

TAREA UNIDAD 8: Acciones de administración básicas de Linux

Alumno: Matias Pennino

Ciclo: 1º DAW 2023 - 2024

Indice:

Actividad 1.- Cuentas de usuarios y grupos locales.

Usando **comandos en un terminal**, crea los siguientes usuarios y grupos:

- Administración:
 - Ana Alonso (encargada).
 - Alejandro Ariza.
- Ventas:
 - Vicente Vázquez (encargado).
 - Vega Vidal.
- Dirección:
 - Tu propio usuario (ya creado durante la instalación de Ubuntu, eres el director o directora).

Nombra a los usuarios con la **inicial del nombre y el primer apellido**, por ejemplo, Vicente Vázquez sería vvazquez (con minúsculas y sin usar tildes ni otros caracteres conflictivos). Guarda al crearlos sus nombres y apellidos completos, y no modifiques el grupo principal de cada usuario (déjalo tal como se crea por defecto: un grupo con el mismo nombre del usuario).

Después de crear los usuarios, crea los grupos de los departamentos (minúsculas y sin caracteres conflictivos) e incluye a los usuarios dentro de los grupos tal como se muestra. Se deben añadir los usuarios a los grupos como grupos secundarios, pero no se deben modificar los grupos principales de los usuarios.

Las **capturas** deben mostrar, al menos:

- Comandos usados para la creación/modificación de los usuarios, junto con la salida obtenida (comando recomendado: adduser).
- Contenido del archivo `"/etc/passwd"` donde se vean claramente todos los usuarios creados/modificados (comando recomendado: cat). Puedes encauzar la salida del comando `"cat"` con `"tail -n"`, siendo n el número de líneas a visualizar.
- Comandos usados para la creación de los grupos (comando recomendado: addgroup).
- Comandos usados para la inclusión de los usuarios en su grupo correspondiente, siempre que no se hubiera incluido previamente en el comando de creación de usuario (comando recomendado: adduser).
- Contenido del archivo `"/etc/group"` donde se visualicen claramente los grupos creados y sus miembros.

Actividad 2.- Estructura departamental y permisos de acceso.

Usando **comandos en un terminal**, crea la siguiente estructura departamental mediante directorios, es decir, un directorio por cada departamento. Crea primero el directorio `"aguadulsoft"` y, dentro de éste, los demás. Introduce al menos **tres ficheros** en cada directorio (documentos, imágenes, vídeos.... Consejo: Con el comando `"touch"` se pueden crear ficheros vacíos). Sitúa la estructura de directorios colgando directamente del directorio raíz (directorio `"/"`) del sistema (se recomienda usar nombres sin mayúsculas y sin tildes).

/

└─aguadulsoft
 |─administracion
 |─ventas
 └─direccion

A continuación configura los permisos de acceso y la pertenencia de cada directorio y los ficheros que contienen de la siguiente manera:

- **Pertenencia** de los directorios y todos los ficheros que contienen:
 - Administración: Pertenece a la encargada de administración y al grupo de administración.
 - Ventas: Pertenece al encargado de ventas y al grupo de ventas.
 - Dirección: Pertenece a tu usuario y al grupo de dirección.
- **Permisos básicos** de lectura, escritura y ejecución:
 - Cada grupo y usuario dueños (encargado o encargada) tiene permisos de lectura, escritura y ejecución sobre su directorio; y permisos de lectura y escritura sobre los ficheros en su interior.
 - Los que no son miembros de un departamento no tienen ningún permiso sobre los directorios ni ficheros de los demás departamentos.
- **Permisos avanzados** con listas de control de acceso (ACL):
 - El director o directora tiene permisos de lectura, escritura y ejecución sobre todos los directorios; y de lectura y escritura sobre todos los ficheros que contienen.
 - Los encargados o encargadas de cada departamento tienen permisos de lectura y ejecución sobre los demás directorios; y permiso de lectura sobre los ficheros de estos; pero no tienen ningún permiso para el directorio de dirección ni sus contenidos.
- **Comprobaciones:**
 - Entra en el sistema con un usuario encargado o encargada de algún departamento (usando **comandos de terminal**).
 - Intenta acceder al directorio de: 1.- Dirección; 2.- Su propio departamento; 3.- Otro departamento. Comenta los resultados.
 - Intenta crear un fichero en el interior de cada directorio al que pueda entrar (por ejemplo, usando el comando "touch"). Comenta los resultados.
 - Realiza las mismas comprobaciones para un usuario que NO sea encargado o encargada de ningún departamento.

Las **capturas** deben mostrar, al menos:

- Comandos usados para la creación de uno de los directorios y los ficheros que contiene.
- Muestra de la estructura departamental de directorios con sus ficheros correspondientes y su ubicación.
- Comandos usados para la asignación de propietarios a directorios y ficheros.
- Comandos usados para la asignación de permisos básicos a directorios y ficheros.
- Contenido del directorio "aguadulsoft" con el formato largo, y de forma recursiva para ver los permisos de todos los ficheros que contiene.

- Comandos usados para la asignación de permisos avanzados a directorios y ficheros.
- Muestra de las ACL de los directorios y ficheros.
- Inicio de sesión con usuario encargado, mostrando intentos de acceso y de creación de fichero en cada uno de los 3 subdirectorios.
- Inicio de sesión con usuario no encargado, mostrando intentos de acceso y de creación de fichero en cada uno de los 3 subdirectorios.

Actividad 3.- Personalización del gestor de arranque.

Utilizando el programa "**grub-customizer**" (no es necesario hacerlo con comandos, pero este programa se debe instalar previamente ya que no viene en los repositorios oficiales de Ubuntu 22.04), configura el gestor de arranque GRUB de la forma siguiente:

- Establece que arranque por defecto Windows, si transcurrido el tiempo de espera predefinido el usuario no ha tecleado nada.
- Cambia el tiempo de espera para elegir el sistema a iniciar, con tiempo de arranque automático a los 10 segundos.

Las **capturas** deben mostrar, al menos:

- Modificación de los parámetros que se piden en grub-customizer.
- Muestra de que todos los cambios anteriores han sido aplicados con el GRUB en ejecución.

Actividad 4.- Servicios del sistema.

Usando **comandos en un terminal**, muestra todos servicios del sistema e indica el estado en el que se encuentra el servicio "**cron**".

A continuación, si el servicio está activo, deberás reiniciarlo, y si está inactivo, deberás activarlo. Cuando termines la actividad, el servicio debe quedar activado.

Las **capturas** deben mostrar, al menos:

- Comando usado para mostrar el listado con todos los servicios del sistema.
- Muestra del estado inicial del servicio "cron".
- Comando usado para iniciar/reiniciar el servicio "cron".
- Muestra del estado final del servicio "cron".

Actividad 5.- Monitorización del sistema.

Utiliza los diferentes **comandos** vistos en la unidad para monitorizar al menos tres elementos del sistema, explicando detalladamente la salida de cada comando.

A continuación, abre un editor de textos y halla el PID de su proceso. Una vez encontrado, utiliza un **comando** para eliminar el proceso asociado al editor de texto.

Las **capturas** deben mostrar, al menos:

- Ejecución de cada uno de los comandos y salida producida.
- Comandos usados para buscar el PID del proceso y eliminarlo.

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

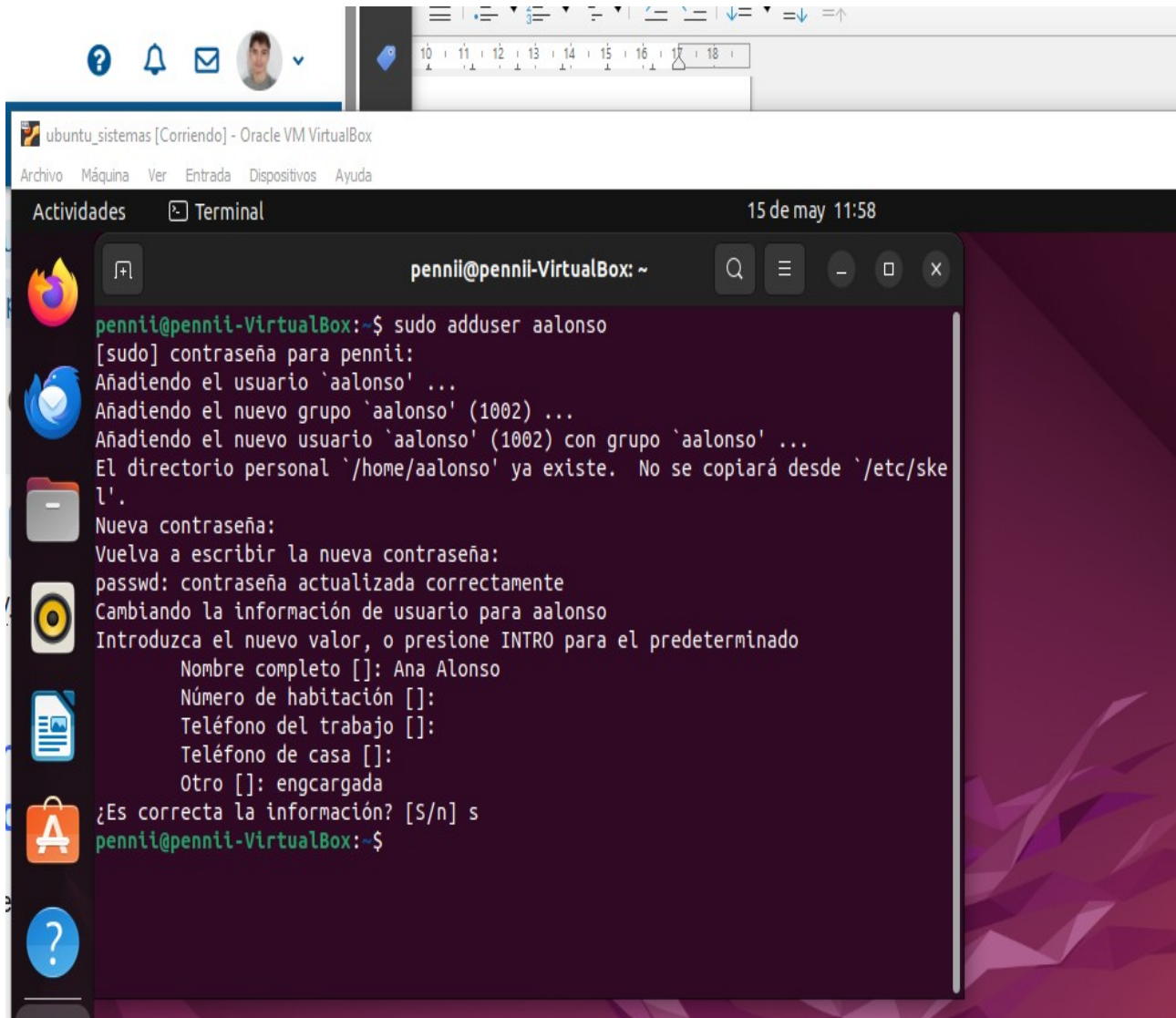
Haz login con el usuario Ana Alonso y, haciendo uso del editor de programaciones "crontab" (asegúrate de que el servicio "cron" está ejecutándose), programa una copia de seguridad de su directorio de trabajo. La copia se almacenará dentro el directorio "/tmp" con el nombre **backup<FECHA>.tar.gz** para ejecutarse **cada sábado a las 23:00h.**

No interesa "machacar" los archivos de copias de seguridad antiguas, por lo que el fichero destino debe contener la fecha del día en que se realizó la copia. Para ello el fichero debe nombrarse de la forma **backup\$(date +%Y%m%d).tar.gz**

Las **capturas** deben mostrar, al menos:

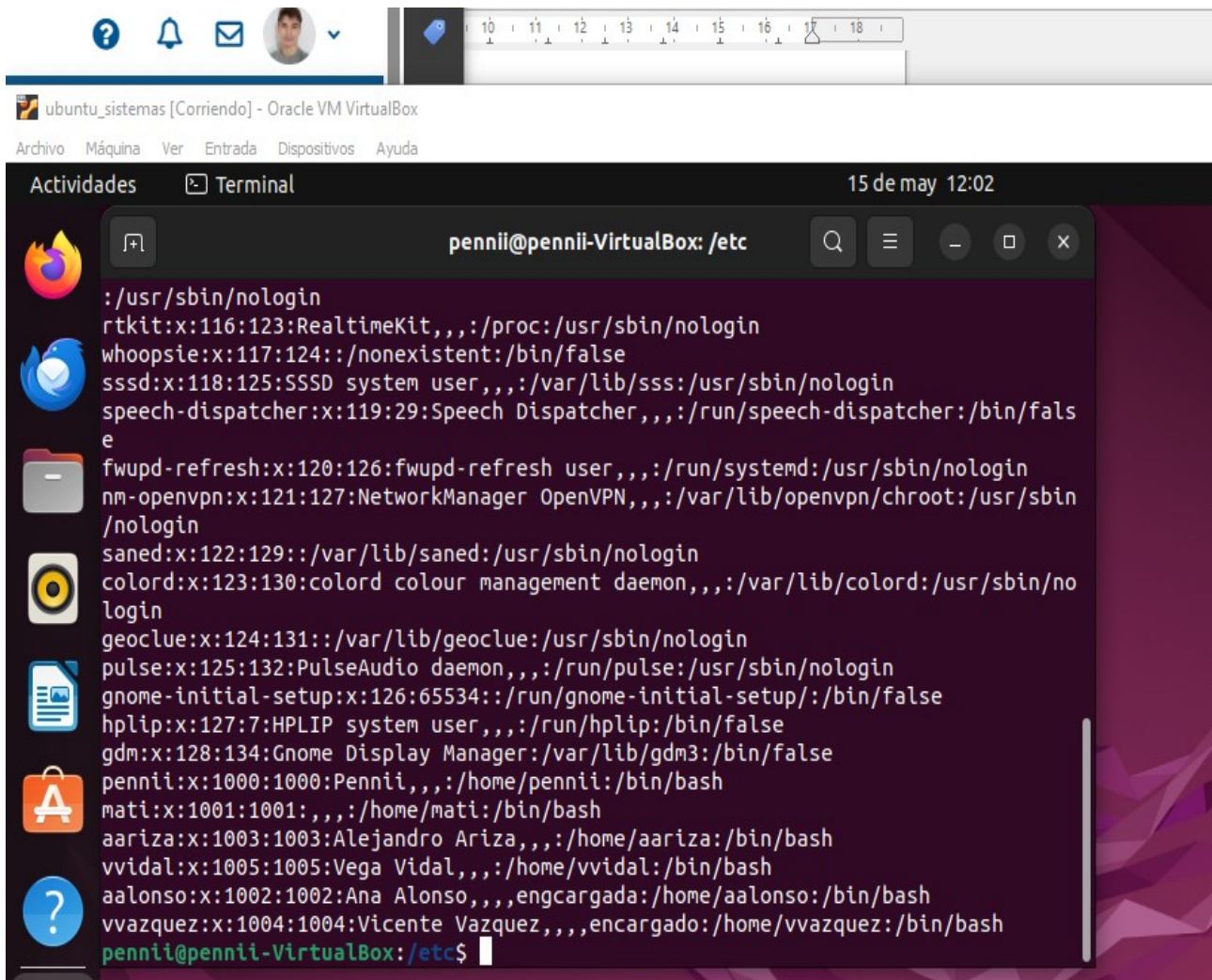
- Contenido del fichero "crontab" con la programación realizada.
- Muestra de la salida del comando "crontab -l".

Actividad 1.



```
pennii@pennii-VirtualBox: ~  
$ sudo adduser aalonso  
[sudo] contraseña para pennii:  
Añadiendo el usuario 'aalonso' ...  
Añadiendo el nuevo grupo 'aalonso' (1002) ...  
Añadiendo el nuevo usuario 'aalonso' (1002) con grupo 'aalonso' ...  
El directorio personal '/home/aalonso' ya existe. No se copiará desde '/etc/skel'.  
Nueva contraseña:  
Vuelva a escribir la nueva contraseña:  
passwd: contraseña actualizada correctamente  
Cambiando la información de usuario para aalonso  
Introduzca el nuevo valor, o presione INTRO para el predeterminado  
Nombre completo []: Ana Alonso  
Número de habitación []:  
Teléfono del trabajo []:  
Teléfono de casa []:  
Otro []: encargada  
¿Es correcta la información? [S/n] s  
pennii@pennii-VirtualBox:~$
```

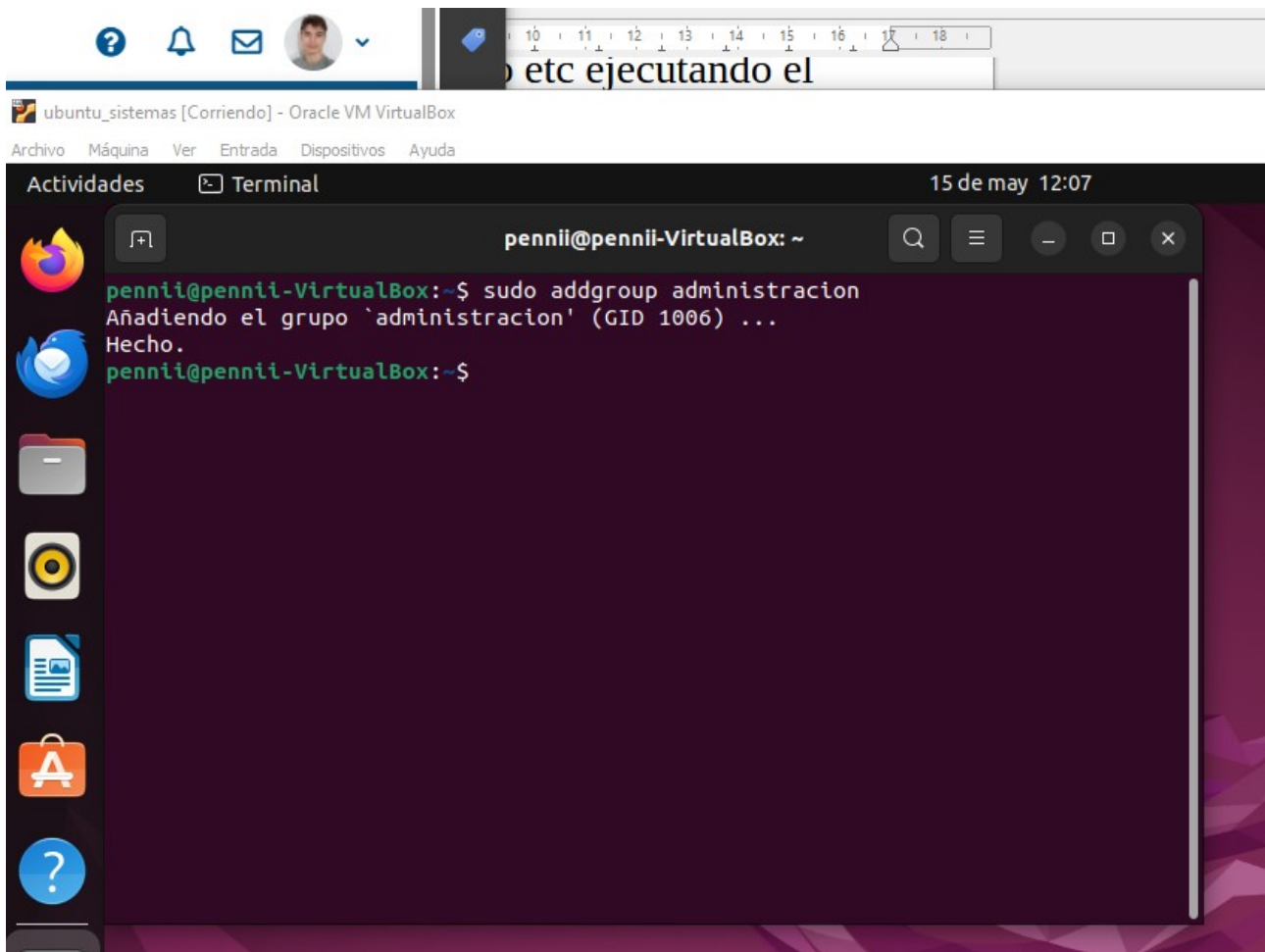
Usamos el comando adduser y agregamos un nuevo usuario, en la captura se ve a Ana Alonso, y el procedimiento sera el mismo para cada usuario. Como contraseña le pondremos a todos “admin2324”



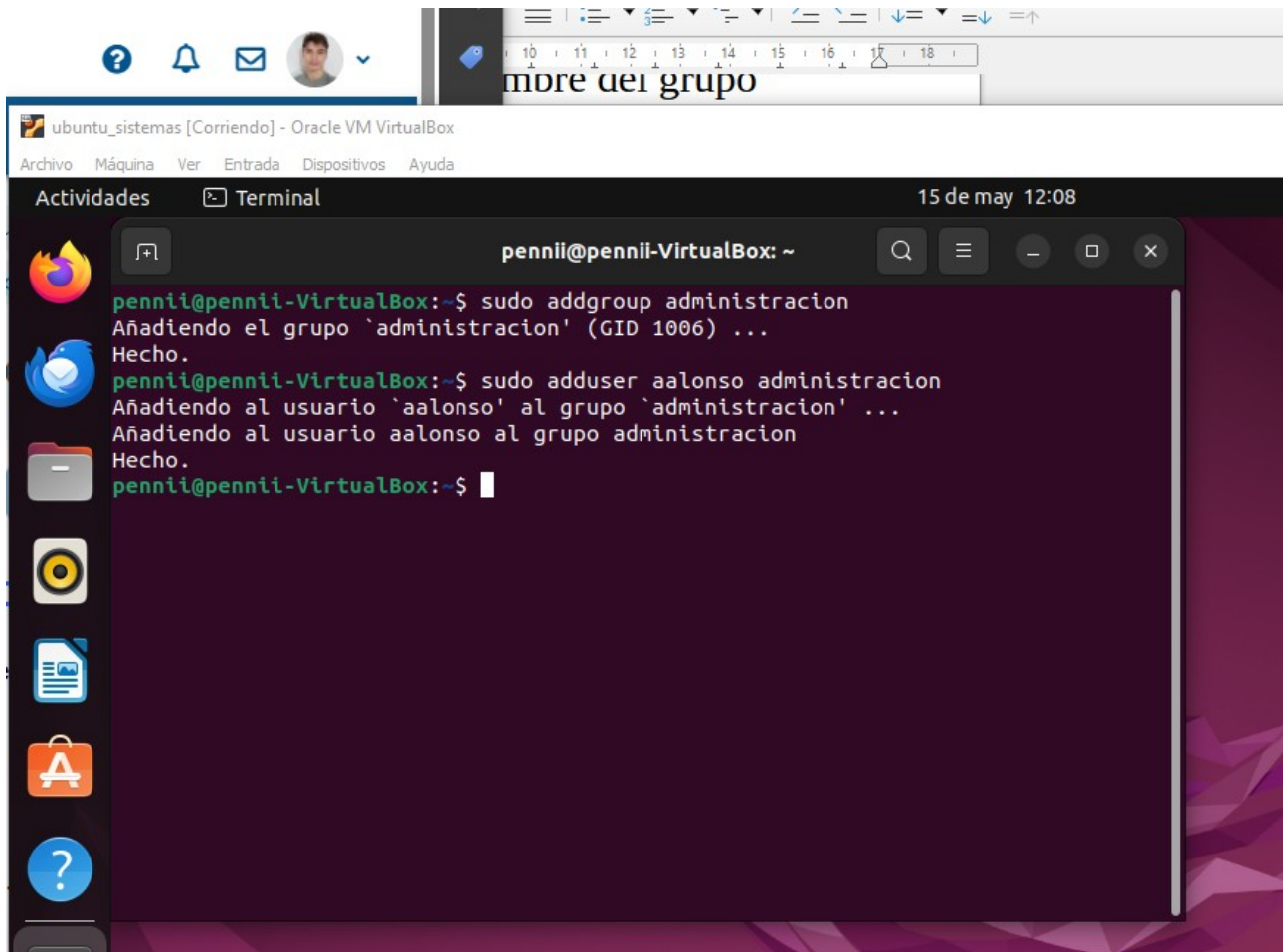
The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: /etc" with a search bar and window controls. The terminal displays the output of the `cat /etc/passwd` command, listing system users and regular users. The system users include `rtkit`, `whoopsie`, `sssd`, `speech-dispatcher`, `fwupd-refresh`, `nm-openvpn`, `saned`, `colord`, `geoclue`, `pulse`, `gnome-initial-setup`, `hplip`, `gdm`, and regular users `pennii`, `mati`, `aariza`, `vvidal`, `aalonso`, and `vvazquez`. Each entry shows the username, UID, GID, name, and shell path.

```
pennii@pennii-VirtualBox: /etc$ cat /etc/passwd
:/usr/sbin/nologin
rtkit:x:116:123:RealtimeKit,,,:/proc:/usr/sbin/nologin
whoopsie:x:117:124:/:nonexistent:/bin/false
sssd:x:118:125:SSSD system user,,,:/var/lib/sss:/usr/sbin/nologin
speech-dispatcher:x:119:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
fwupd-refresh:x:120:126:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
nm-openvpn:x:121:127:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
saned:x:122:129:/:/var/lib/saned:/usr/sbin/nologin
colord:x:123:130:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:124:131:/:/var/lib/geoclue:/usr/sbin/nologin
pulse:x:125:132:PulseAudio daemon,,,:/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:126:65534:/:/run/gnome-initial-setup:/bin/false
hplip:x:127:7:HPLIP system user,,,:/run/hplip:/bin/false
gdm:x:128:134:Gnome Display Manager:/var/lib/gdm3:/bin/false
pennii:x:1000:1000:Pennii,,,:/home/pennii:/bin/bash
mati:x:1001:1001:,,,:/home/mati:/bin/bash
aariza:x:1003:1003:Alejandro Ariza,,,:/home/aariza:/bin/bash
vvidal:x:1005:1005:Vega Vidal,,,:/home/vvidal:/bin/bash
aalonso:x:1002:1002:Ana Alonso,,,:engcargada:/home/aalonso:/bin/bash
vvazquez:x:1004:1004:Vicente Vazquez,,,:engcargado:/home/vvazquez:/bin/bash
pennii@pennii-VirtualBox: /etc$
```

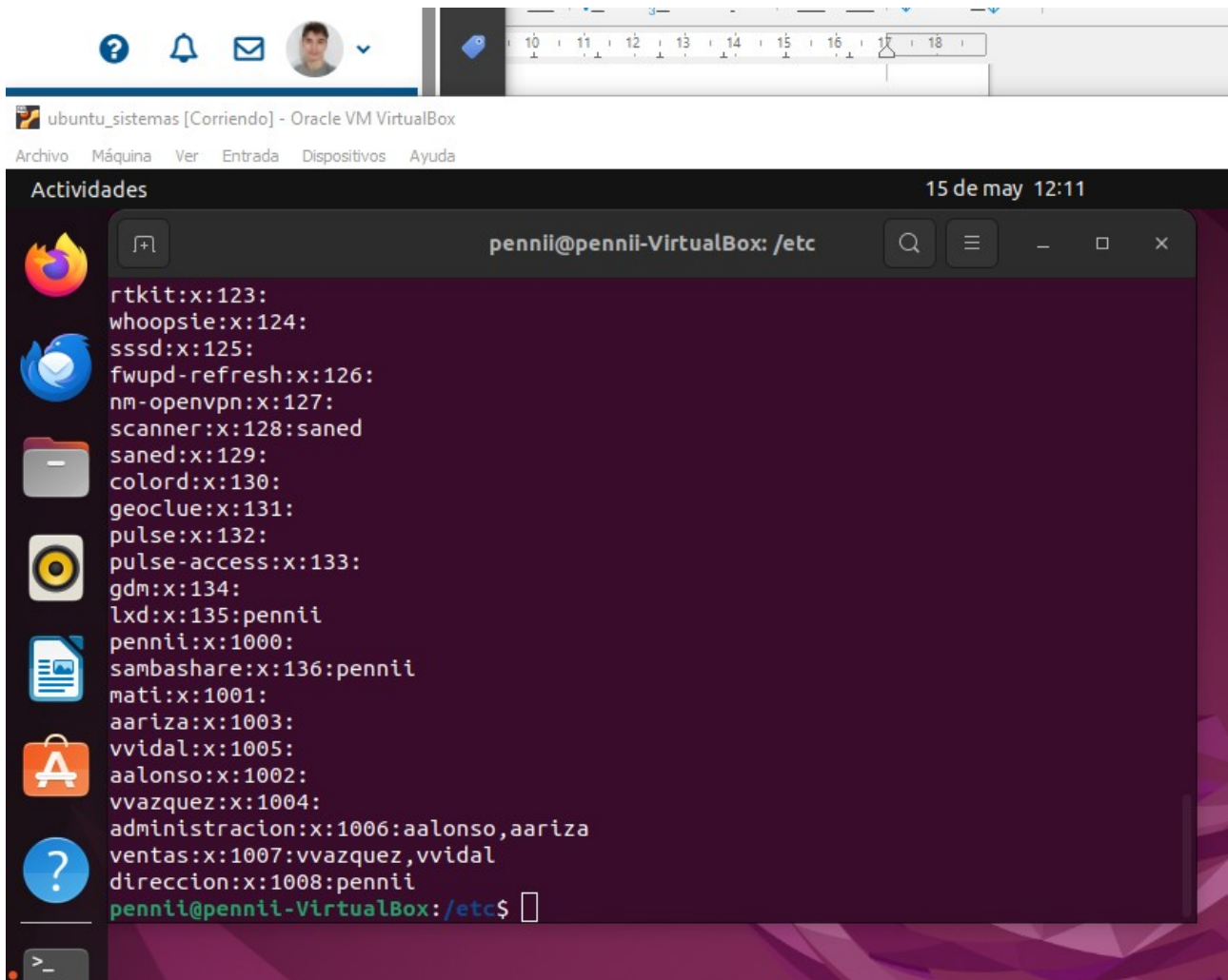
Primero iremos al directorio `etc` ejecutando el comando `cd ../../etc` y luego ejecutaremos `cat passwd` para ver el contenido del fichero



Crearemos los grupos usando el comando **sudo addgroup n**, siendo n el nombre del grupo



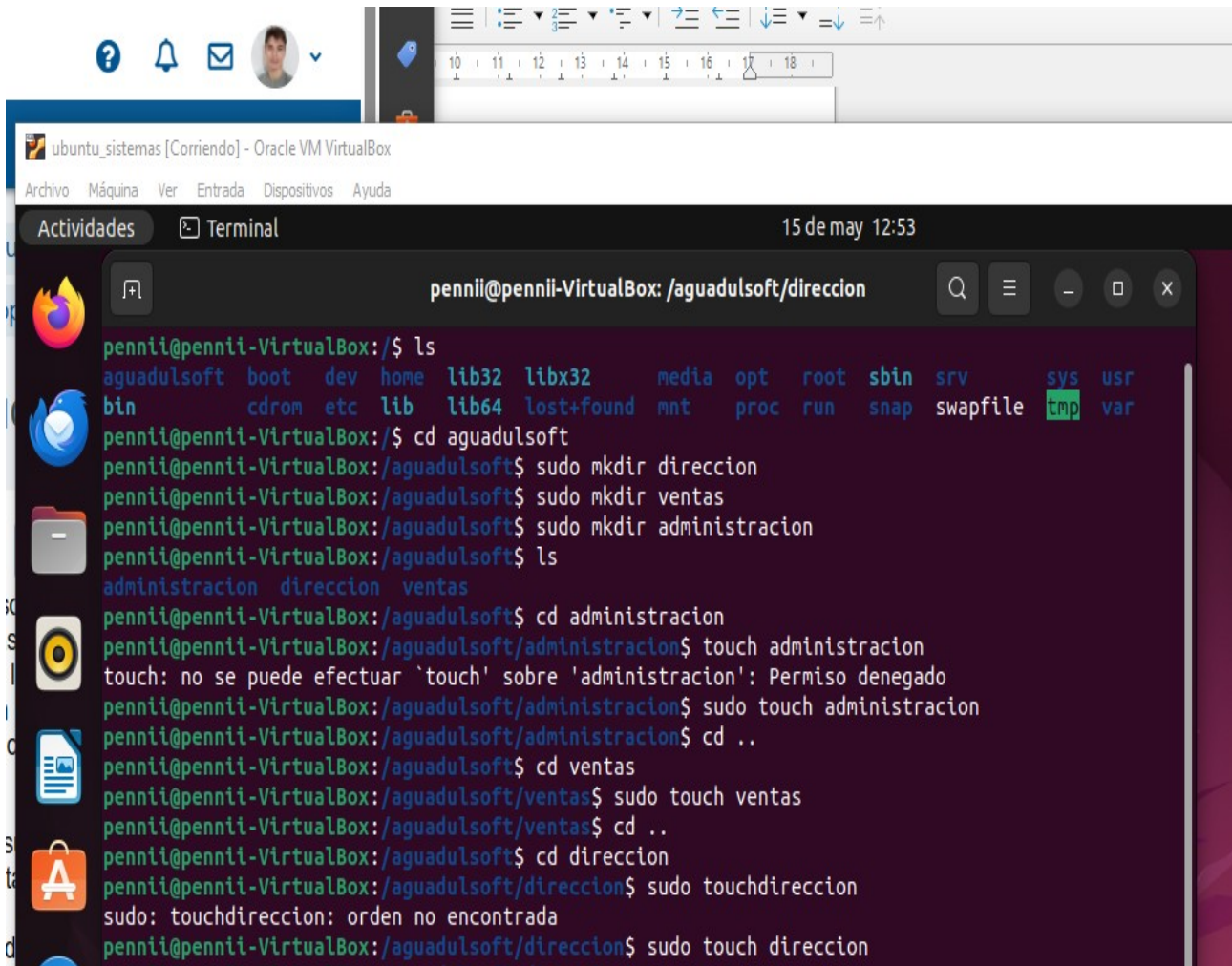
Utilizamos el comando adduser agregando el nombre del grupo al usuario que queremos agregar y con eso el usuario ya estara en el grupo



```
pennii@pennii-VirtualBox: /etc
rtkit:x:123:
whoopsie:x:124:
sssd:x:125:
fwupd-refresh:x:126:
nm-openvpn:x:127:
scanner:x:128:saned
saned:x:129:
colord:x:130:
geoclue:x:131:
pulse:x:132:
pulse-access:x:133:
gdm:x:134:
lxd:x:135:pennii
pennii:x:1000:
sambashare:x:136:pennii
mati:x:1001:
aariza:x:1003:
vvidal:x:1005:
aalonso:x:1002:
vvazquez:x:1004:
administracion:x:1006:aalonso,aariza
ventas:x:1007:vvazquez,vvidal
direccion:x:1008:pennii
pennii@pennii-VirtualBox:/etc$
```

Otra vez en el directorio etc ejecutaremos **cat group**, y podremos ver la lista de grupos y sus usuarios

Actividad 2



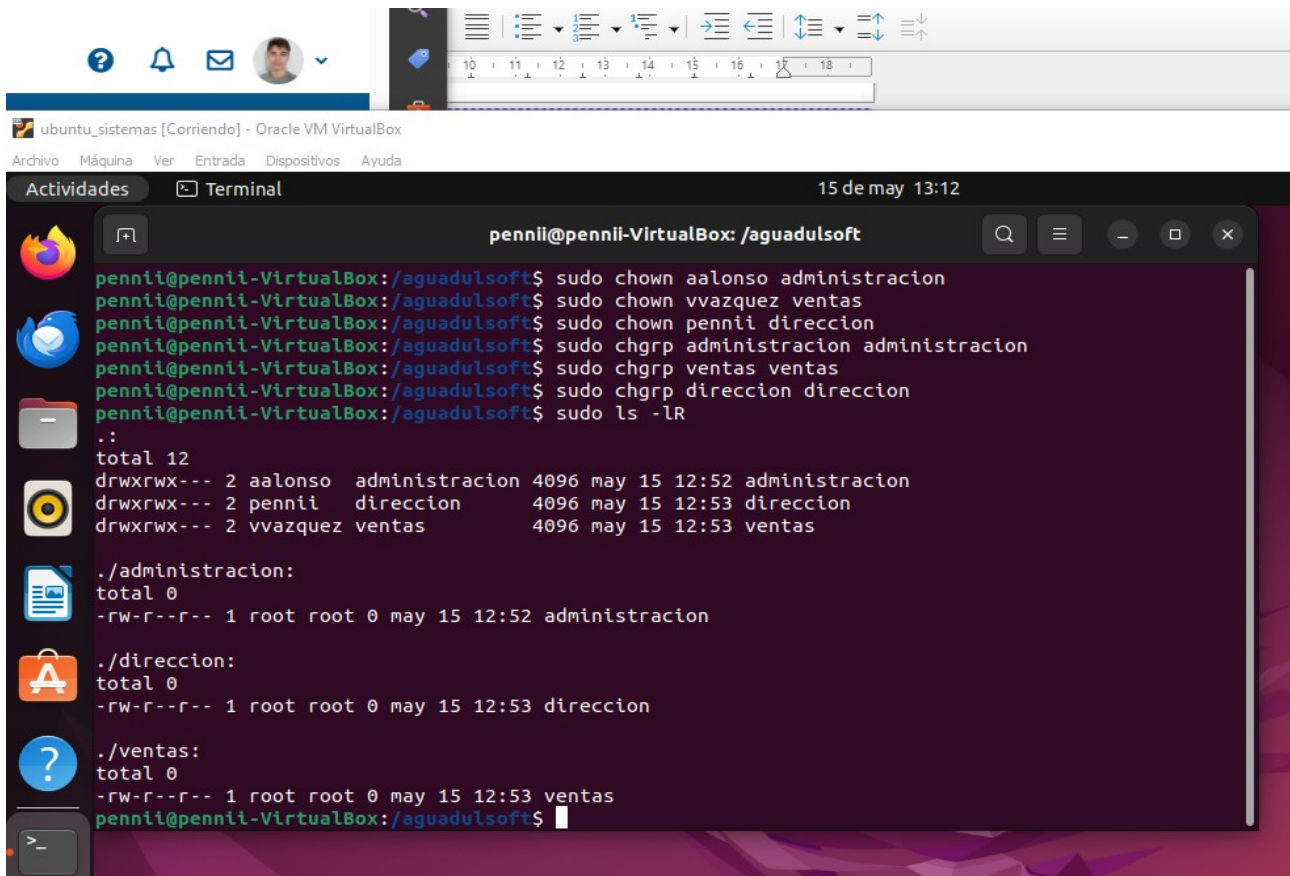
The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: /aguadulsoft/direccion" with a search bar and window controls. The terminal output is as follows:

```
pennii@pennii-VirtualBox:/$ ls
aguadulsoft boot dev home lib32 libx32 media opt root sbin srv sys usr
bin          cdrom etc  lib  lib64 lost+found mnt  proc run  snap swapfile tmp  var

pennii@pennii-VirtualBox:/aguadulsoft$ cd aguadulsoft
pennii@pennii-VirtualBox:/aguadulsoft$ sudo mkdir direccion
pennii@pennii-VirtualBox:/aguadulsoft$ sudo mkdir ventas
pennii@pennii-VirtualBox:/aguadulsoft$ sudo mkdir administracion
pennii@pennii-VirtualBox:/aguadulsoft$ ls
administracion direccion ventas

pennii@pennii-VirtualBox:/aguadulsoft$ cd administracion
pennii@pennii-VirtualBox:/aguadulsoft/administracion$ touch administracion
touch: no se puede efectuar 'touch' sobre 'administracion': Permiso denegado
pennii@pennii-VirtualBox:/aguadulsoft/administracion$ sudo touch administracion
pennii@pennii-VirtualBox:/aguadulsoft/administracion$ cd ..
pennii@pennii-VirtualBox:/aguadulsoft$ cd ventas
pennii@pennii-VirtualBox:/aguadulsoft/ventas$ sudo touch ventas
pennii@pennii-VirtualBox:/aguadulsoft/ventas$ cd ..
pennii@pennii-VirtualBox:/aguadulsoft$ cd direccion
pennii@pennii-VirtualBox:/aguadulsoft/direccion$ sudo touchdireccion
sudo: touchdireccion: orden no encontrada
pennii@pennii-VirtualBox:/aguadulsoft/direccion$ sudo touch direccion
```

Con el comando **mkdir** crearemos el directorio y con **touch** crearemos los ficheros



The screenshot shows a terminal window titled 'pennii@pennii-VirtualBox: /aguadulsoft'. The user 'pennii' is in the directory '/aguadulsoft'. They execute a series of commands to set ownership and group permissions:

```
pennii@pennii-VirtualBox:/aguadulsoft$ sudo chown aalonso administracion
pennii@pennii-VirtualBox:/aguadulsoft$ sudo chown vvazquez ventas
pennii@pennii-VirtualBox:/aguadulsoft$ sudo chown pennii direccion
pennii@pennii-VirtualBox:/aguadulsoft$ sudo chgrp administracion administracion
pennii@pennii-VirtualBox:/aguadulsoft$ sudo chgrp ventas ventas
pennii@pennii-VirtualBox:/aguadulsoft$ sudo chgrp direccion direccion
pennii@pennii-VirtualBox:/aguadulsoft$ sudo ls -lR
```

The output of 'ls -lR' shows the directory structure and permissions:

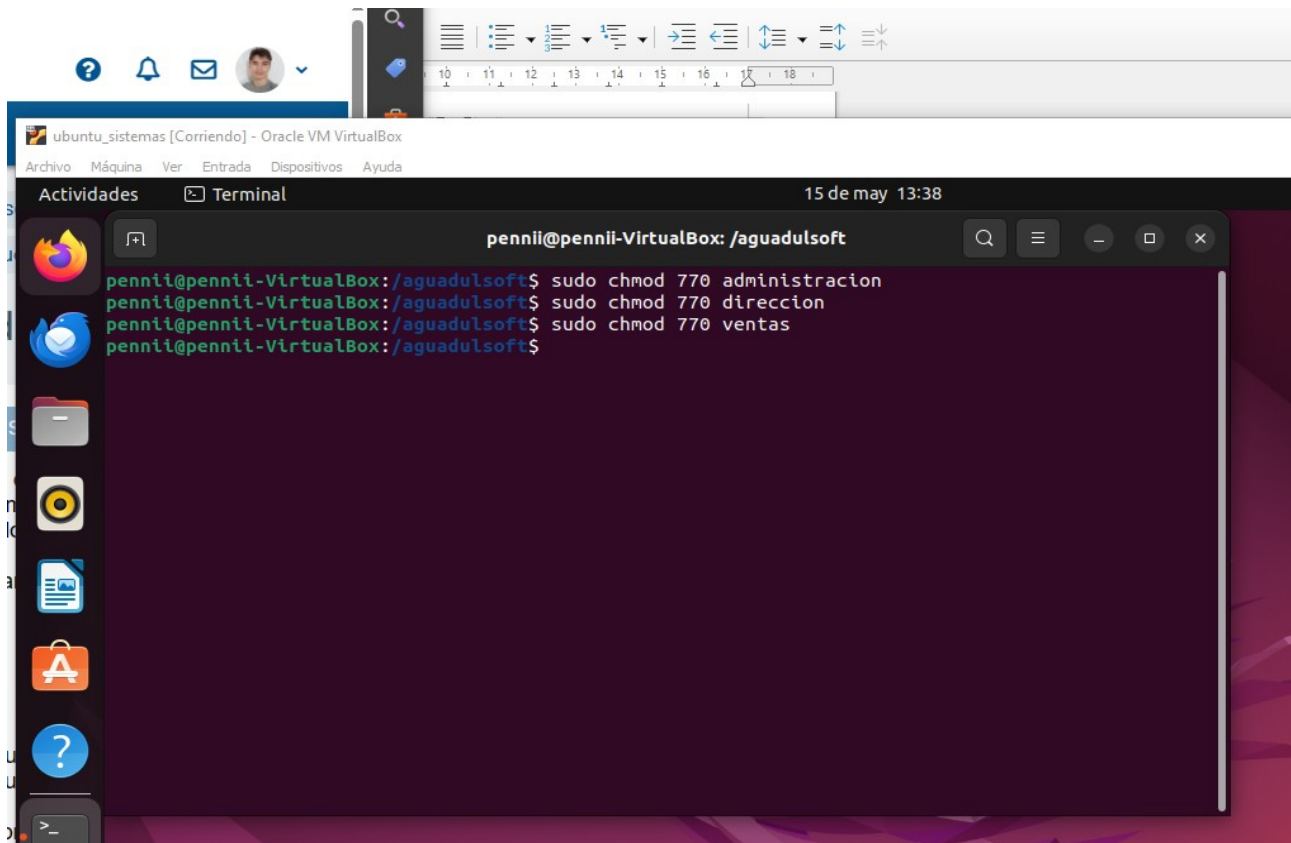
```
..
total 12
drwxrwx--- 2 aalonso administracion 4096 may 15 12:52 administracion
drwxrwx--- 2 pennii direccion 4096 may 15 12:53 direccion
drwxrwx--- 2 vvazquez ventas 4096 may 15 12:53 ventas

./administracion:
total 0
-rw-r--r-- 1 root root 0 may 15 12:52 administracion

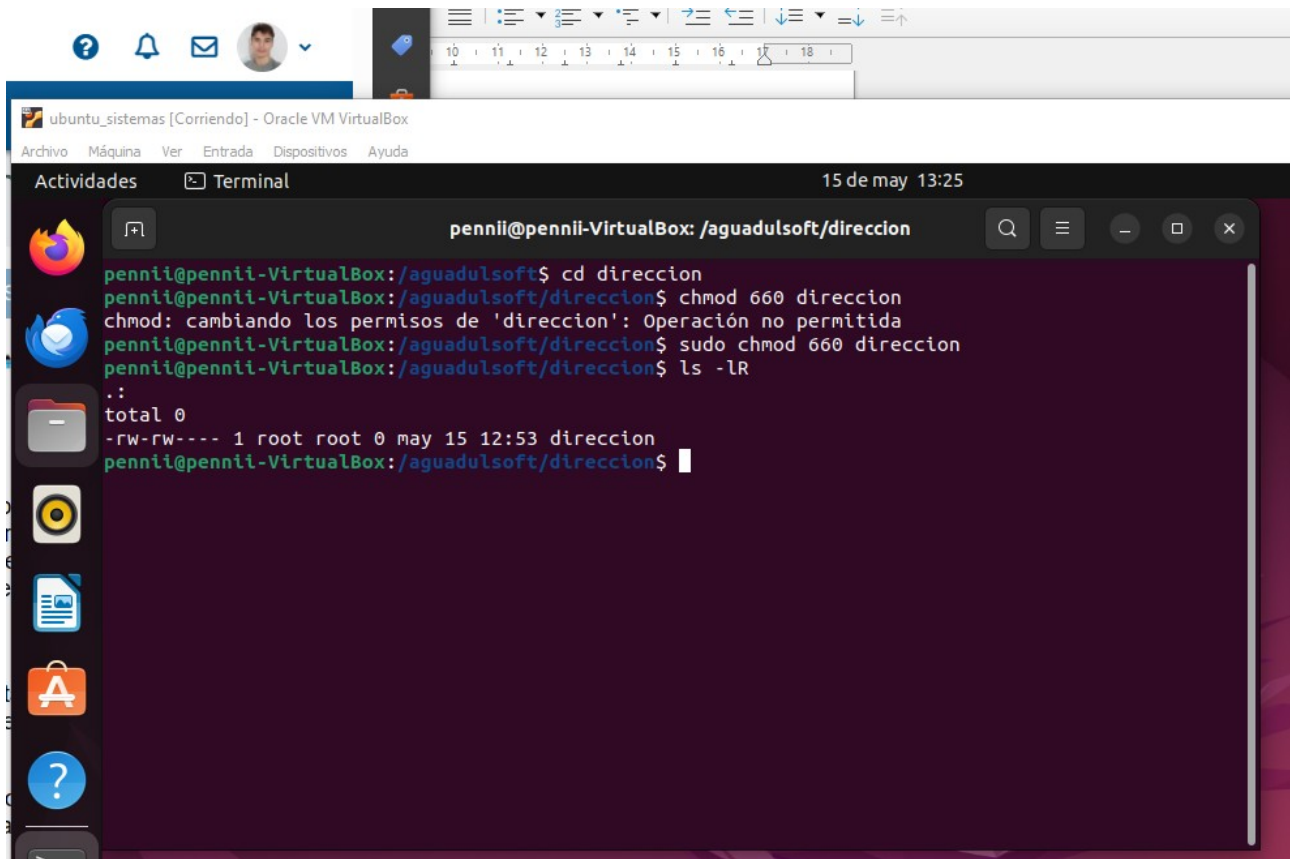
./direccion:
total 0
-rw-r--r-- 1 root root 0 may 15 12:53 direccion

./ventas:
total 0
-rw-r--r-- 1 root root 0 may 15 12:53 ventas
pennii@pennii-VirtualBox:/aguadulsoft$
```

Asignamos la pertenencia de cada fichero a los usuarios y grupos y con el comando **ls -lR** podremos ver la lista de directorios en aguadulsoft y sus respectivos dueños, además de sus permisos que modificaremos mas adelante



Asignamos los permisos a cada directorio con **chmod**, como queremos que el dueño y el grupo de cada directorio tengan todos los permisos les damos 77 y al resto de usuarios 0 para que no tengan ningun permiso.



```
pennii@pennii-VirtualBox: /aguadulsoft/direccion
pennii@pennii-VirtualBox:/aguadulsoft$ cd direccion
pennii@pennii-VirtualBox:/aguadulsoft/direccion$ chmod 660 direccion
chmod: cambiando los permisos de 'direccion': Operación no permitida
pennii@pennii-VirtualBox:/aguadulsoft/direccion$ sudo chmod 660 direccion
pennii@pennii-VirtualBox:/aguadulsoft/direccion$ ls -lR
.:
total 0
-rw-rw---- 1 root root 0 may 15 12:53 direccion
pennii@pennii-VirtualBox:/aguadulsoft/direccion$
```

Ahora en cada directorio asignaremos los permisos de lectura y escritura. Para hacerlo en los directorios donde no tenemos permisos usaremos el comando **su** para cambiar de usuario y hacerlo con su respectivo usuario.

The screenshot shows a terminal window titled 'pennii@pennii-VirtualBox: /' with the date and time '15 de may 13:37'. The user has executed the command 'sudo ls -lR aguadulsoft'. The output shows the directory structure of 'aguadulsoft'.

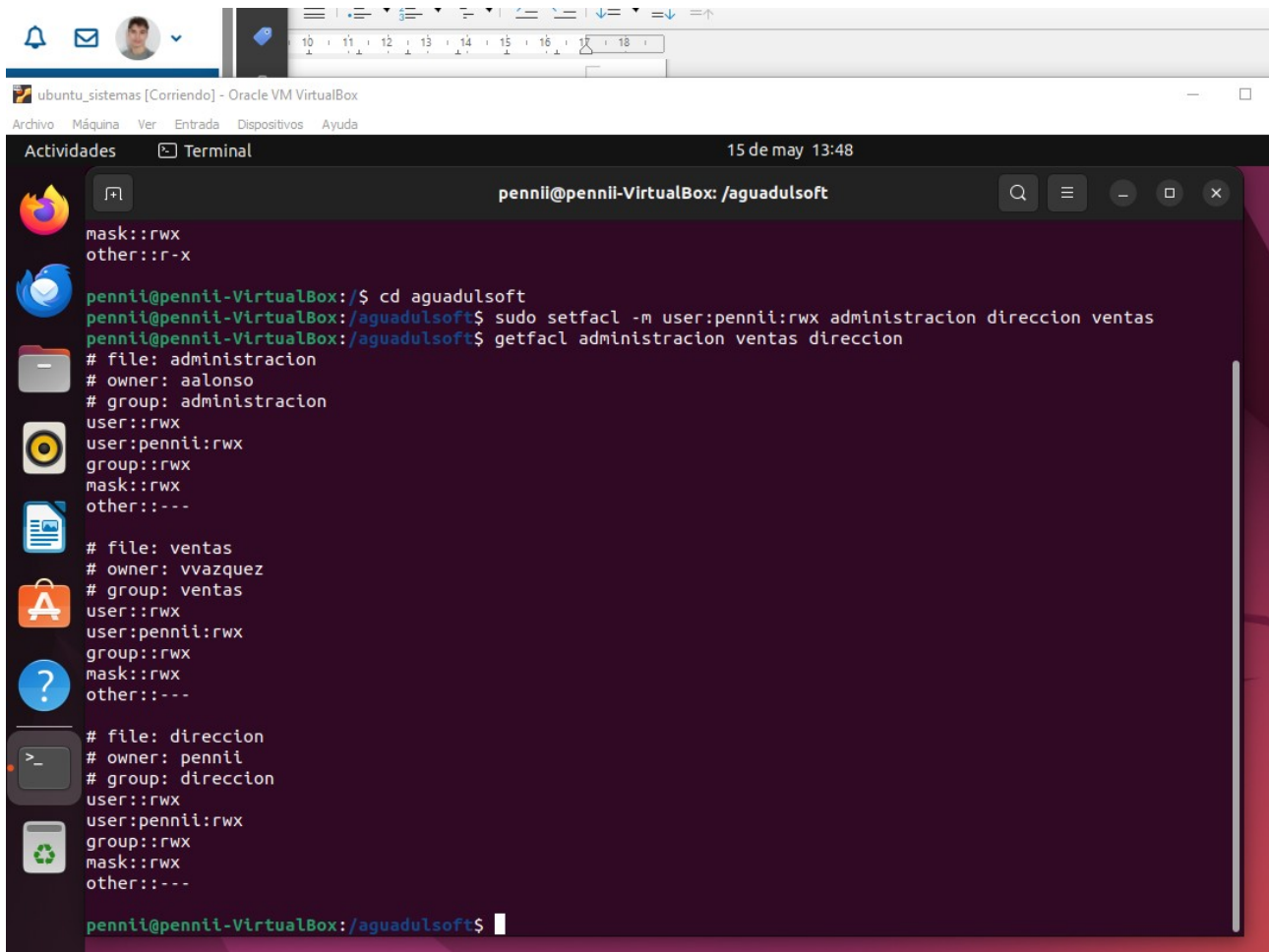
```
pennii@pennii-VirtualBox:/$ sudo ls -lR aguadulsoft
aguadulsoft:
total 12
drwxrwx--- 2 aalonso administracion 4096 may 15 12:52 administracion
drwxrwx--- 2 pennii direccion 4096 may 15 12:53 direccion
drwxrwx--- 2 vvazquez ventas 4096 may 15 12:53 ventas

aguadulsoft/administracion:
total 0
-rw-rw---- 1 root root 0 may 15 12:52 administracion

aguadulsoft/direccion:
total 0
-rw-rw---- 1 root root 0 may 15 12:53 direccion

aguadulsoft/ventas:
total 0
-rw-rw---- 1 root root 0 may 15 12:53 ventas
pennii@pennii-VirtualBox:/$
```

Y así nos quedaría la estructura del directorio aguadulsoft



The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: /aguadulsoft" with a dark purple background. The terminal output shows the following commands and results:

```
mask::rwx
other::r-x

pennii@pennii-VirtualBox:/$ cd aguadulsoft
pennii@pennii-VirtualBox:/aguadulsoft$ sudo setfacl -m user:pennii:rwx administracion direccion ventas
pennii@pennii-VirtualBox:/aguadulsoft$ getfacl administracion ventas direccion

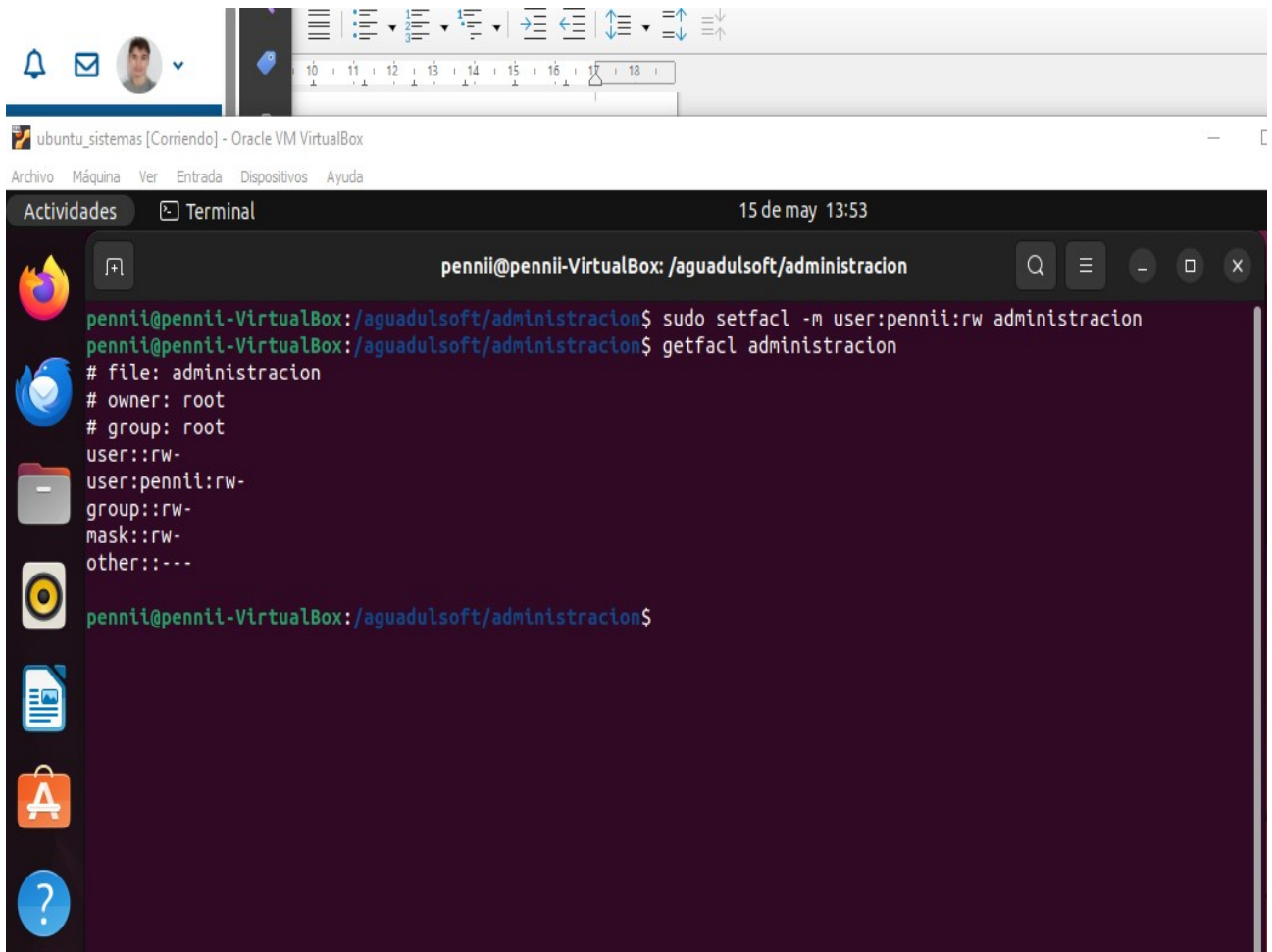
# file: administracion
# owner: aalonso
# group: administracion
user::rwx
user:pennii:rwx
group::rwx
mask::rwx
other::---

# file: ventas
# owner: vvazquez
# group: ventas
user::rwx
user:pennii:rwx
group::rwx
mask::rwx
other::---

# file: direccion
# owner: pennii
# group: direccion
user::rwx
user:pennii:rwx
group::rwx
mask::rwx
other::---
```

The terminal window is part of an Ubuntu system running in Oracle VM VirtualBox. The top of the window shows the VirtualBox interface with a menu bar (Archivo, Máquina, Ver, Entrada, Dispositivos, Ayuda) and a status bar (15 de may 13:48). The left sidebar shows the Ubuntu desktop environment with icons for Firefox, Telegram, Files, and the Dash.

Con el comando **setfacl** asignaremos permisos al director para lectura, ejecucion y escritura en todos los directorios de aguadulsoft

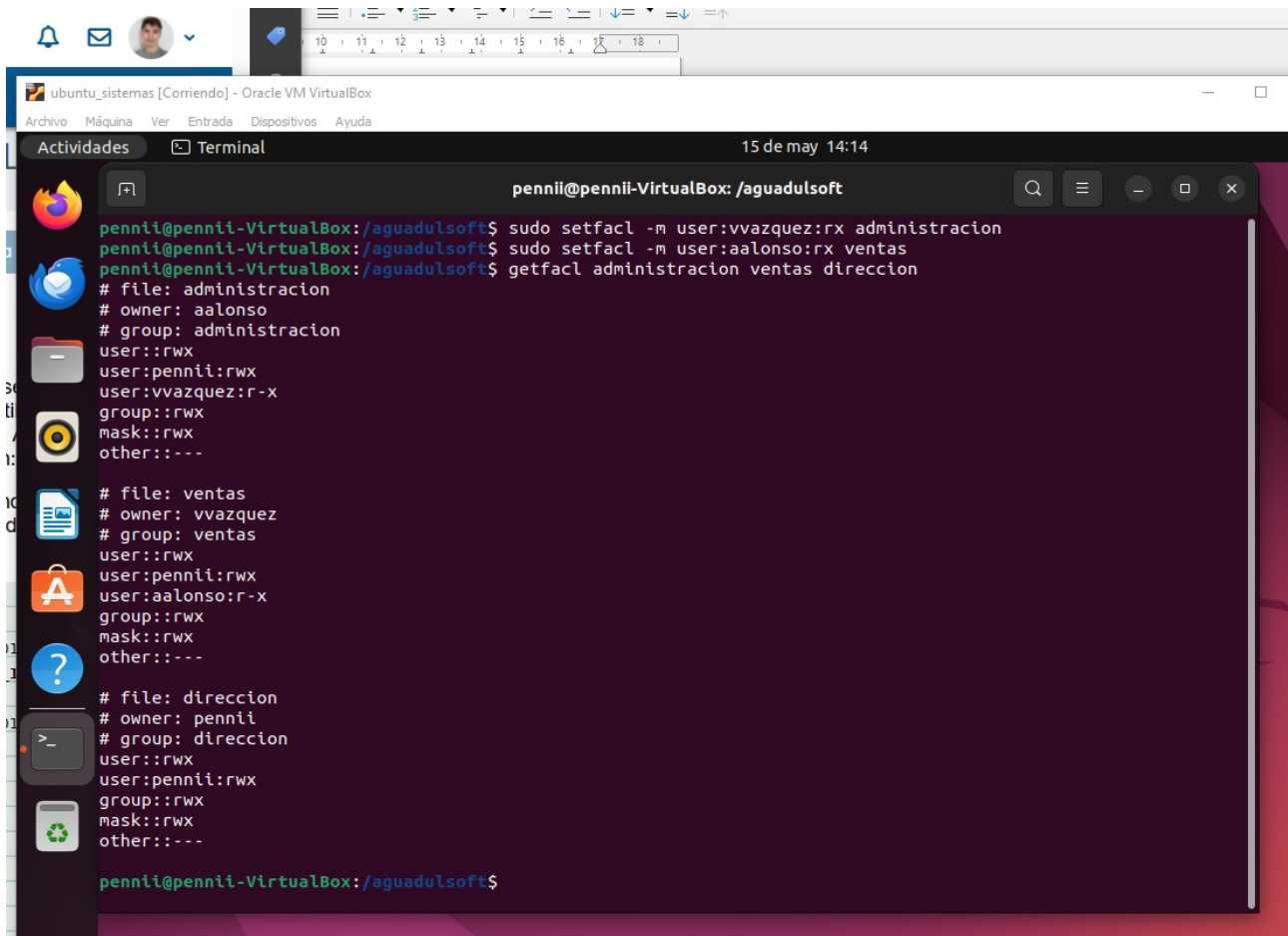


The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: /aguadulsoft/administracion". The terminal output is as follows:

```
pennii@pennii-VirtualBox:/aguadulsoft/administracion$ sudo setfacl -m user:pennii:rw administracion
pennii@pennii-VirtualBox:/aguadulsoft/administracion$ getfacl administracion
# file: administracion
# owner: root
# group: root
user::rw-
user:pennii:rw-
group::rw-
mask::rw-
other::---
```

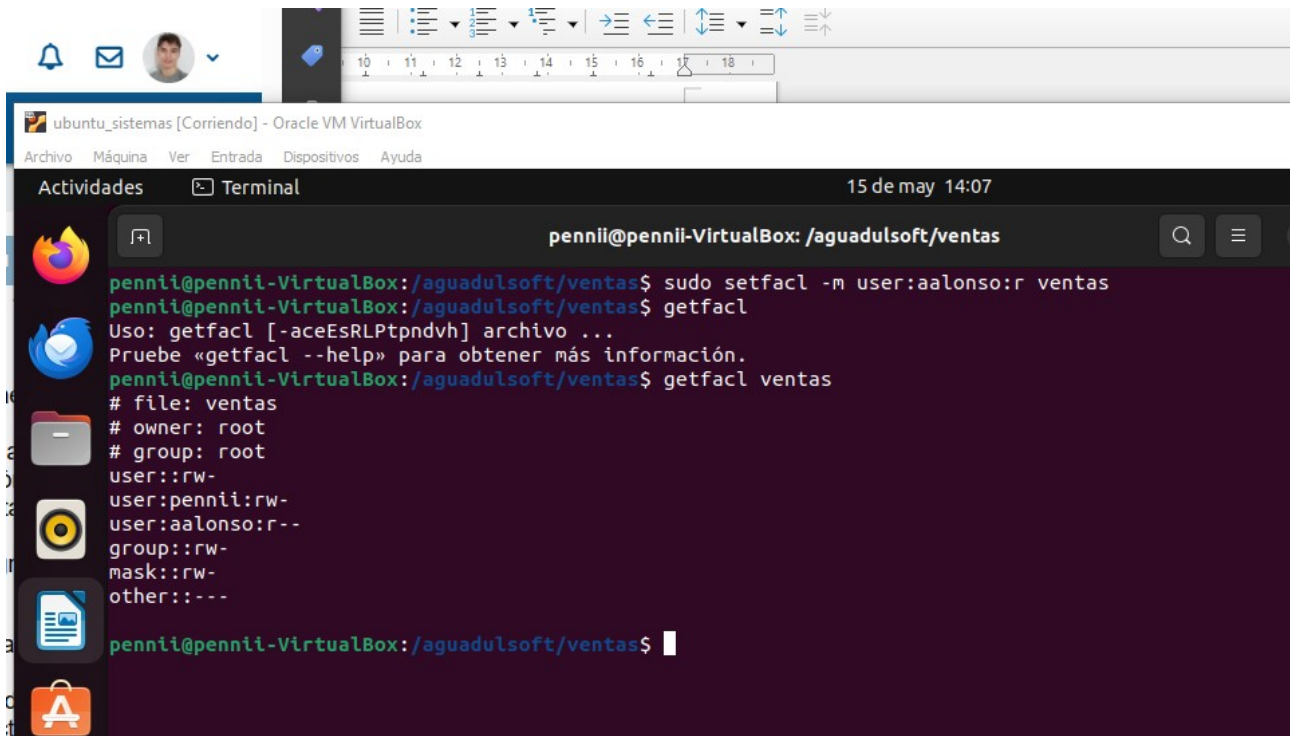
The terminal window is part of a desktop environment with a sidebar on the left containing icons for Firefox, a mail client, a file manager, a terminal, and a help icon. The top of the window shows the system menu with a bell icon, a mail icon, and a user profile icon. The title bar of the terminal window indicates the current directory is "/aguadulsoft/administracion".

Repetiremos el proceso unicamente para lectura y escritura en los ficheros que esten dentro de los subdirectorios, el proceso sera el mismo para cada uno



```
pennii@pennii-VirtualBox: /aguadulsoft
pennii@pennii-VirtualBox:/aguadulsoft$ sudo setfacl -m user:vvazquez:rx administracion
pennii@pennii-VirtualBox:/aguadulsoft$ sudo setfacl -m user:aalonso:rx ventas
pennii@pennii-VirtualBox:/aguadulsoft$ getfacl administracion ventas direccion
# file: administracion
# owner: aalonso
# group: administracion
user::rwx
user:pennii:rwx
user:vvazquez:r-x
group::rwx
mask::rwx
other::---
# file: ventas
# owner: vvazquez
# group: ventas
user::rwx
user:pennii:rwx
user:aalonso:r-x
group::rwx
mask::rwx
other::---
# file: direccion
# owner: pennii
# group: direccion
user::rwx
user:pennii:rwx
group::rwx
mask::rwx
other::---
```

Asignamos los permisos de los encargados en los directorios de aguadulsoft

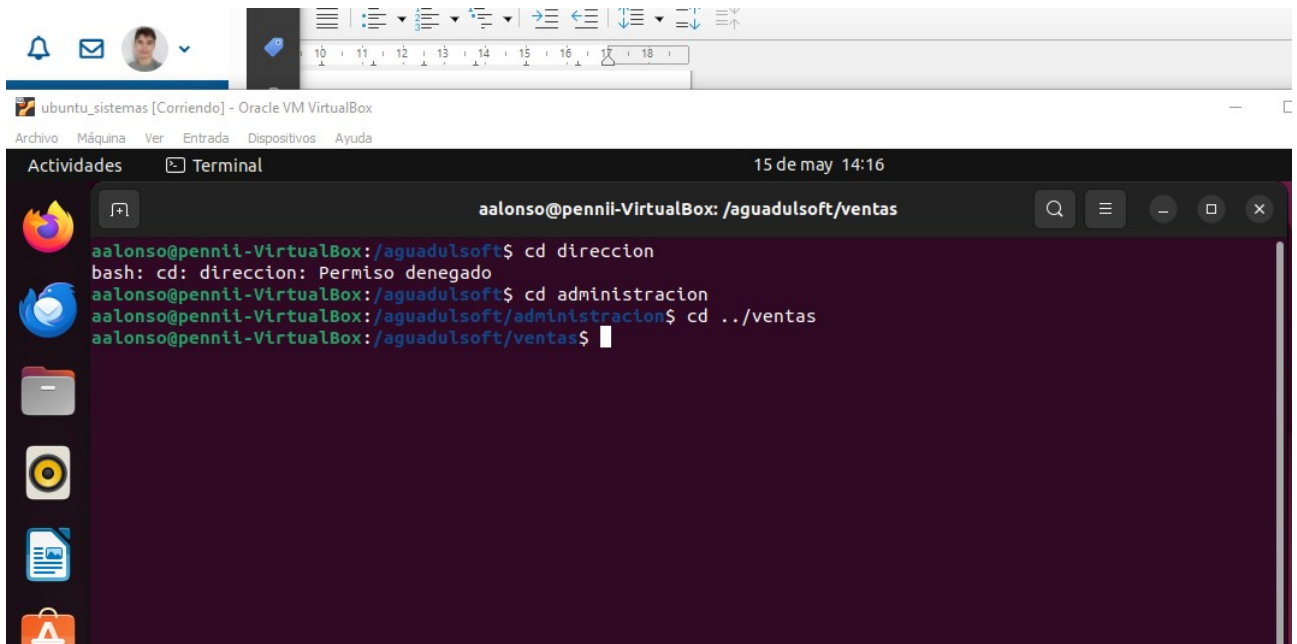


The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: /aguadulsoft/ventas" with a search icon and a menu icon in the top right. The terminal output is as follows:

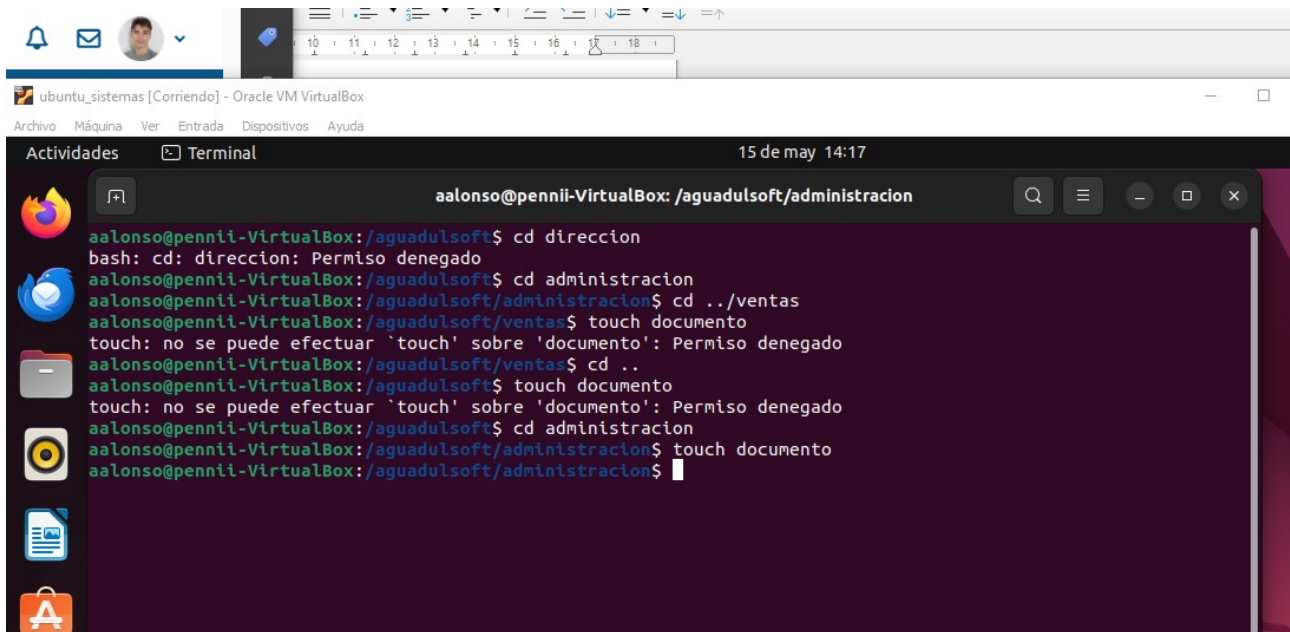
```
pennii@pennii-VirtualBox:/aguadulsoft/ventas$ sudo setfacl -m user:aalonso:r ventas
pennii@pennii-VirtualBox:/aguadulsoft/ventas$ getfacl
Uso: getfacl [-aceEsRLPtpndvh] archivo ...
Pruebe «getfacl --help» para obtener más información.
pennii@pennii-VirtualBox:/aguadulsoft/ventas$ getfacl ventas
# file: ventas
# owner: root
# group: root
user::rw-
user:pennii:rw-
user:aalonso:r--
group::rw-
mask::rw-
other::---
```

The terminal window is part of a desktop environment with a taskbar on the left showing icons for Firefox, Telegram, Files, Music, Documents, and the App Store. The top of the window shows the title bar "ubuntu_sistemas [Corriendo] - Oracle VM VirtualBox" and a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". The date and time "15 de may 14:07" are displayed in the top right corner of the terminal window.

De la misma forma que hicimos con el director a cada encargado le asignaremos permiso de lectura en los archivos de los otros directorios.



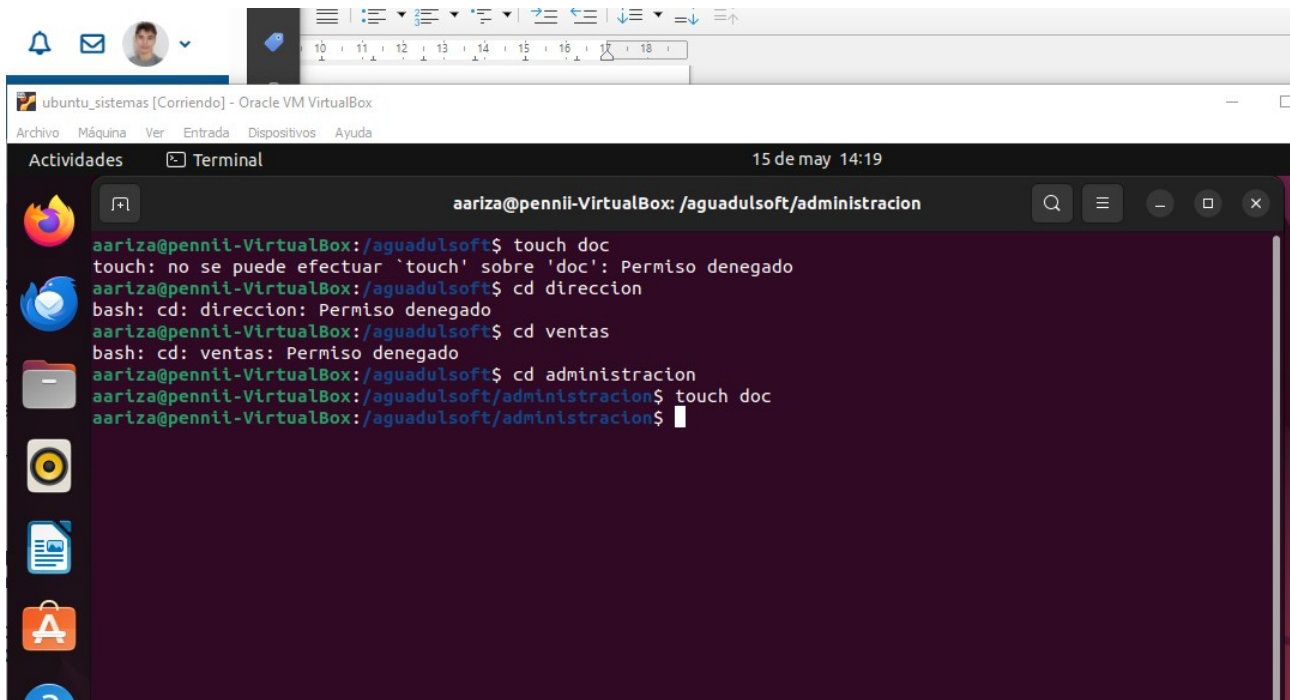
Entramos con el encargado de administracion, no podremos entrar en direccion, pero si en los otros directorios.



The screenshot shows a terminal window titled 'aalonso@pennii-VirtualBox: /aguadulsoft/administracion'. The user 'aalonso' is navigating through a directory structure. The terminal output is as follows:

```
aalonso@pennii-VirtualBox:/aguadulsoft$ cd direccion
bash: cd: direccion: Permiso denegado
aalonso@pennii-VirtualBox:/aguadulsoft$ cd administracion
aalonso@pennii-VirtualBox:/aguadulsoft/administracion$ cd ../ventas
aalonso@pennii-VirtualBox:/aguadulsoft/ventas$ touch documento
touch: no se puede efectuar 'touch' sobre 'documento': Permiso denegado
aalonso@pennii-VirtualBox:/aguadulsoft/ventas$ cd ..
aalonso@pennii-VirtualBox:/aguadulsoft$ touch documento
touch: no se puede efectuar 'touch' sobre 'documento': Permiso denegado
aalonso@pennii-VirtualBox:/aguadulsoft$ cd administracion
aalonso@pennii-VirtualBox:/aguadulsoft/administracion$ touch documento
aalonso@pennii-VirtualBox:/aguadulsoft/administracion$
```

El usuario podra crear un documento unicamente en su directorio, en los otros no tiene permiso



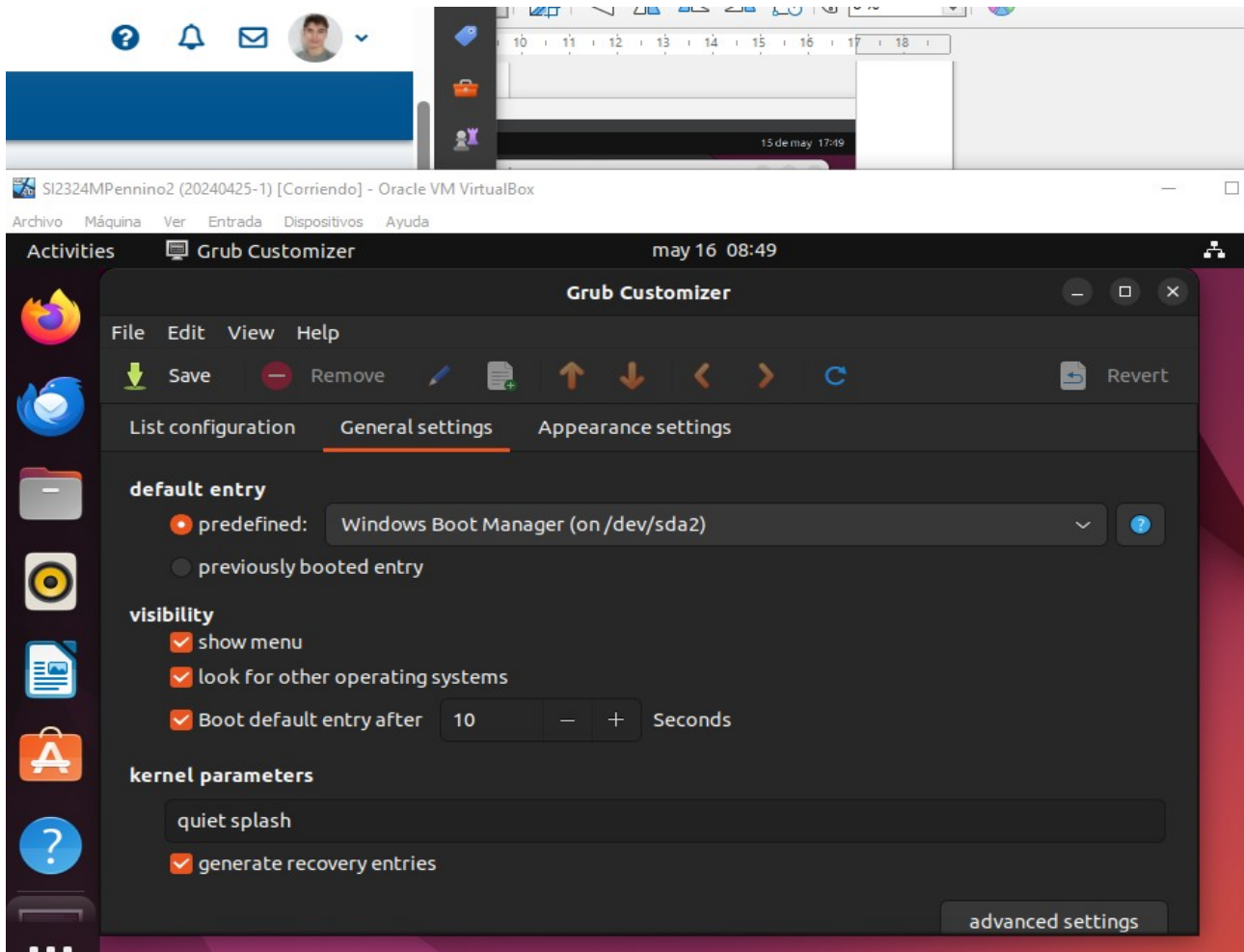
The screenshot shows a terminal window titled "aariza@pennii-VirtualBox: /aguadulsoft/administracion". The user "aariza" is in the directory "/aguadulsoft/administracion". The terminal shows the following commands and output:

```
aariza@pennii-VirtualBox:/aguadulsoft$ touch doc
touch: no se puede efectuar `touch' sobre `doc': Permiso denegado
aariza@pennii-VirtualBox:/aguadulsoft$ cd direccion
bash: cd: direccion: Permiso denegado
aariza@pennii-VirtualBox:/aguadulsoft$ cd ventas
bash: cd: ventas: Permiso denegado
aariza@pennii-VirtualBox:/aguadulsoft$ cd administracion
aariza@pennii-VirtualBox:/aguadulsoft/administracion$ touch doc
aariza@pennii-VirtualBox:/aguadulsoft/administracion$
```

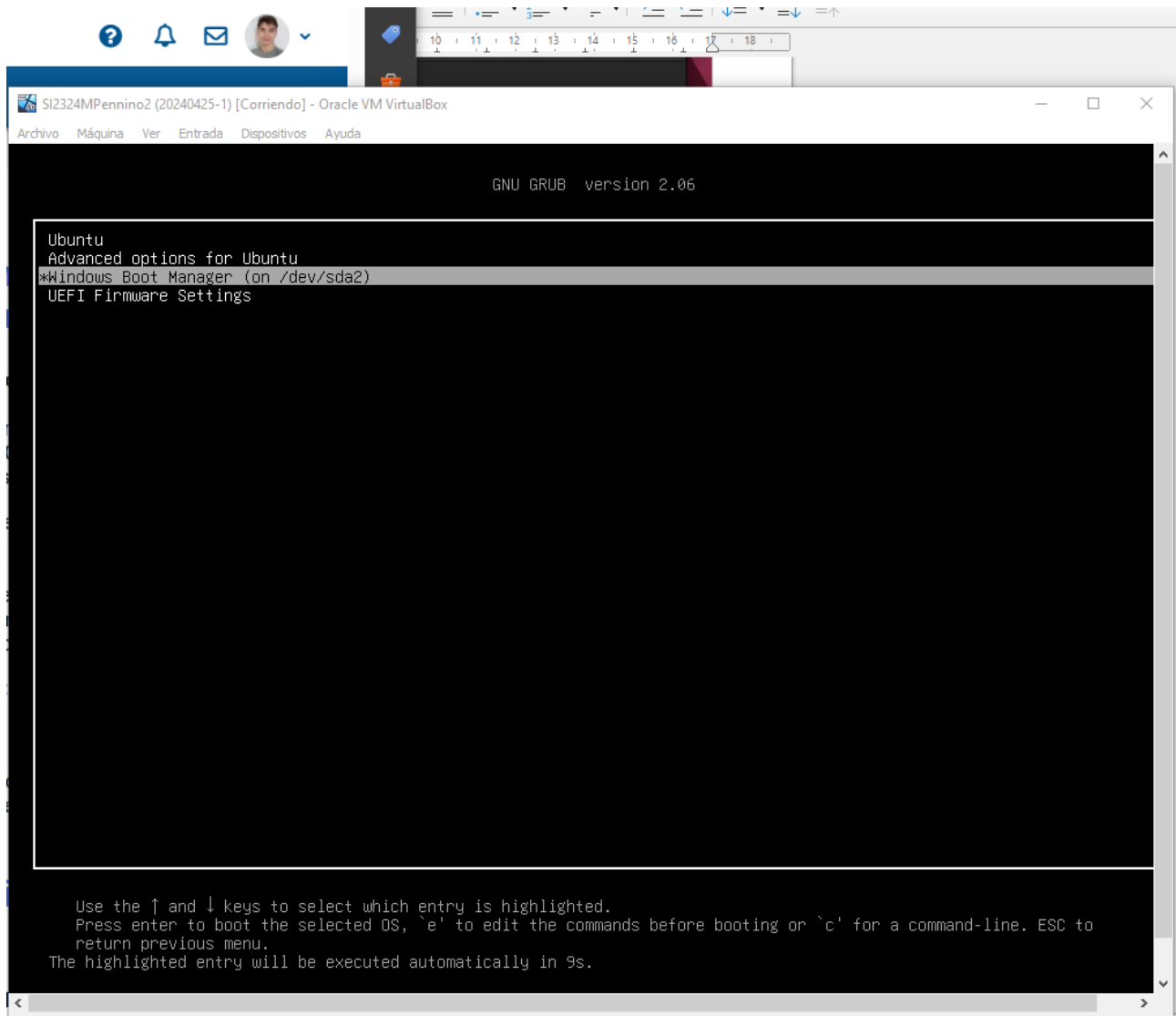
The terminal window is part of a desktop environment with a sidebar on the left containing icons for Firefox, Telegram, Files, Music, Documents, and Applications. The top of the window shows the title bar and a menu bar with options like "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". The date and time "15 de may 14:19" are displayed in the top right corner.

Un usuario que no es encargado unicamente puede entrar a su directorio y solo puede crear documentos ahi

Actividad 3

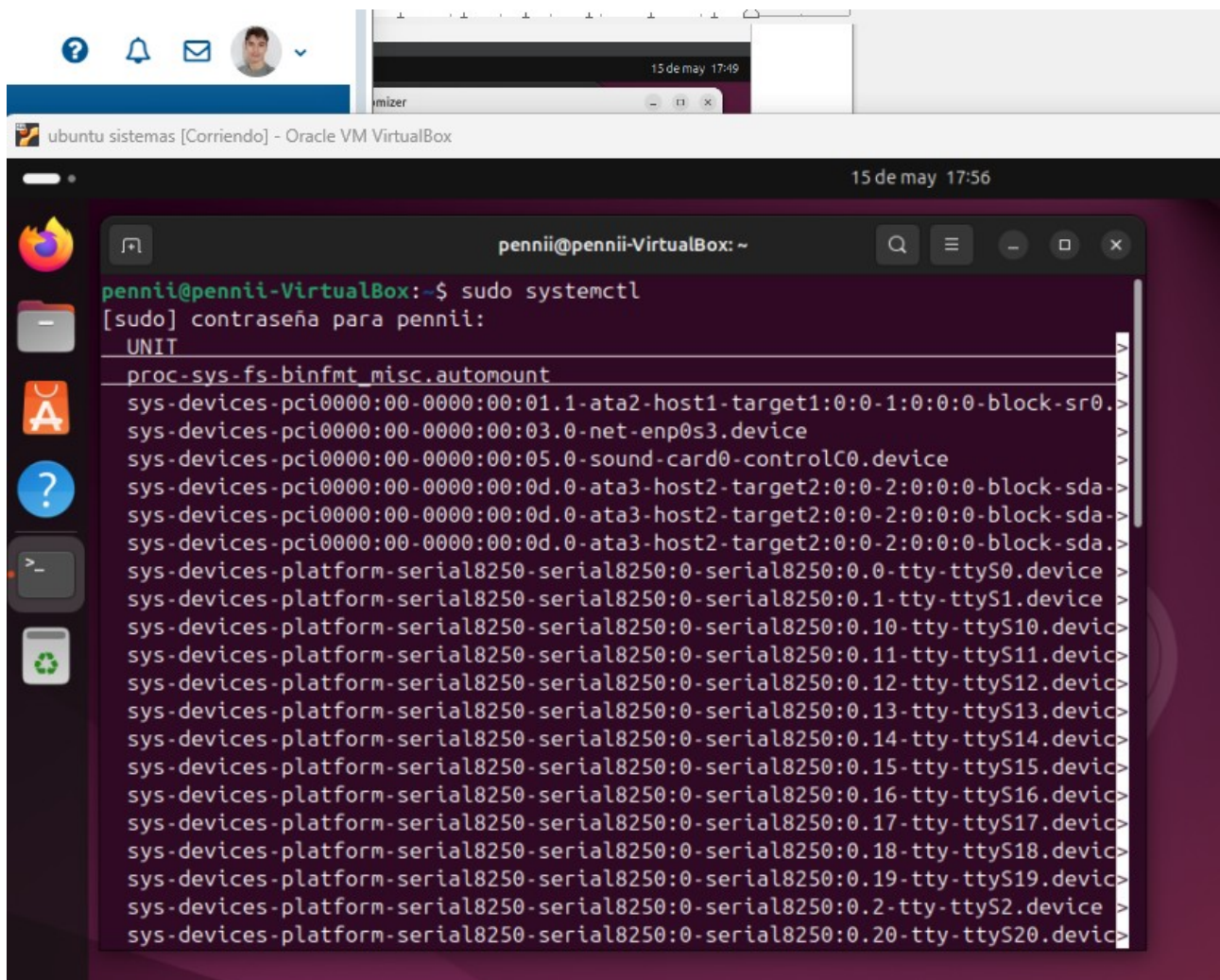


Estableceremos al gestor de arranque de windows como el sistema por defecto y que se inicie a los 10 segundos si es que el usuario no hace nada



Aqui se puede ver como funciona correctamente la configuracion

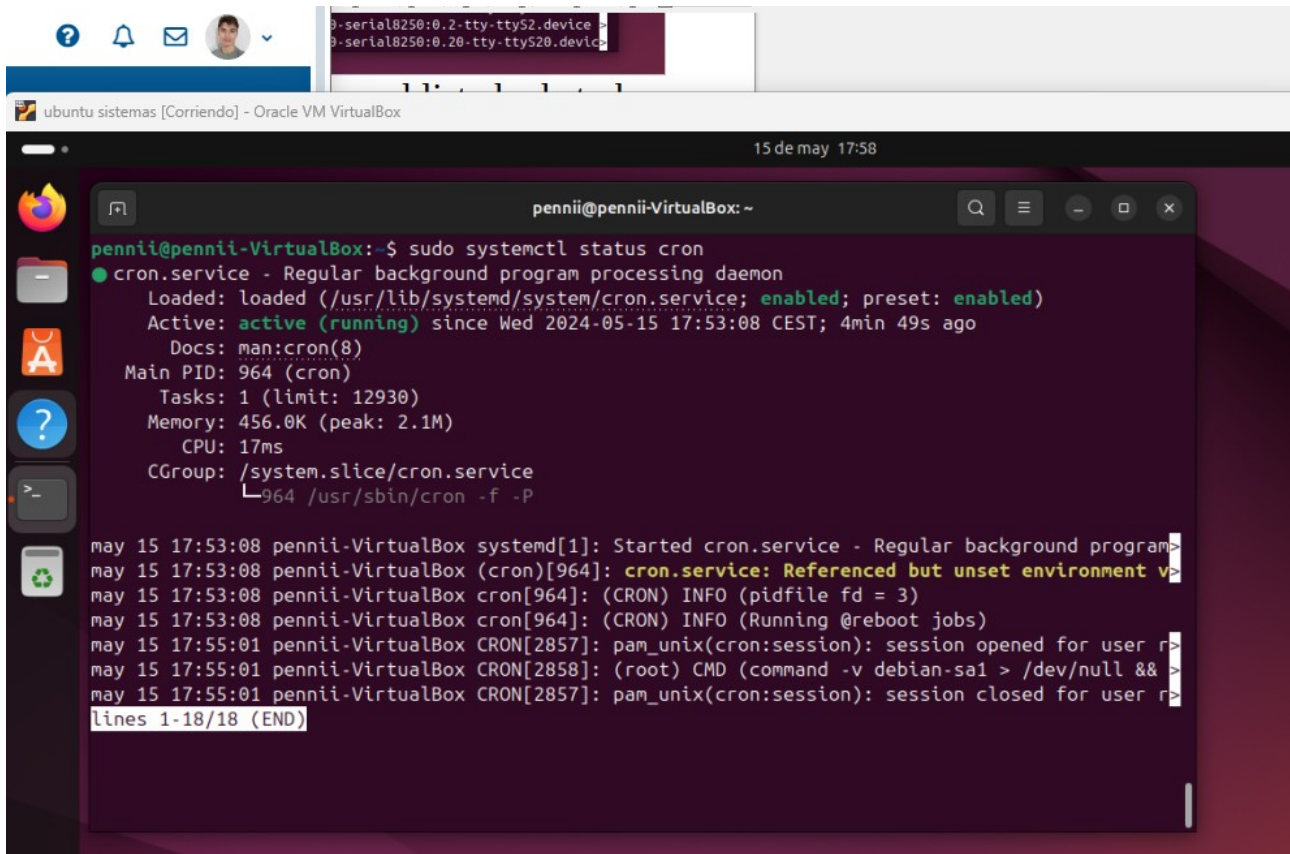
Actividad 4



The screenshot shows a terminal window titled 'pennii@pennii-VirtualBox: ~' running the command 'sudo systemctl'. The prompt asks for the password for 'pennii'. Below the prompt, a list of system units is displayed, including 'proc-sys-fs-binfmt_misc.automount' and various 'sys-devices' units for PCI and platform devices. The list is truncated with '>' at the end of each line.

```
pennii@pennii-VirtualBox: ~  
[sudo] contraseña para pennii:  
UNIT  
proc-sys-fs-binfmt_misc.automount  
sys-devices-pci0000:00-0000:00:01.1-ata2-host1-target1:0:0-1:0:0:0-block-sr0.>  
sys-devices-pci0000:00-0000:00:03.0-net-enp0s3.device  
sys-devices-pci0000:00-0000:00:05.0-sound-card0-controlC0.device  
sys-devices-pci0000:00-0000:00:0d.0-ata3-host2-target2:0:0-2:0:0:0-block-sda->  
sys-devices-pci0000:00-0000:00:0d.0-ata3-host2-target2:0:0-2:0:0:0-block-sda->  
sys-devices-pci0000:00-0000:00:0d.0-ata3-host2-target2:0:0-2:0:0:0-block-sda->  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.0-tty-ttyS0.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.1-tty-ttyS1.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.10-tty-ttyS10.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.11-tty-ttyS11.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.12-tty-ttyS12.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.13-tty-ttyS13.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.14-tty-ttyS14.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.15-tty-ttyS15.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.16-tty-ttyS16.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.17-tty-ttyS17.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.18-tty-ttyS18.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.19-tty-ttyS19.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.2-tty-ttyS2.device>  
sys-devices-platform-serial8250-serial8250:0-serial8250:0.20-tty-ttyS20.device>
```

Ejecutando **systemctl** veremos el listado de todos los servicios

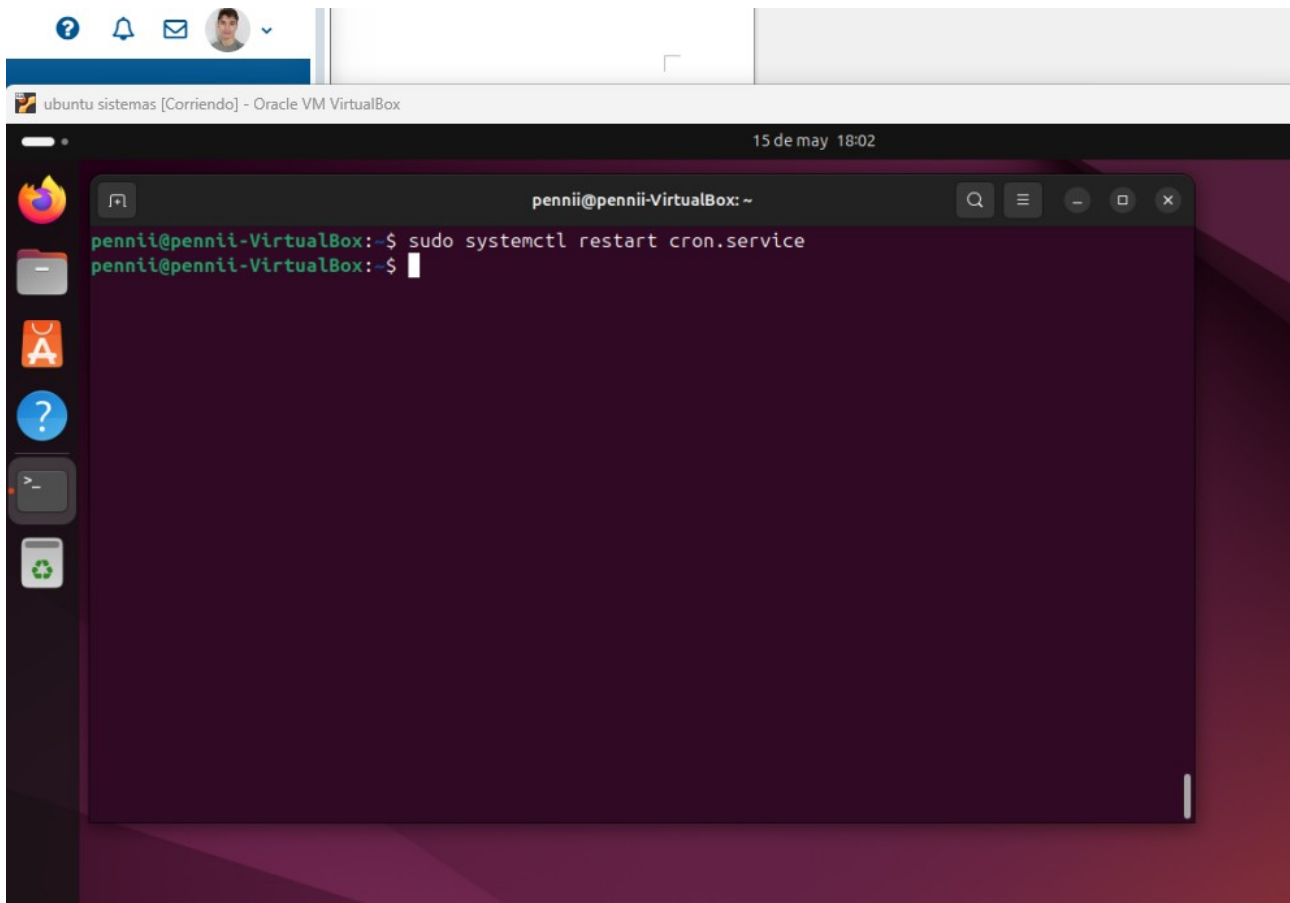


The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: ~" with a search icon, menu icon, and window control buttons. The terminal displays the output of the command `sudo systemctl status cron`. The output shows that the `cron.service` is a regular background program processing daemon, loaded from `/usr/lib/systemd/system/cron.service`, and is currently active (running) since Wednesday, May 15, 2024, at 17:53:08 CEST, 4 minutes and 49 seconds ago. The service has 1 task (limit: 12930), uses 456.0K of memory (peak: 2.1M), and has a CPU usage of 17ms. The CGroup is `/system.slice/cron.service`. The terminal also shows a log of the service starting and running, including a message about a referenced but unset environment variable.

```
pennii@pennii-VirtualBox:~$ sudo systemctl status cron
● cron.service - Regular background program processing daemon
   Loaded: loaded (/usr/lib/systemd/system/cron.service; enabled; preset: enabled)
   Active: active (running) since Wed 2024-05-15 17:53:08 CEST; 4min 49s ago
     Docs: man:cron(8)
    Main PID: 964 (cron)
      Tasks: 1 (limit: 12930)
     Memory: 456.0K (peak: 2.1M)
        CPU: 17ms
    CGroup: /system.slice/cron.service
           └─964 /usr/sbin/cron -f -P

may 15 17:53:08 pennii-VirtualBox systemd[1]: Started cron.service - Regular background program
may 15 17:53:08 pennii-VirtualBox (cron)[964]: cron.service: Referenced but unset environment v
may 15 17:53:08 pennii-VirtualBox cron[964]: (CRON) INFO (pidfile fd = 3)
may 15 17:53:08 pennii-VirtualBox cron[964]: (CRON) INFO (Running @reboot jobs)
may 15 17:55:01 pennii-VirtualBox CRON[2857]: pam_unix(cron:session): session opened for user r
may 15 17:55:01 pennii-VirtualBox CRON[2858]: (root) CMD (command -v debian-sa1 > /dev/null && >
may 15 17:55:01 pennii-VirtualBox CRON[2857]: pam_unix(cron:session): session closed for user r
lines 1-18/18 (END)
```

Si al comando anterior le agregamos **status n**, siendo n el nombre de un servicio, veremos todos los detalles de ese servicio



Como el servicio ya esta iniciado lo reiniciamos

The screenshot shows a terminal window titled "pennii@pennii-VirtualBox: ~" with a search icon, a menu icon, and window control buttons. The terminal displays the following commands and output:

```
pennii@pennii-VirtualBox:~$ sudo systemctl restart cron.service
pennii@pennii-VirtualBox:~$ sudo systemctl status cron
```

● cron.service - Regular background program processing daemon

Loaded: loaded (/usr/lib/systemd/system/cron.service; **enabled**; preset: **enabled**)

Active: **active (running)** since Wed 2024-05-15 18:02:15 CEST; 59s ago

Docs: man:cron(8)

Main PID: 3131 (cron)

Tasks: 1 (limit: 12930)

Memory: 340.0K (peak: 516.0K)

CPU: 3ms

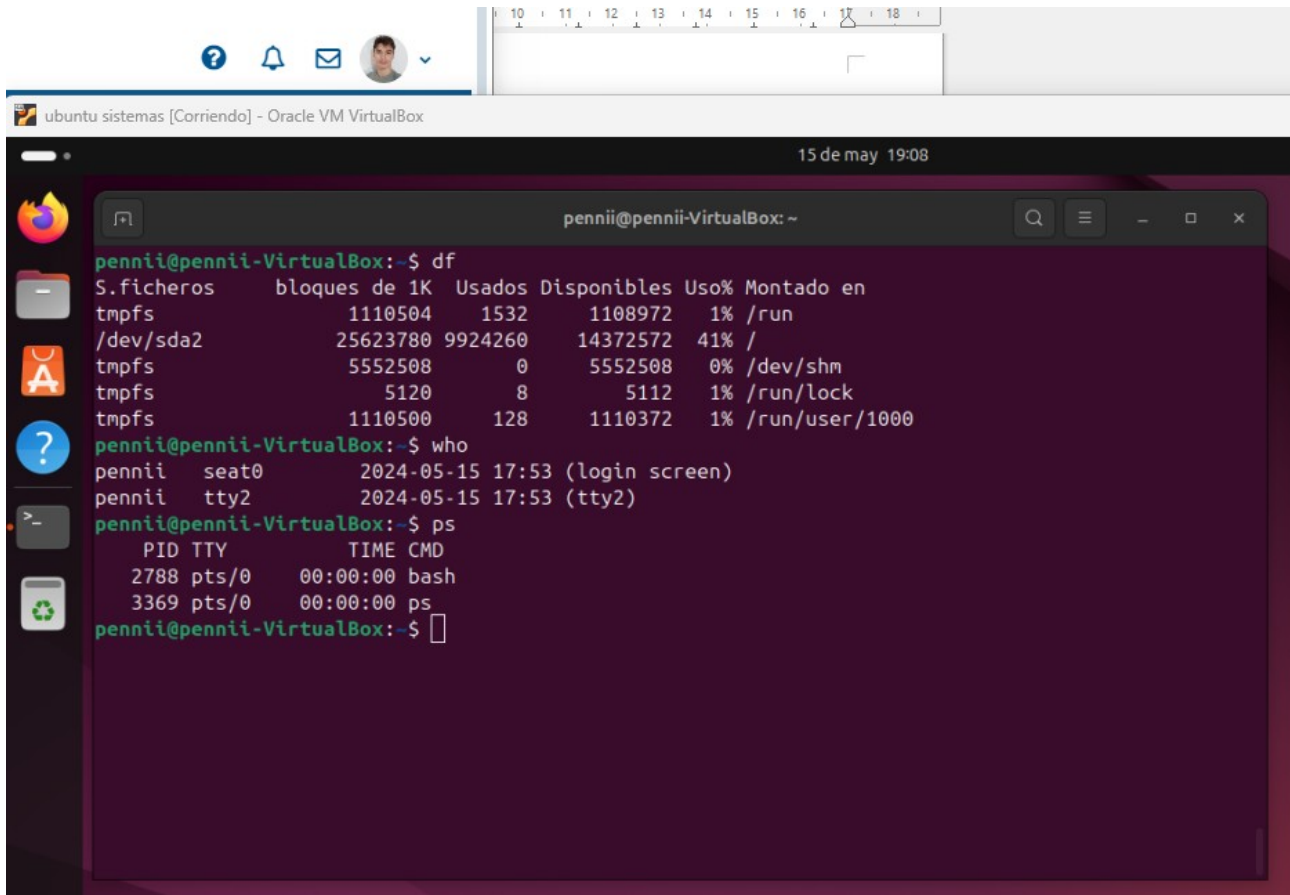
CGroup: /system.slice/cron.service

└─3131 /usr/sbin/cron -f -P

```
may 15 18:02:15 pennii-VirtualBox systemd[1]: Started cron.service - Regular background program
may 15 18:02:15 pennii-VirtualBox (cron)[3131]: cron.service: Referenced but unset environment
may 15 18:02:15 pennii-VirtualBox cron[3131]: (CRON) INFO (pidfile fd = 3)
may 15 18:02:15 pennii-VirtualBox cron[3131]: (CRON) INFO (Skipping @reboot jobs -- not system
lines 1-15/15 (END)
```

Y mostramos el estado final del servicio

Actividad 5



The screenshot shows a terminal window titled 'pennii@pennii-VirtualBox: ~' with a dark purple background. The terminal displays the output of three commands: 'df', 'who', and 'ps'. The 'df' command shows disk usage for various filesystems. The 'who' command shows the login session for the user 'pennii'. The 'ps' command shows the running processes.

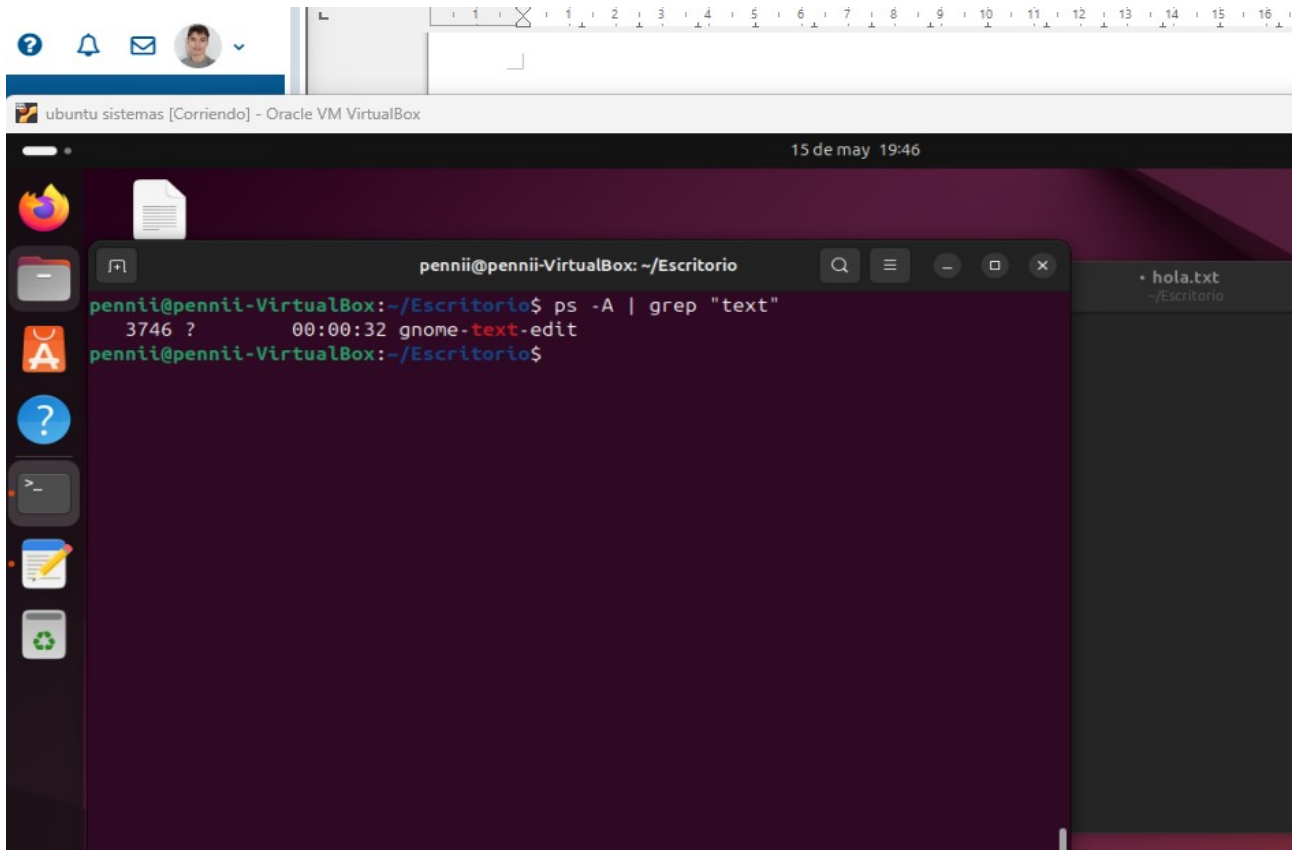
```
pennii@pennii-VirtualBox:~$ df
S.ficheros      bloques de 1K  Usados  Disponibles  Uso%  Montado en
tmpfs           1110504      1532    1108972      1%  /run
/dev/sda2       25623780  9924260  14372572    41%  /
tmpfs           5552508        0    5552508      0%  /dev/shm
tmpfs           5120         8       5112      1%  /run/lock
tmpfs           1110500      128    1110372      1%  /run/user/1000

pennii@pennii-VirtualBox:~$ who
pennii  seat0          2024-05-15 17:53 (login screen)
pennii  tty2            2024-05-15 17:53 (tty2)

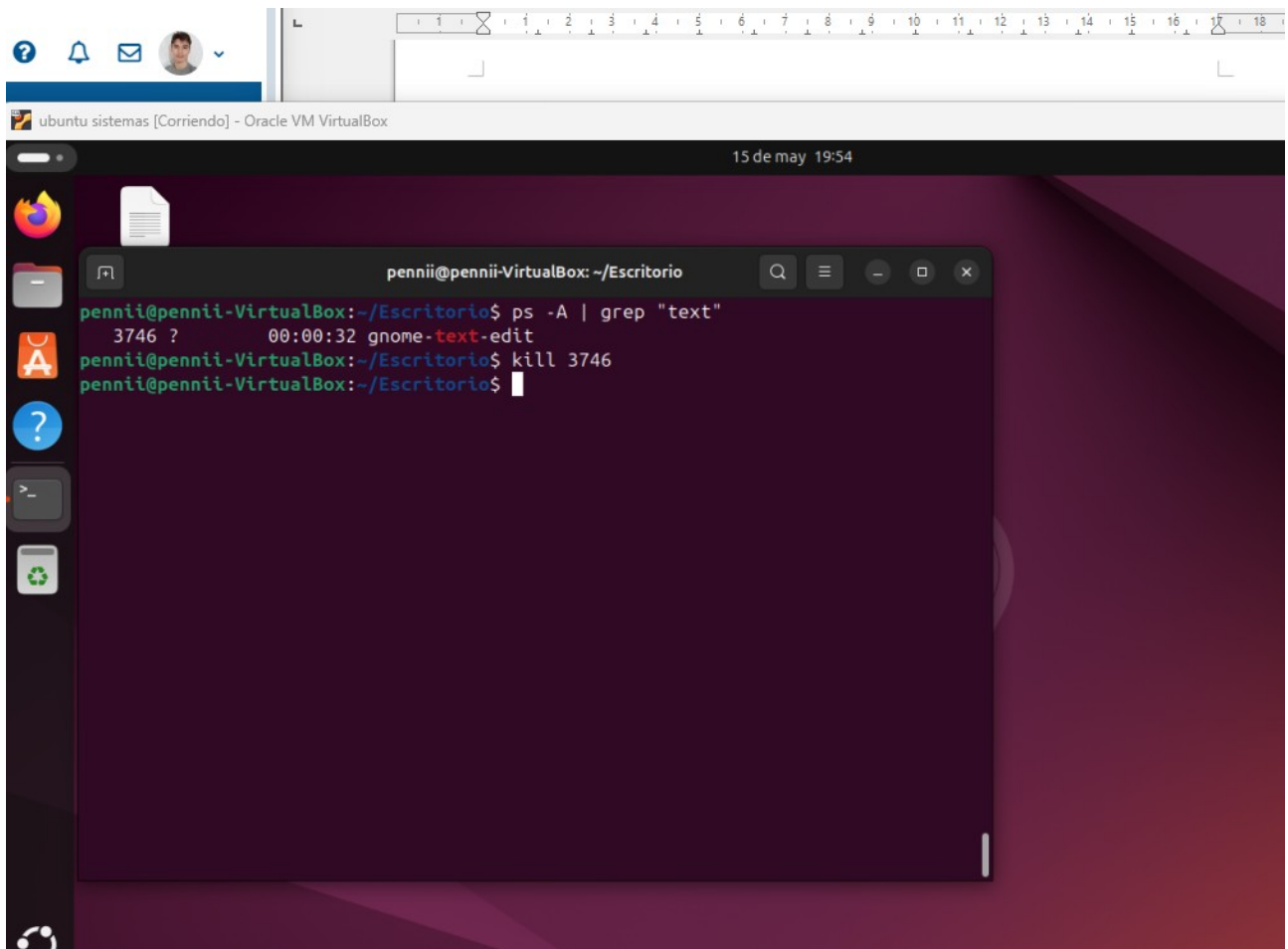
pennii@pennii-VirtualBox:~$ ps
  PID TTY          TIME CMD
 2788 pts/0    00:00:00 bash
 3369 pts/0    00:00:00 ps
pennii@pennii-VirtualBox:~$
```

Se ejecutaron los siguientes comandos:

- df: Muestra el espacio libre del sistema de ficheros
- who: muestra en que momento inicio cada usuario al sistema
- ps: muestra los procesos que se estan ejecutando

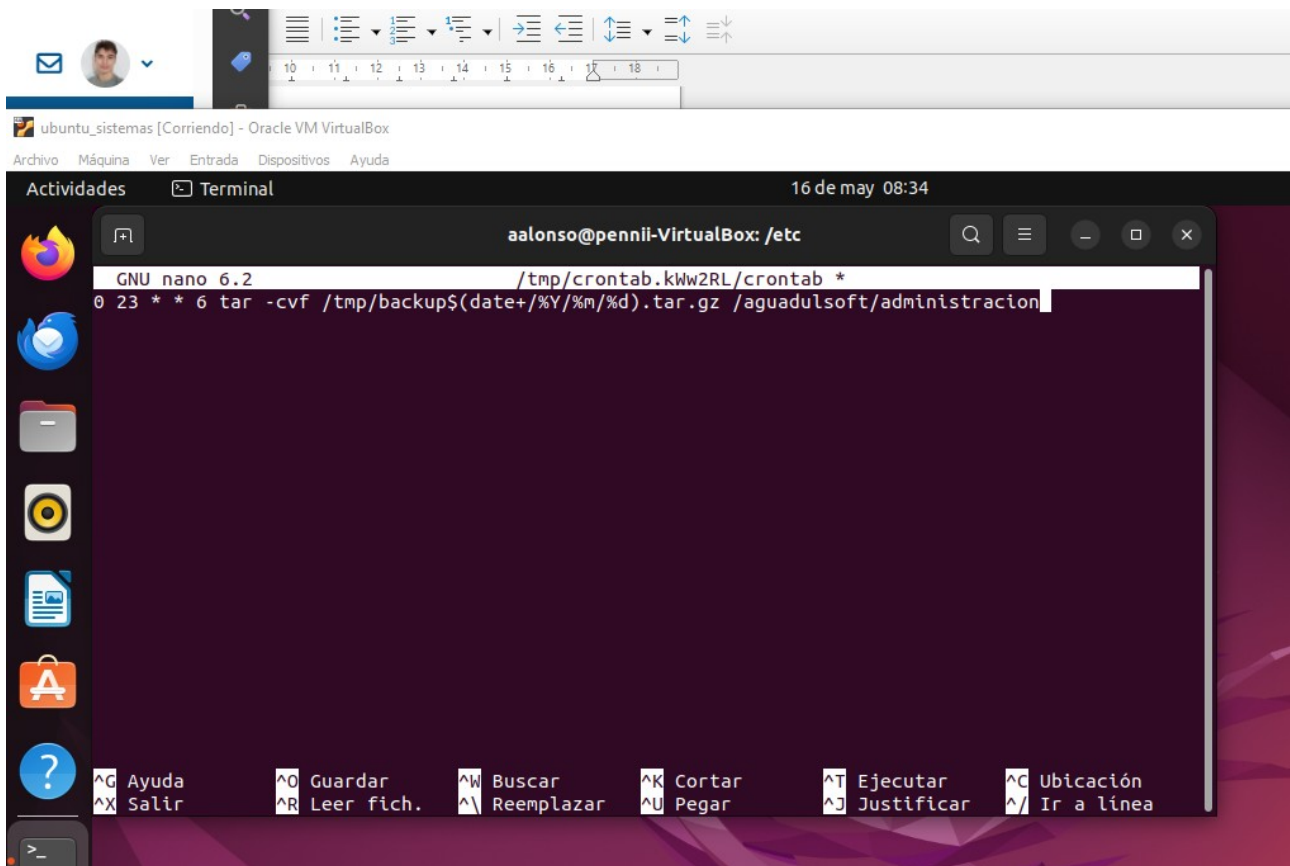


Se creo el archivo de texto “hola” y mediante el comando **ps -A** podremos ver todos los procesos que se estan ejecutando. Escribiendo **grep “text”**, podremos filtrar los resultados para encontrar mas facilmente el proceso y su PID



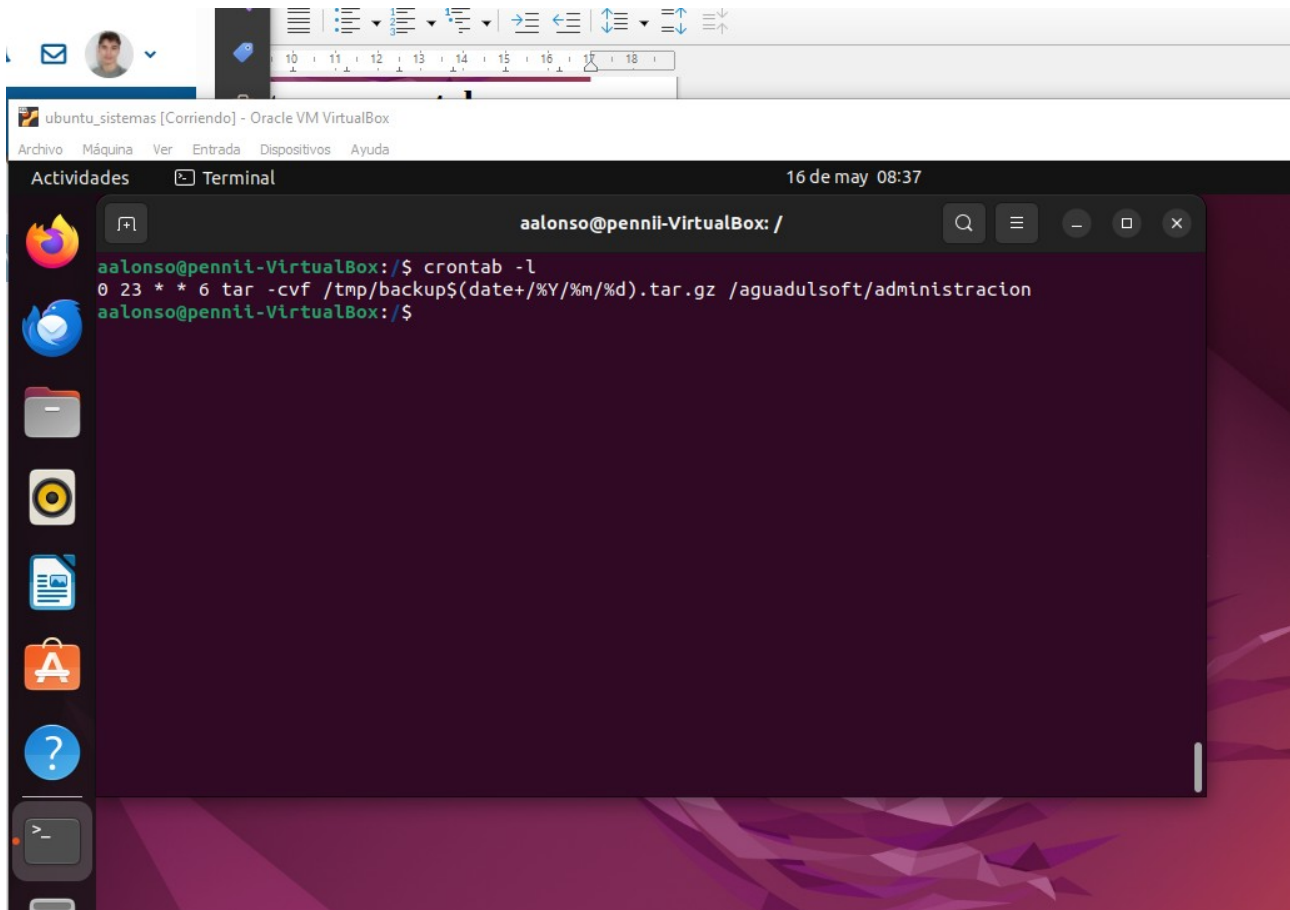
Con el comando **kill** podremos terminar el proceso si introducimos su PID

Actividad 6



Para programar tareas ejecutamos **crontab -e**, y ahí ya podremos hacerlo en el editor.

Programamos la tarea para que se realice todos los sábados a las 23:00 y especificamos donde guardar el directorio de trabajo



Ejecutando el comando **crontab -l**, podremos ver como la tarea fue programada con exito