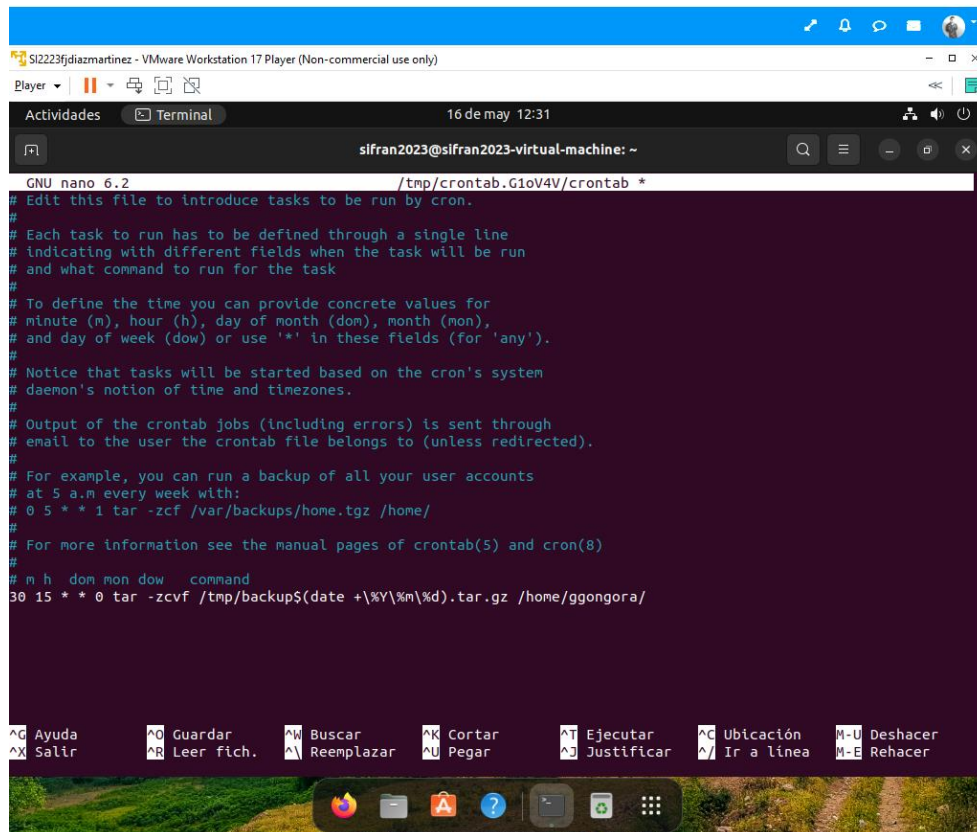
La adición de “gedit” hace que apunte hacia el editor de textos que viene por defecto instalado en Ubuntu.

Una vez hecho esto, nos devuelve una cifra “3779” la cual es el PID del editor que acabamos de abrir.

Mediante el comando “kill” indicamos el PID y así finalizamos el proceso que habíamos abierto anteriormente.

## Actividad 6.- Copia de seguridad y programación de tareas.

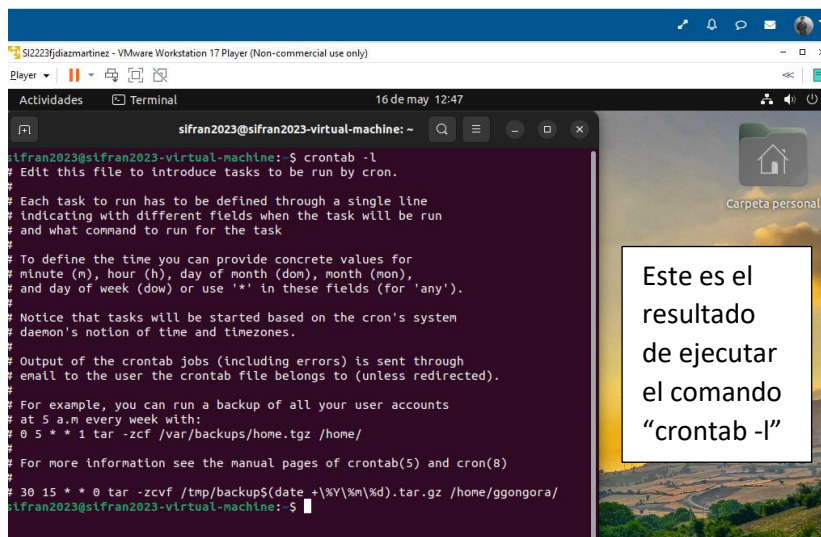
Para terminar, debemos, mediante terminal configurar una copia de seguridad cada domingo a las 15:00. Necesitaremos crear un archivo con “crontab -e” introducir la línea con la información que precisamos y posteriormente con “crontab -l” visualizaremos el contenido del archivo.



```
GNU nano 6.2 /tmp/crontab.G1oV4V/crontab *
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
30 15 * * 0 tar -zcvf /tmp/backup$(date +%Y%m%d).tar.gz /home/ggongora/
```

Este es el resultado escribir el comando “crontab -e” y seleccionar un editor de texto.

Al final del documento hemos escrito la sentencia que nos pide la actividad, dónde la primera parte se refiere a la hora y fecha, la segunda (-zcvf) las acciones que se van a realizar, la ruta dónde se guardará el archivo, con el correspondiente formato para que no elimine archivos previos y el directorio del que se quiere realizar la copia de seguridad.



```
sifran2023@si2223fjdiazmartinez:~$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
30 15 * * 0 tar -zcvf /tmp/backup$(date +%Y%m%d).tar.gz /home/ggongora/
sifran2023@si2223fjdiazmartinez:~$
```

Resultados de la ACL completos:

```
# file: aguadulsoft
# owner: root
# group: root
user::rwx
user:sifran2023:rwx
group::r-x
group:direccion:rwx
mask::rwx
other::rwx

# file: aguadulsoft/gestion
# owner: ggongora
# group: gestion
user::rwx
user:sifran2023:rwx
user:ggongora:r-x
user:mmartinez:rwx
group::rwx
group:gestion:r-x
group:direccion:rwx
mask::rwx
other::---

# file: aguadulsoft/gestion/arch2.txt
# owner: ggongora
# group: gestion
user::rw-
user:sifran2023:rw-
user:ggongora:r-x
user:mmartinez:r--
group::---
group:gestion:r-x
group:direccion:rwx
mask::rwx
other::---

# file: aguadulsoft/gestion/arch1.txt
# owner: ggongora
# group: gestion
user::rw-
user:sifran2023:rw-
user:ggongora:r-x
user:mmartinez:r--
group::---
group:gestion:r-x
group:direccion:rwx
mask::rwx
other::---

# file: aguadulsoft/gestion/arch3.txt
# owner: ggongora
# group: gestion
user::rw-
user:sifran2023:rw-
user:ggongora:r-x
```



```
user:mmartinez:r--
group:---
group:gestion:r-x
group:direccion:rwX
mask::rwX
other:---

# file: aguadulsoft/img1.jpg
# owner: sifran2023
# group: sifran2023
user::rwX
group::rwX
other::rwX

# file: aguadulsoft/marketing
# owner: mmartinez
# group: marketing
user::rwX
user:sifran2023:rwX
user:ggongora:rwX
user:mmartinez:r-x
group::rwX
group:marketing:r-x
group:direccion:rwX
mask::rwX
other:---

# file: aguadulsoft/marketing/img3.jpg
# owner: mmartinez
# group: marketing
user::rw-
user:sifran2023:rw-
user:ggongora:r--
user:mmartinez:r-x
group:---
group:marketing:r-x
group:direccion:rwX
mask::rwX
other:---

# file: aguadulsoft/marketing/img1.jpg
# owner: mmartinez
# group: marketing
user::rw-
user:sifran2023:rw-
user:ggongora:r--
user:mmartinez:r-x
group:---
group:marketing:r-x
group:direccion:rwX
mask::rwX
other:---

# file: aguadulsoft/marketing/img2.jpg
# owner: mmartinez
# group: marketing
user::rw-
```

```
user:sifran2023:rw-
user:ggongora:r--
user:mmartinez:r-x
group:---
group:marketing:r-x
group:direccion:rwX
mask::rwX
other:---

# file: aguadulsoft/direccion
# owner: sifran2023
# group: direccion
user::rwX
user:sifran2023:rwX
user:ggongora:---
user:mmartinez:---
group::rwX
group:direccion:r-x
mask::rwX
other:---

# file: aguadulsoft/direccion/vid3.wmv
# owner: sifran2023
# group: direccion
user::rw-
user:sifran2023:rwX
user:ggongora:---
user:mmartinez:---
group:---
group:direccion:r-x
mask::rwX
other:---

# file: aguadulsoft/direccion/vid1.wmv
# owner: sifran2023
# group: direccion
user::rw-
user:sifran2023:rwX
user:ggongora:---
user:mmartinez:---
group:---
group:direccion:r-x
mask::rwX
other:---

# file: aguadulsoft/direccion/vid2.wmv
# owner: sifran2023
# group: direccion
user::rw-
user:sifran2023:rwX
user:ggongora:---
user:mmartinez:---
group:---
group:direccion:r-x
mask::rwX
other:---
```

