



**TAREA 08**

**CONFIGURACIÓN DE  
DISPOSITIVOS PARA LA  
INSTALACIÓN DE  
SISTEMAS  
INFORMÁTICOS**

**BASTIONADO DE REDES Y SISTEMAS**

**ALBA MOREJÓN GARCÍA**

**2024/2025**

**Ciberseguridad en Entornos de las Tecnologías de la Información**

El departamento de I+D tiene unos resultados extraordinarios por lo logros conseguidos en el descubrimiento de un nuevo sistema de propulsión eléctrica en los coches fabricados por la compañía. Existen intereses económico de empresas de la competencia y actores externos por hacerse con esta información para poder aplicarla a sus modelos.

El CISO de la compañía quiere que se investigue si el sistema donde se guarda la información sensible y crítica es segura. Por lo que ha pedido que se revisen las medidas de seguridad relativas a estos sistemas.

Buscando:

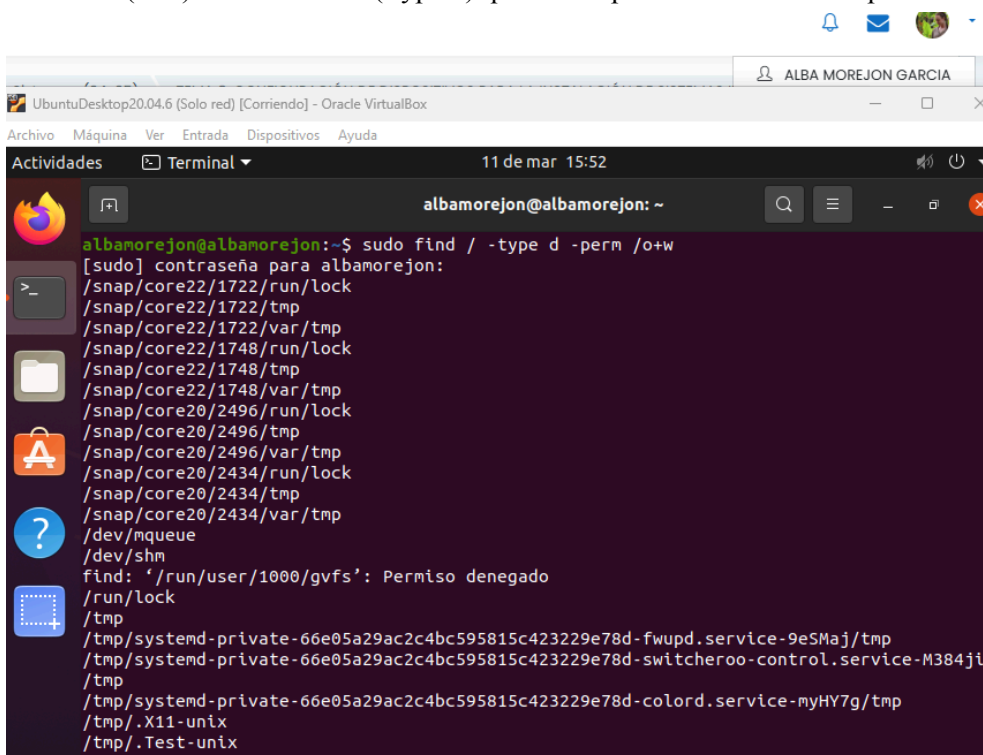
- Los directorios que tienen permisos de escritura.
- Los directorios que tienen permisos de ejecución.
- Ficheros con el SUID o SGID activado, que permitan ejecutar los ficheros con permisos de root, incluyendo si existe algún fichero con permisos de root entre los de la siguiente lista:  
<https://gtfobins.github.io>
- Los ficheros de la variable PATH, comprobando qué usuarios tienen acceso de escritura en esos directorios.
- Las carpetas compartidas mal configuradas que permiten realizar acciones no controladas.
- Las particiones que tienen permisos para ejecutar ficheros y otras características que tienen impacto sobre la seguridad.
- Borrado seguro de archivos.

El escenario se puede realizar con un sistema operativo Linux Ubuntu.

### 1. Los directorios que tienen permisos de escritura

`find / -type d -perm /o+w`

buscamos (find) los directorios (-type d) que tienen permisos de escritura para otros usuarios (-perm /o+w)

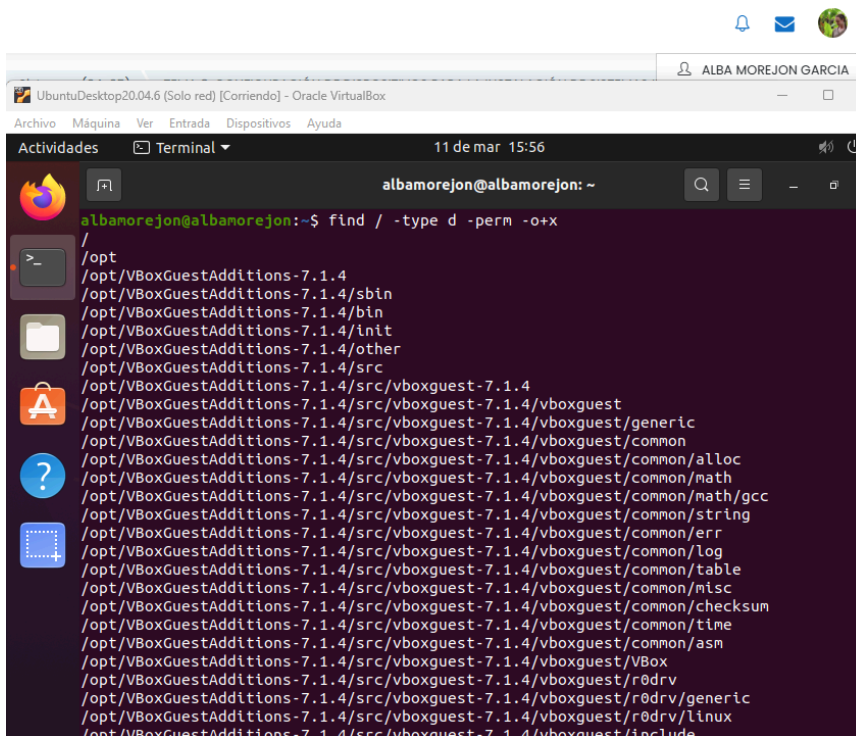


```
albamorejon@albamorejon: ~  
albamorejon@albamorejon:~$ sudo find / -type d -perm /o+w  
[sudo] contraseña para albamorejon:  
/snap/core22/1722/run/lock  
/snap/core22/1722/tmp  
/snap/core22/1722/var/tmp  
/snap/core22/1748/run/lock  
/snap/core22/1748/tmp  
/snap/core22/1748/var/tmp  
/snap/core20/2496/run/lock  
/snap/core20/2496/tmp  
/snap/core20/2496/var/tmp  
/snap/core20/2434/run/lock  
/snap/core20/2434/tmp  
/snap/core20/2434/var/tmp  
/dev/mqueue  
/dev/shm  
find: '/run/user/1000/gvfs': Permiso denegado  
/run/lock  
/tmp  
/tmp/systemd-private-66e05a29ac2c4bc595815c423229e78d-fwupd.service-9eSMaj/tmp  
/tmp/systemd-private-66e05a29ac2c4bc595815c423229e78d-switcheroo-control.service-M384jl  
/tmp  
/tmp/systemd-private-66e05a29ac2c4bc595815c423229e78d-colord.service-myHY7g/tmp  
/tmp/.X11-unix  
/tmp/.Test-unix  
/tmp/.X11-unix
```

## 2. Los directorios que tienen permisos de ejecución.

`find / -type d -perm -o+x`

buscamos (find) los directorios (-type d) que tienen permisos de ejecución para otros usuarios (-perm -o+x)



```
albamorejon@albamorejon:~$ find / -type d -perm -o+x
/
/opt
/opt/VBoxGuestAdditions-7.1.4
/opt/VBoxGuestAdditions-7.1.4/sbin
/opt/VBoxGuestAdditions-7.1.4/bin
/opt/VBoxGuestAdditions-7.1.4/init
/opt/VBoxGuestAdditions-7.1.4/other
/opt/VBoxGuestAdditions-7.1.4/src
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/generic
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/alloc
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/math
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/math/gcc
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/string
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/err
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/log
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/table
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/misc
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/checksum
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/time
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/common/asm
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/VBox
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/r0drv
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/r0drv/generic
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/r0drv/linux
/opt/VBoxGuestAdditions-7.1.4/src/vboxguest-7.1.4/vboxguest/include
```

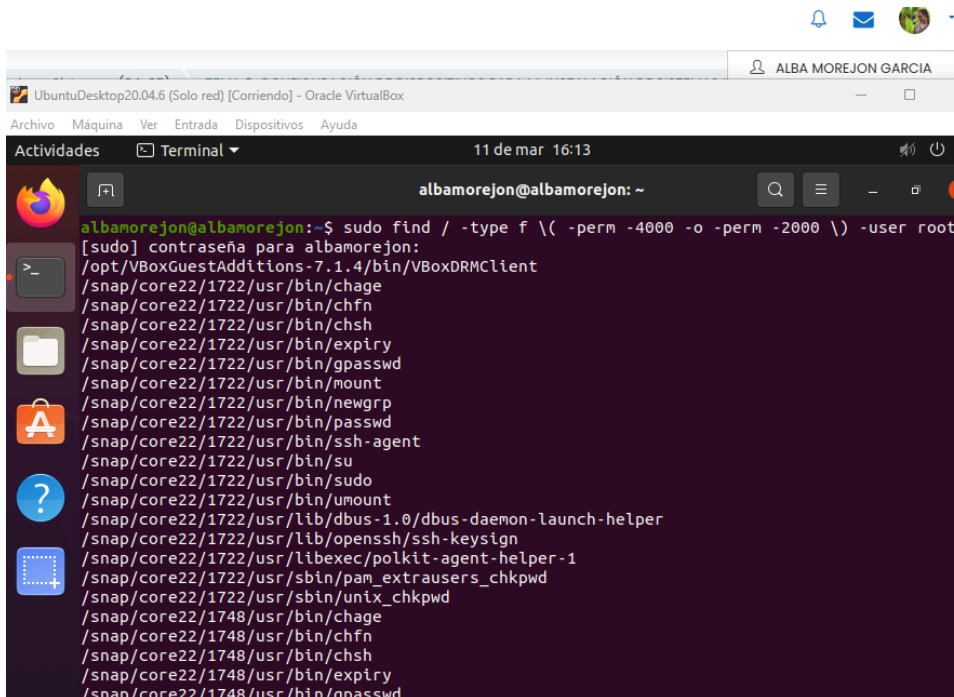
## 3. Ficheros con el SUID o SGID activado, que permitan ejecutar los ficheros con permisos de root, incluyendo si existe algún fichero con permisos de root entre los de la siguiente lista:

<https://gtfobins.github.io>

SUID (/u+s)=4000

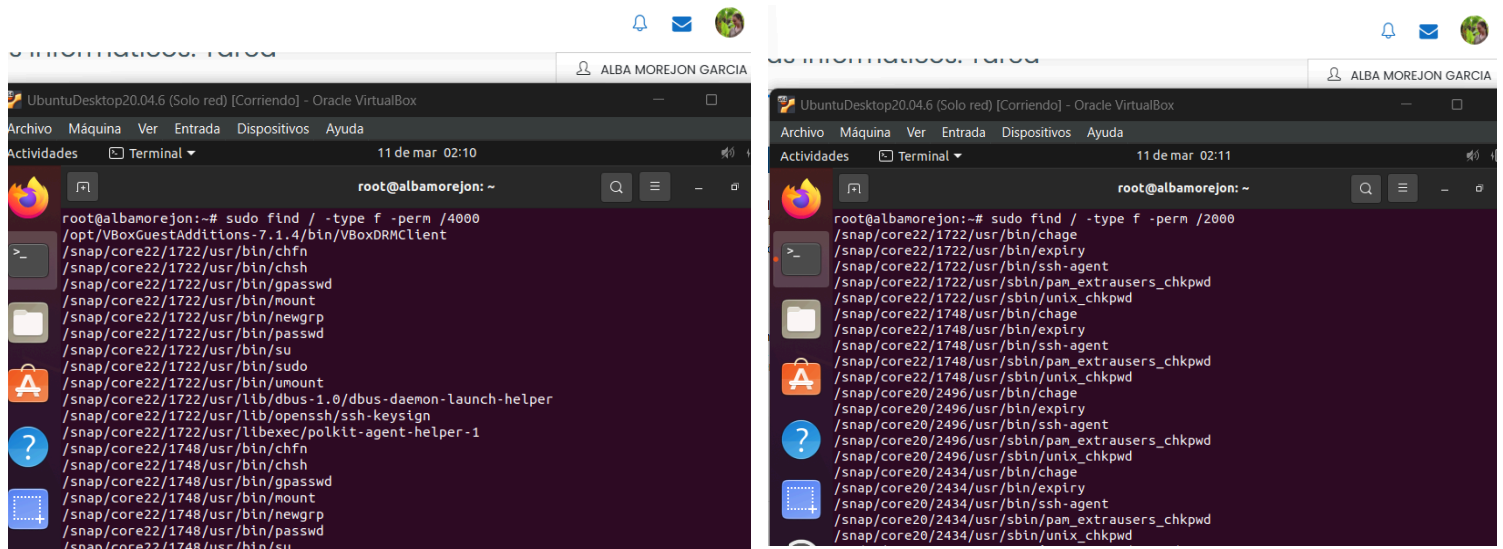
SGID (/g+s)=2000

`sudo find / -type f \( -perm -4000 -o -perm -2000 \) -user root`

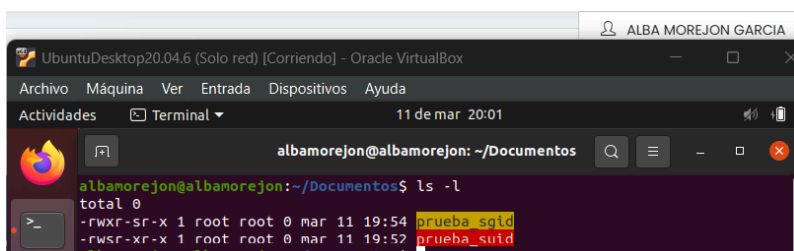
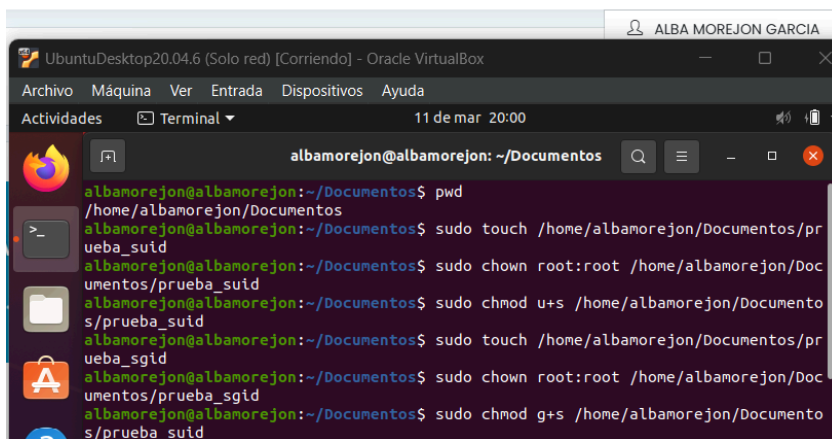


```
albamorejon@albamorejon:~$ sudo find / -type f \( -perm -4000 -o -perm -2000 \) -user root
[sudo] contraseña para albamorejon:
/opt/VBoxGuestAdditions-7.1.4/bin/VBoxDRMClient
/snap/core22/1722/usr/bin/chage
/snap/core22/1722/usr/bin/chfn
/snap/core22/1722/usr/bin/chsh
/snap/core22/1722/usr/bin/expiry
/snap/core22/1722/usr/bin/gpasswd
/snap/core22/1722/usr/bin/mount
/snap/core22/1722/usr/bin/newgrp
/snap/core22/1722/usr/bin/passwd
/snap/core22/1722/usr/bin/ssh-agent
/snap/core22/1722/usr/bin/su
/snap/core22/1722/usr/bin/sudo
/snap/core22/1722/usr/bin/umount
/snap/core22/1722/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core22/1722/usr/lib/openssh/ssh-keysign
/snap/core22/1722/usr/libexec/polkit-agent-helper-1
/snap/core22/1722/usr/sbin/pam_extrausers_chkpwd
/snap/core22/1722/usr/sbin/unix_chkpwd
/snap/core22/1748/usr/bin/chage
/snap/core22/1748/usr/bin/chfn
/snap/core22/1748/usr/bin/chsh
/snap/core22/1748/usr/bin/expiry
/snap/core22/1748/usr/bin/gpasswd
```

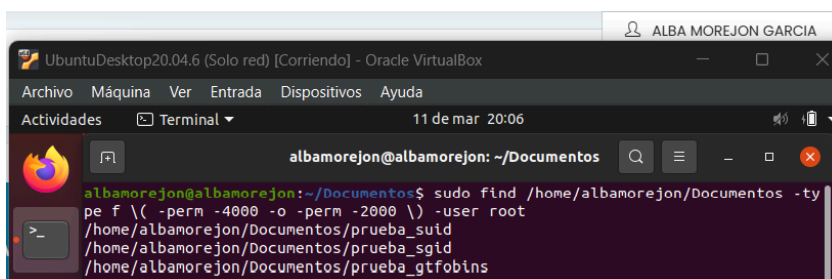
Se podrían buscar los ficheros, suid o sgid por separado



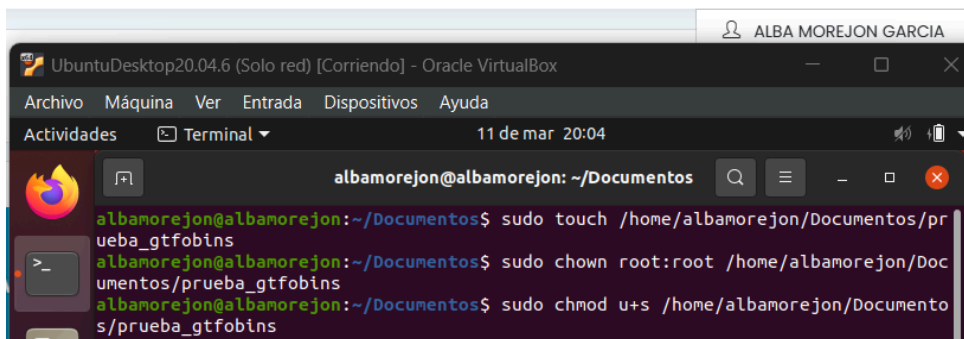
Creamos unos ficheros para hacer la prueba de los ficheros suid y sgid



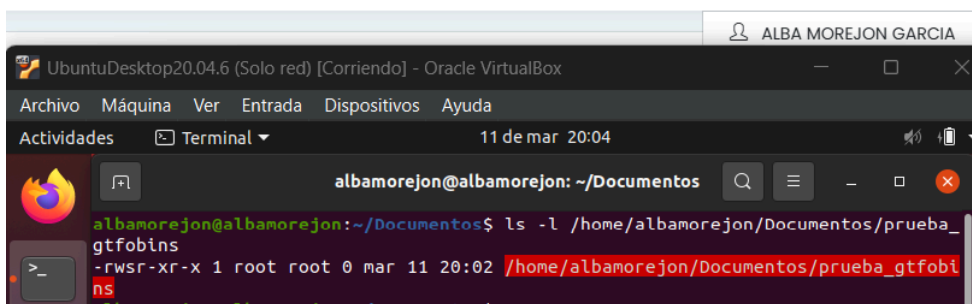
Resultados



Creamos un fichero más, para hacer la prueba de comparar con la lista de [gtfobins.github.io](https://gtfobins.github.io)

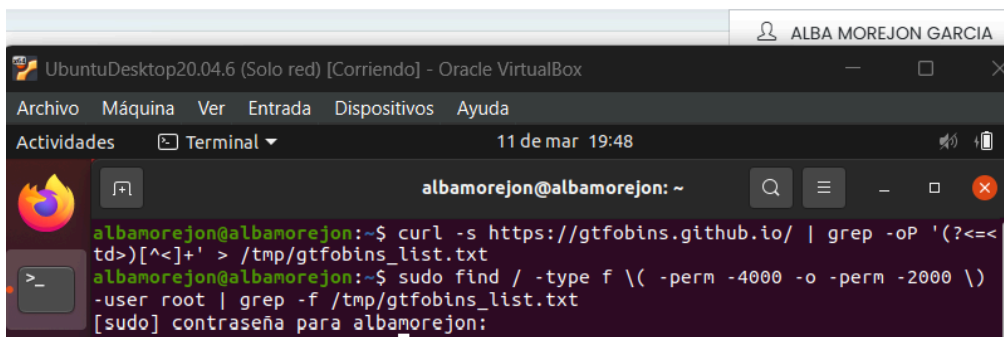


```
albamorejon@albamorejon: ~/Documentos
albamorejon@albamorejon:~/Documentos$ sudo touch /home/albamorejon/Documentos/prueba_gtfobins
albamorejon@albamorejon:~/Documentos$ sudo chown root:root /home/albamorejon/Documentos/prueba_gtfobins
albamorejon@albamorejon:~/Documentos$ sudo chmod u+s /home/albamorejon/Documentos/prueba_gtfobins
```



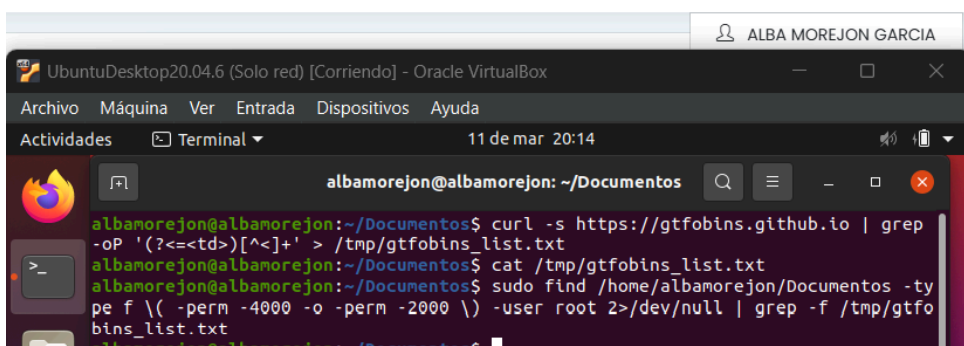
```
albamorejon@albamorejon: ~/Documentos
albamorejon@albamorejon:~/Documentos$ ls -l /home/albamorejon/Documentos/prueba_gtfobins
-rwsr-xr-x 1 root root 0 mar 11 20:02 /home/albamorejon/Documentos/prueba_gtfobins
```

No conseguimos encontrar nada, estamos descargando el contenido de la página (curl), extrayendo los nombres de los ficheros (grep) y guardando la lista en un archivo (>), para después encontrar los ficheros con el SUID o SGID activado y compararlos con la lista creada.



```
albamorejon@albamorejon: ~
albamorejon@albamorejon:~$ curl -s https://gtfobins.github.io/ | grep -oP '(?<=<td>)[^<]+' > /tmp/gtfobins_list.txt
albamorejon@albamorejon:~$ sudo find / -type f \( -perm -4000 -o -perm -2000 \) -user root | grep -f /tmp/gtfobins_list.txt
[sudo] contraseña para albamorejon:
```

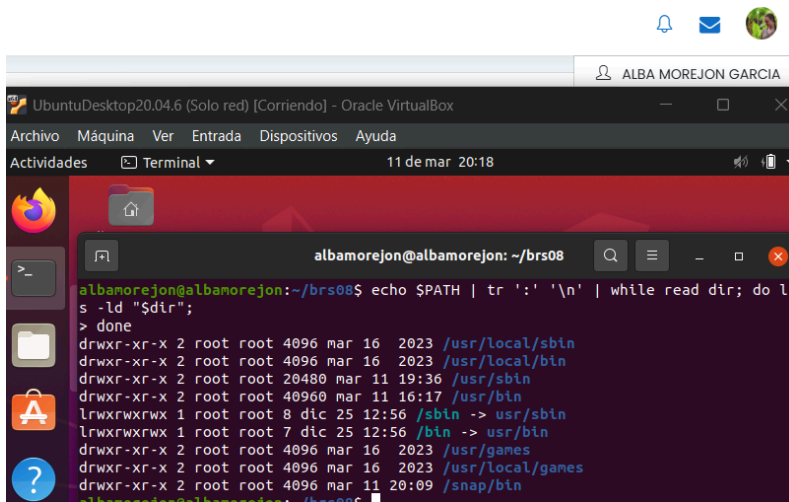
Se probó de diferentes métodos, pero no pude conseguirlo



```
albamorejon@albamorejon: ~/Documentos
albamorejon@albamorejon:~/Documentos$ curl -s https://gtfobins.github.io | grep -oP '(?<=<td>)[^<]+' > /tmp/gtfobins_list.txt
albamorejon@albamorejon:~/Documentos$ cat /tmp/gtfobins_list.txt
albamorejon@albamorejon:~/Documentos$ sudo find /home/albamorejon/Documentos -type f \( -perm -4000 -o -perm -2000 \) -user root 2>/dev/null | grep -f /tmp/gtfobins_list.txt
albamorejon@albamorejon:~/Documentos$
```

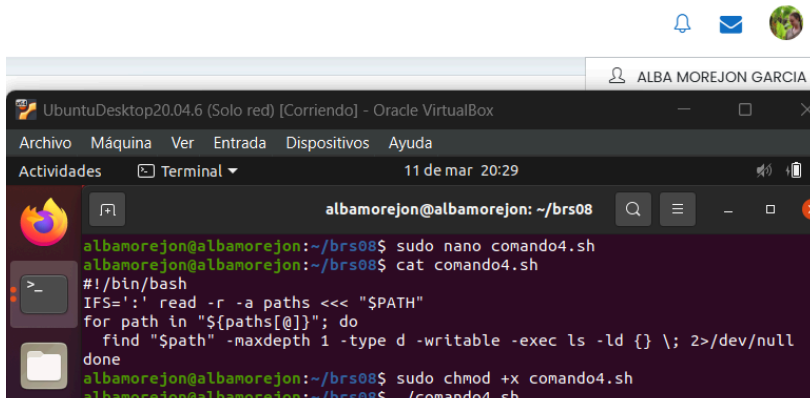
#### 4. Las carpetas compartidas mal configuradas que permiten realizar acciones no controladas.

Listamos los permisos de los directorios del PATH, pudiendo ver los de escritura

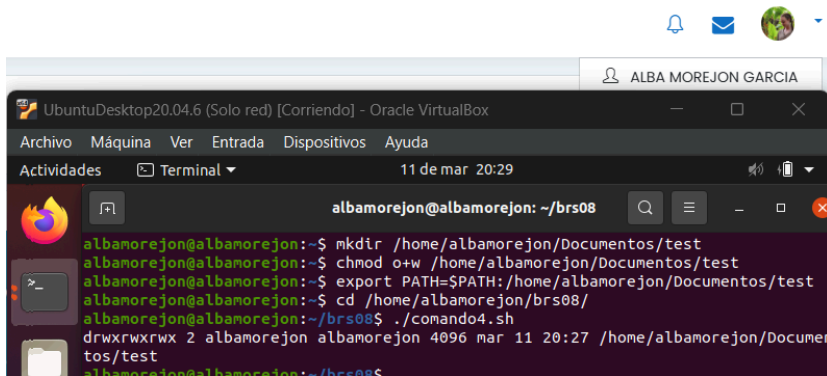


```
albamorejon@albamorejon: ~/brs08
albamorejon@albamorejon:~/brs08$ echo $PATH | tr ':' '\n' | while read dir; do l
s -ld "$dir";
> done
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/local/sbin
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/local/bin
drwxr-xr-x 2 root root 20480 mar 11 19:36 /usr/sbin
drwxr-xr-x 2 root root 40960 mar 11 16:17 /usr/bin
lrwxrwxrwx 1 root root 8 dic 25 12:56 /sbin -> usr/sbin
lrwxrwxrwx 1 root root 7 dic 25 12:56 /bin -> usr/bin
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/games
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/local/games
drwxr-xr-x 2 root root 4096 mar 11 20:09 /snap/bin
```

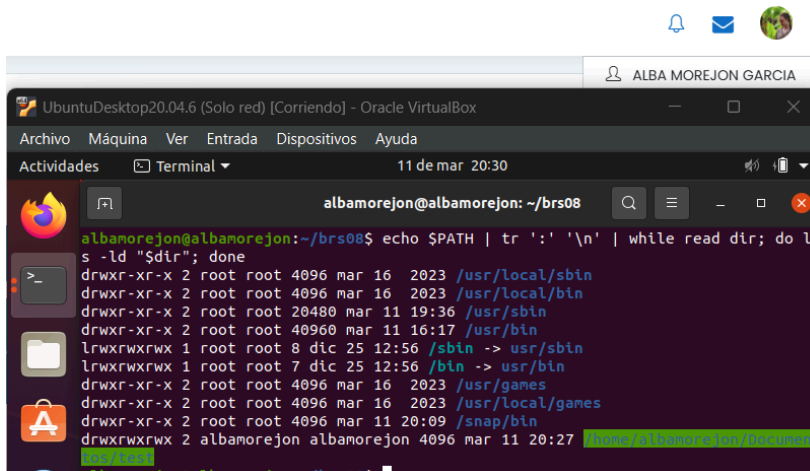
En caso de que se quiera hacer un script para ello:



```
albamorejon@albamorejon:~/brs08$ sudo nano comando4.sh
albamorejon@albamorejon:~/brs08$ cat comando4.sh
#!/bin/bash
IFS=: read -r -a paths <<< "$PATH"
for path in "${paths[@]}"; do
    find "$path" -maxdepth 1 -type d -writable -exec ls -ld {} \; 2>/dev/null
done
albamorejon@albamorejon:~/brs08$ sudo chmod +x comando4.sh
albamorejon@albamorejon:~/brs08$ ./comando4.sh
```



```
albamorejon@albamorejon:~$ mkdir /home/albamorejon/Documentos/test
albamorejon@albamorejon:~$ chmod o+w /home/albamorejon/Documentos/test
albamorejon@albamorejon:~$ export PATH=$PATH:/home/albamorejon/Documentos/test
albamorejon@albamorejon:~$ cd /home/albamorejon/brs08/
albamorejon@albamorejon:~/brs08$ ./comando4.sh
drwxrwxrwx 2 albamorejon albamorejon 4096 mar 11 20:27 /home/albamorejon/Documen
tos/test
albamorejon@albamorejon:~/brs08$
```

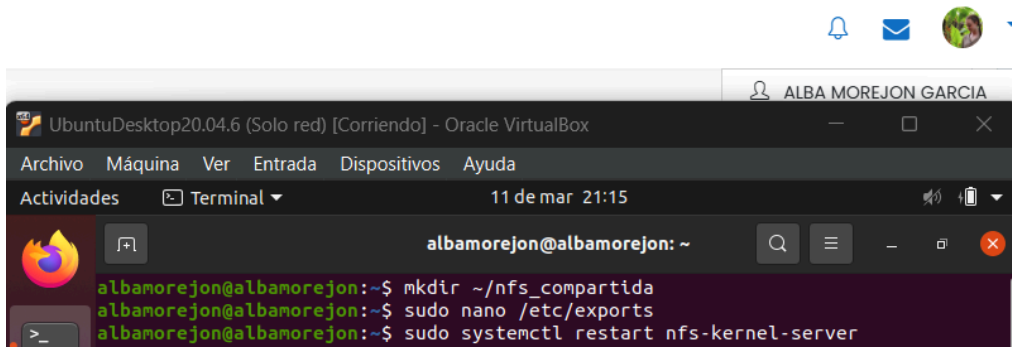


```
albamorejon@albamorejon:~/brs08$ echo $PATH | tr ':' '\n' | while read dir; do l
s -ld "$dir"; done
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/local/sbin
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/local/bin
drwxr-xr-x 2 root root 20480 mar 11 19:36 /usr/sbin
drwxr-xr-x 2 root root 40960 mar 11 16:17 /usr/bin
lrwxrwxrwx 1 root root 8 dic 25 12:56 /sbin -> usr/sbin
lrwxrwxrwx 1 root root 7 dic 25 12:56 /bin -> usr/bin
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/games
drwxr-xr-x 2 root root 4096 mar 16 2023 /usr/local/games
drwxr-xr-x 2 root root 4096 mar 11 20:09 /snap/bin
drwxrwxrwx 2 albamorejon albamorejon 4096 mar 11 20:27 /home/albamorejon/Documen
tos/test
albamorejon@albamorejon:~/brs08$
```

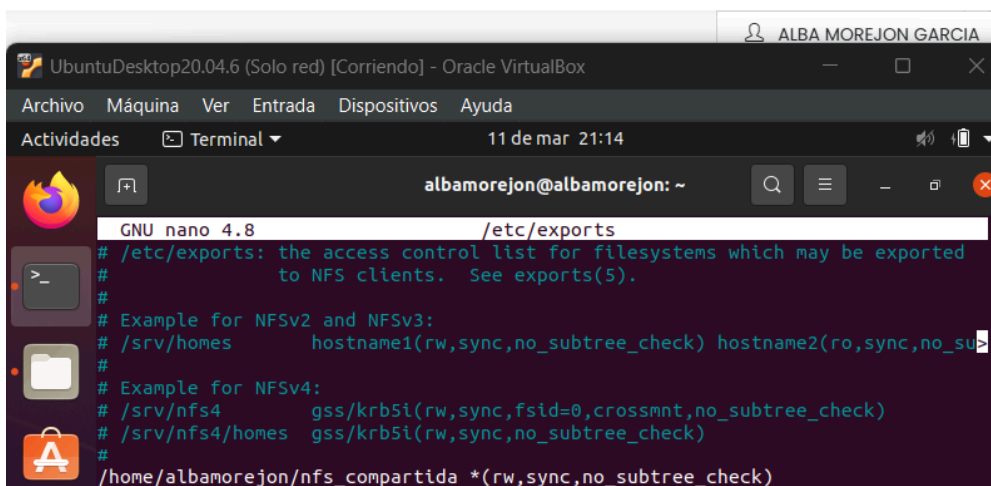


## 5. Las carpetas compartidas mal configuradas que permiten realizar acciones no controladas.

Probamos a hacer una carpeta compartida y conocer cómo se comparte y da los permisos

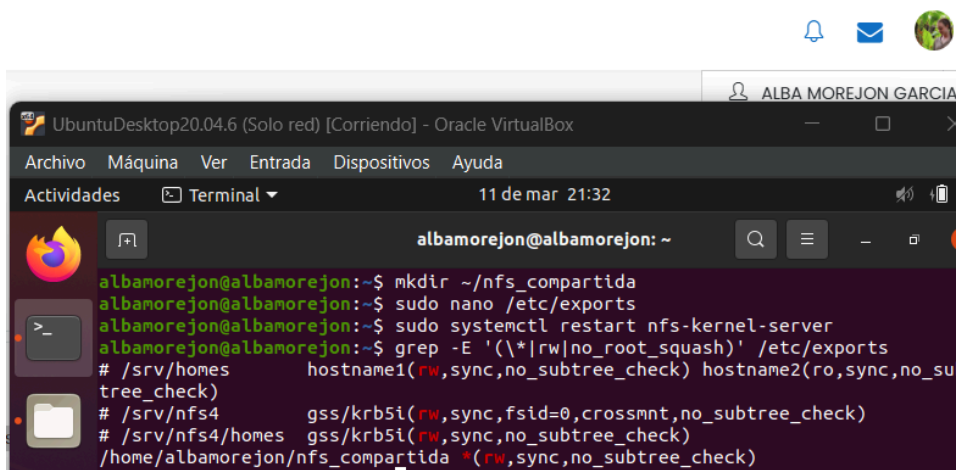


```
albamorejon@albamorejon: ~  
albamorejon@albamorejon:~$ mkdir ~/nfs_compartida  
albamorejon@albamorejon:~$ sudo nano /etc/exports  
albamorejon@albamorejon:~$ sudo systemctl restart nfs-kernel-server
```



```
GNU nano 4.8 /etc/exports  
# /etc/exports: the access control list for filesystems which may be exported  
# to NFS clients.  See exports(5).  
#  
# Example for NFSv2 and NFSv3:  
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)  
#  
# Example for NFSv4:  
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)  
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)  
#  
/home/albamorejon/nfs_compartida *(rw,sync,no_subtree_check)
```

Buscamos las carpetas compartidas que pueden tener configuraciones más débiles



```
albamorejon@albamorejon: ~  
albamorejon@albamorejon:~$ mkdir ~/nfs_compartida  
albamorejon@albamorejon:~$ sudo nano /etc/exports  
albamorejon@albamorejon:~$ sudo systemctl restart nfs-kernel-server  
albamorejon@albamorejon:~$ grep -E '(\[*|rw|no_root_squash)' /etc/exports  
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)  
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)  
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)  
/home/albamorejon/nfs_compartida *(rw,sync,no_subtree_check)
```

Algunas configuraciones de NFS pueden ser inseguras y permitir acciones no controladas. Aquí hay algunas opciones que pueden representar un problema de seguridad:

1. Permisos demasiado amplios (rw para todos), la opción read/write permite lectura y escritura a todos los clientes. Esto puede ser peligroso si no se controla adecuadamente, ya que cualquier cliente en la red puede modificar los archivos compartidos.

Ej: /home/usuario/nfs\_compartida \*(rw,sync,no\_subtree\_check)

Solución: Limita el acceso a un rango específico de IP y usa root\_squash para mapear las solicitudes del usuario root en los clientes a un usuario no privilegiado en el servidor.

Ej: /home/usuario/nfs\_compartida 192.168.1.0/24 (rw,sync,no\_subtree\_check,root\_squash)

2. Acceso sin restricciones (\*), usar \* para permitir acceso a todos los clientes en la red puede ser inseguro. Es mejor especificar direcciones IP o rangos de IP específicos. El problema de seguridad es que permite que cualquier dispositivo en la red acceda a la carpeta compartida, lo que puede incluir dispositivos no autorizados.

Ej.: /home/usuario/nfs\_compartida \*(rw,sync,no\_subtree\_check)

Solución: Limita el acceso a un rango específico de IP.

Ej.: /home/tu\_usuario/nfs\_compartida 192.168.1.0/24(rw,sync,no\_subtree\_check)

3. Sin autenticación (no\_root\_squash):

La opción no\_root\_squash permite que los usuarios root en los clientes tengan privilegios root en el servidor NFS. Esto puede ser un gran riesgo de seguridad, ya que permite a los usuarios root en los clientes realizar cualquier acción en el servidor.

Ej.: /home/usuario/nfs\_compartida 192.168.1.0/24(rw,sync,no\_subtree\_check,no\_root\_squash)

Solución: Usa root\_squash para mapear las solicitudes del usuario root en los clientes a un usuario no privilegiado en el servidor.

Ej.: /home/usuario/nfs\_compartida 192.168.1.0/24(rw,sync,no\_subtree\_check,root\_squash)

## 6. Las particiones que tienen permisos para ejecutar ficheros y otras características que tienen impacto sobre la seguridad.

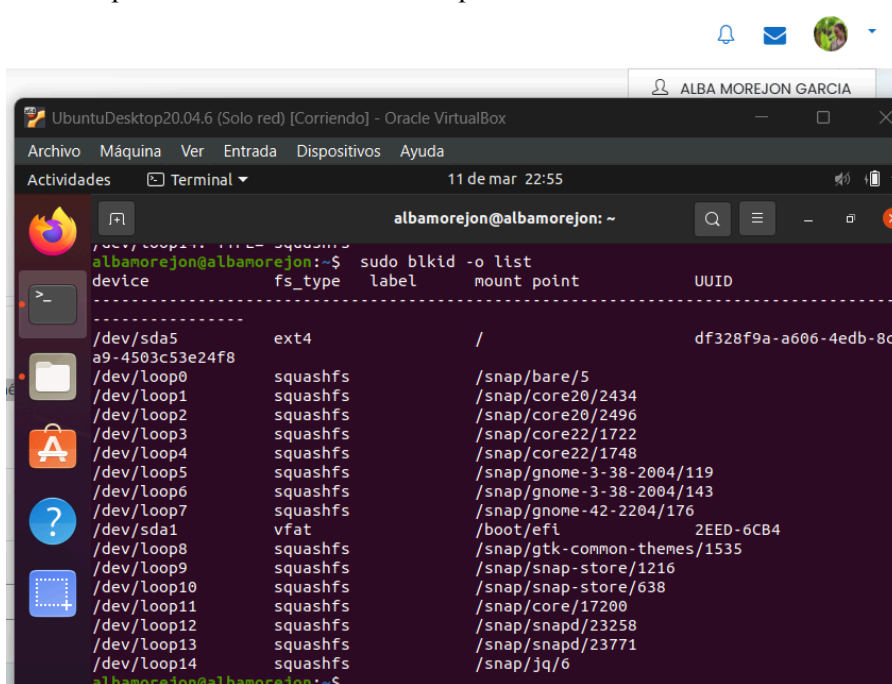
Como en directorio /dev, donde se guarda la configuración e información de las particiones del disco duro. Por tanto la forma más rápida sería esta primera opción:

Con este comando podremos listar todas las particiones y sus puntos de montaje

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
loop0	7:0	0	4K	1	loop	/snap/bare/5
loop1	7:1	0	63,7M	1	loop	/snap/core20/2434
loop2	7:2	0	63,8M	1	loop	/snap/core20/2496
loop3	7:3	0	73,9M	1	loop	/snap/core22/1722
loop4	7:4	0	73,9M	1	loop	/snap/core22/1748
loop5	7:5	0	346,3M	1	loop	/snap/gnome-3-38-2004/119
loop6	7:6	0	349,7M	1	loop	/snap/gnome-3-38-2004/143
loop7	7:7	0	505,1M	1	loop	/snap/gnome-42-2204/176
loop8	7:8	0	91,7M	1	loop	/snap/gtk-common-themes/1535
loop9	7:9	0	12,2M	1	loop	/snap/snap-store/1216
loop10	7:10	0	46M	1	loop	/snap/snap-store/638
loop11	7:11	0	104,2M	1	loop	/snap/core/17200
loop12	7:12	0	44,3M	1	loop	/snap/snapd/23258
loop13	7:13	0	44,5M	1	loop	/snap/snapd/23771
loop14	7:14	0	240K	1	loop	/snap/jq/6
sda	8:0	0	30G	0	disk	
├─sda1	8:1	0	512M	0	part	/boot/efi
├─sda2	8:2	0	1K	0	part	
└─sda5	8:5	0	29,5G	0	part	/
sr0	11:0	1	1024M	0	rom	

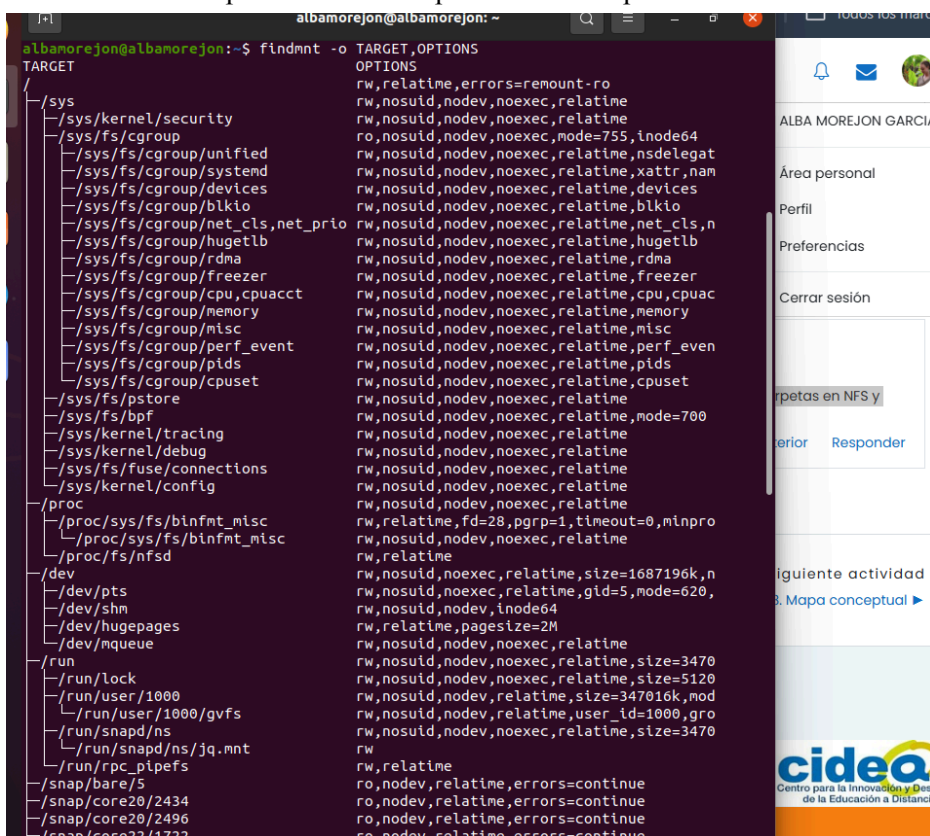


También podremos usar este comando para ello



```
albamorejon@albamorejon:~$ sudo blkid -o list
device      fs_type    label      mount point      UUID
-----
/dev/sda5   ext4       /          /                 df328f9a-a606-4edb-8c
a9-4503c53e24f8
/dev/loop0  squashfs  /snap/bare/5
/dev/loop1  squashfs  /snap/core20/2434
/dev/loop2  squashfs  /snap/core20/2496
/dev/loop3  squashfs  /snap/core22/1722
/dev/loop4  squashfs  /snap/core22/1748
/dev/loop5  squashfs  /snap/gnome-3-38-2004/119
/dev/loop6  squashfs  /snap/gnome-3-38-2004/143
/dev/loop7  squashfs  /snap/gnome-42-2204/176
/dev/sda1   vfat      /boot/efi   2EED-6CB4
/dev/loop8  squashfs  /snap/gtk-common-themes/1535
/dev/loop9  squashfs  /snap/snap-store/1216
/dev/loop10 squashfs  /snap/snap-store/638
/dev/loop11 squashfs  /snap/core/17200
/dev/loop12 squashfs  /snap/snapd/23258
/dev/loop13 squashfs  /snap/snapd/23771
/dev/loop14 squashfs  /snap/jq/6
```

Con este comando podremos ver los permisos de las particiones



```
albamorejon@albamorejon:~$ findmnt -o TARGET,OPTIONS
TARGET      OPTIONS
-----
/sys        rw,relatime,errors=remount-ro
/sys/kernel/security rw,nosuid,nodev,noexec,relatime
/sys/fs/cgroup ro,nosuid,nodev,noexec,mode=755,inode64
/sys/fs/cgroup/unified rw,nosuid,nodev,noexec,relatime,nsdelegat
/sys/fs/cgroup/systemd rw,nosuid,nodev,noexec,relatime,xattr,nam
/sys/fs/cgroup/devices rw,nosuid,nodev,noexec,relatime,devices
/sys/fs/cgroup/blkio rw,nosuid,nodev,noexec,relatime,blkio
/sys/fs/cgroup/net_cls,net_prio rw,nosuid,nodev,noexec,relatime,net_cls,n
/sys/fs/cgroup/hugetlb rw,nosuid,nodev,noexec,relatime,hugetlb
/sys/fs/cgroup/rdma rw,nosuid,nodev,noexec,relatime,rdma
/sys/fs/cgroup/freezer rw,nosuid,nodev,noexec,relatime,freezer
/sys/fs/cgroup/cpu,cpuacct rw,nosuid,nodev,noexec,relatime,cpu,cpuac
/sys/fs/cgroup/memory rw,nosuid,nodev,noexec,relatime,memory
/sys/fs/cgroup/misc rw,nosuid,nodev,noexec,relatime,misc
/sys/fs/cgroup/perf_event rw,nosuid,nodev,noexec,relatime,perf_even
/sys/fs/cgroup/pids rw,nosuid,nodev,noexec,relatime,pids
/sys/fs/cgroup/cpuset rw,nosuid,nodev,noexec,relatime,cpuset
/sys/fs/pstore rw,nosuid,nodev,noexec,relatime
/sys/fs/bpf rw,nosuid,nodev,noexec,relatime,mode=700
/sys/kernel/tracing rw,nosuid,nodev,noexec,relatime
/sys/kernel/debug rw,nosuid,nodev,noexec,relatime
/sys/fs/fuse/connections rw,nosuid,nodev,noexec,relatime
/sys/kernel/config rw,nosuid,nodev,noexec,relatime
/proc       rw,nosuid,nodev,noexec,relatime
/proc/sys/fs/binfmt_misc rw,relatime,fd=28,pgroup=1,timeout=0,minpro
/proc/sys/fs/binfmt_misc rw,nosuid,nodev,noexec,relatime
/proc/fs/nfsd rw,relatime
/dev        rw,nosuid,noexec,relatime,size=1687196k,n
/dev/pts    rw,nosuid,noexec,relatime,gid=5,mode=620,
/dev/shm    rw,nosuid,nodev,inode64
/dev/hugepages rw,relatime,pagesize=2M
/dev/mqueue rw,nosuid,nodev,noexec,relatime
/run        rw,nosuid,nodev,noexec,relatime,size=3470
/run/lock   rw,nosuid,nodev,noexec,relatime,size=5120
/run/user/1000 rw,nosuid,nodev,relatime,size=347016k,mod
/run/user/1000/gvfs rw,nosuid,nodev,relatime,user_id=1000,gro
/run/snapd/ns rw,nosuid,nodev,noexec,relatime,size=3470
/run/snapd/ns/jq.mnt rw
/run/rpc_pipefs rw,relatime
/snap/bare/5 ro,nodev,relatime,errors=continue
/snap/core20/2434 ro,nodev,relatime,errors=continue
/snap/core20/2496 ro,nodev,relatime,errors=continue
/snap/core22/1722 ro,nodev,relatime,errors=continue
```

## 7. Borrado seguro de archivos.

Hay diferentes tipos de borrado seguro, en este caso hemos elegido wipe

Con el comando wipe borrará un archivo pero antes lo sobrescribirá varias veces para que sea imposible recuperar los datos.

```
sudo apt install wipe
```

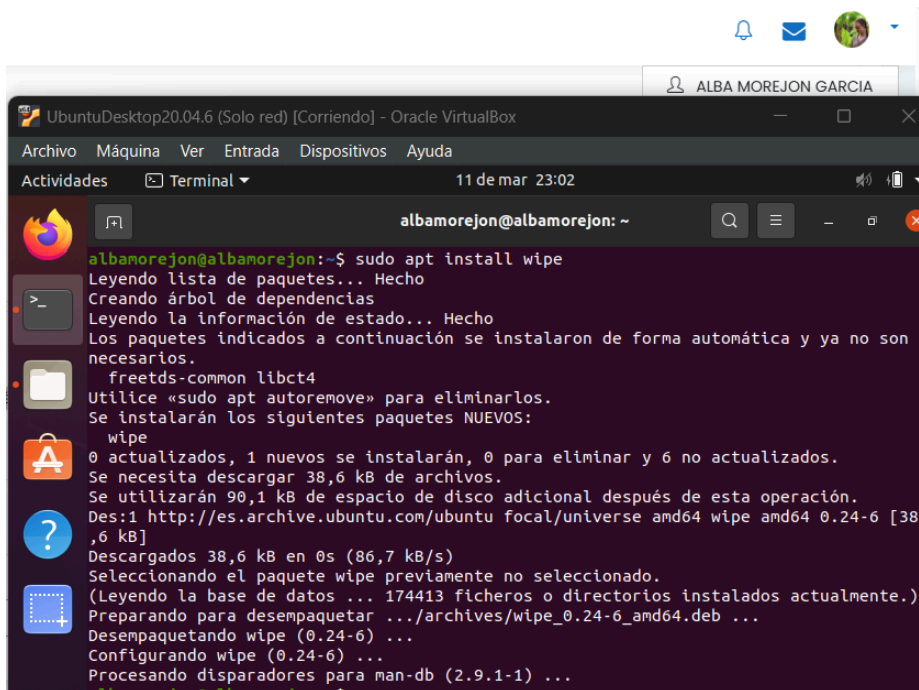
```
wipe archivo.txt
```

Podemos borrar una carpeta de manera segura y de forma recursiva en todas sus subcarpetas utilizando `-r`  
`wipe -r carpeta/`

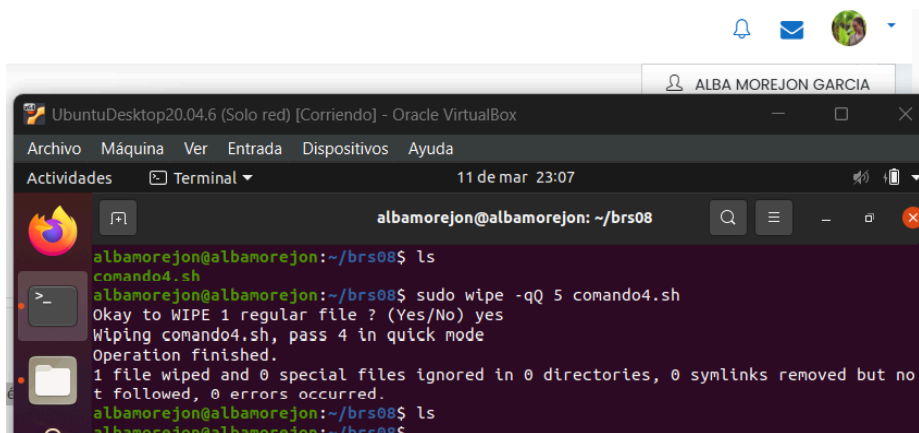
Si queremos decidir la cantidad de veces que se va a sobrescribir el archivo antes de borrarse podemos añadir `-qQ`

Con `-q` indicamos que haga un borrado rápido haciendo solo 4 sobrescripciones por defecto sobre el archivo. Al añadir `-Q` le indicamos cuantas sobrescripciones queremos hacer.

`wipe -qQ 5 archivo.txt`



```
albamorejon@albamorejon:~$ sudo apt install wipe
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  freetds-common libct4
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
  wipe
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 38,6 kB de archivos.
Se utilizarán 90,1 kB de espacio de disco adicional después de esta operación.
Des:1 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 wipe amd64 0.24-6 [38,6 kB]
Descargados 38,6 kB en 0s (86,7 kB/s)
Seleccionando el paquete wipe previamente no seleccionado.
(Leyendo la base de datos ... 174413 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../archives/wipe_0.24-6_amd64.deb ...
Desempaquetando wipe (0.24-6) ...
Configurando wipe (0.24-6) ...
Procesando disparadores para man-db (2.9.1-1) ...
albamorejon@albamorejon:~$
```



```
albamorejon@albamorejon:~/brs08$ ls
comando4.sh
albamorejon@albamorejon:~/brs08$ sudo wipe -qQ 5 comando4.sh
Okay to WIPE 1 regular file ? (Yes/No) yes
Wiping comando4.sh, pass 4 in quick mode
Operation finished.
1 file wiped and 0 special files ignored in 0 directories, 0 symlinks removed but no
t followed, 0 errors occurred.
albamorejon@albamorejon:~/brs08$ ls
albamorejon@albamorejon:~/brs08$
```