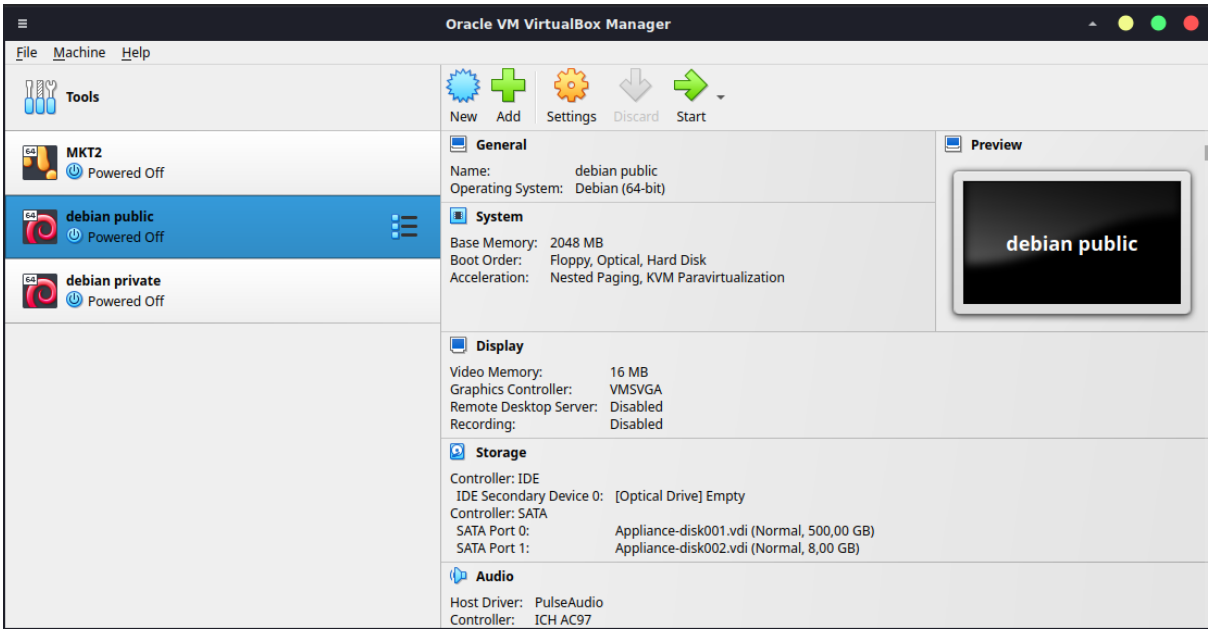
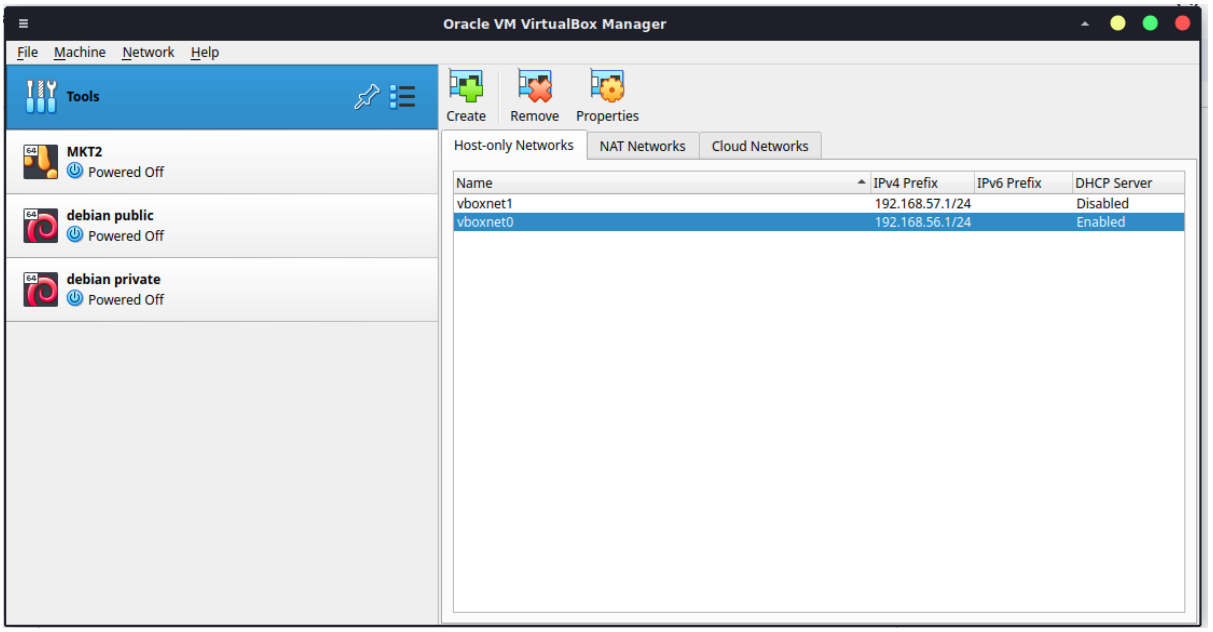
Entrega clase 1 lab en clase

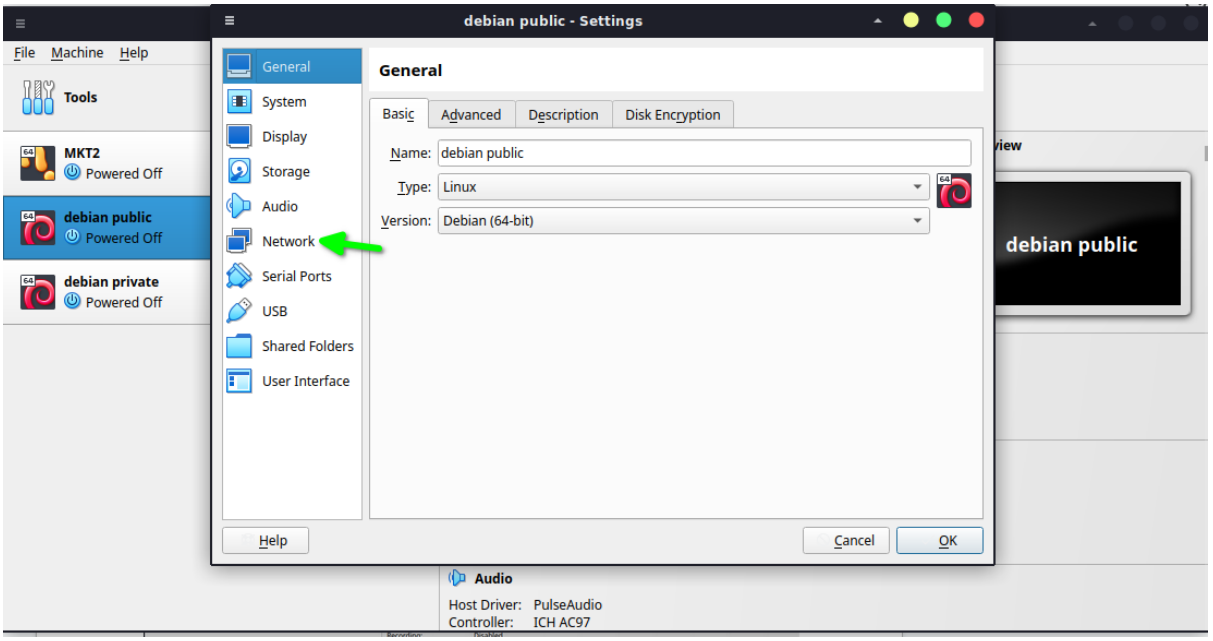
Parte 1: Explorar sistemas de archivos en Linux

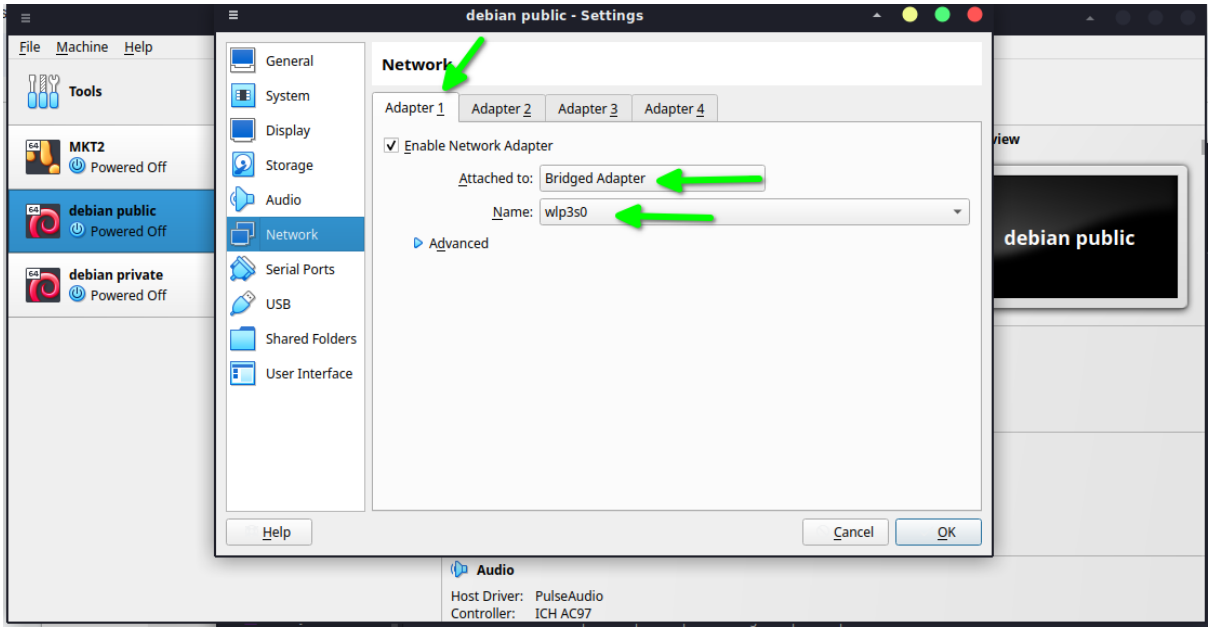
Entorno

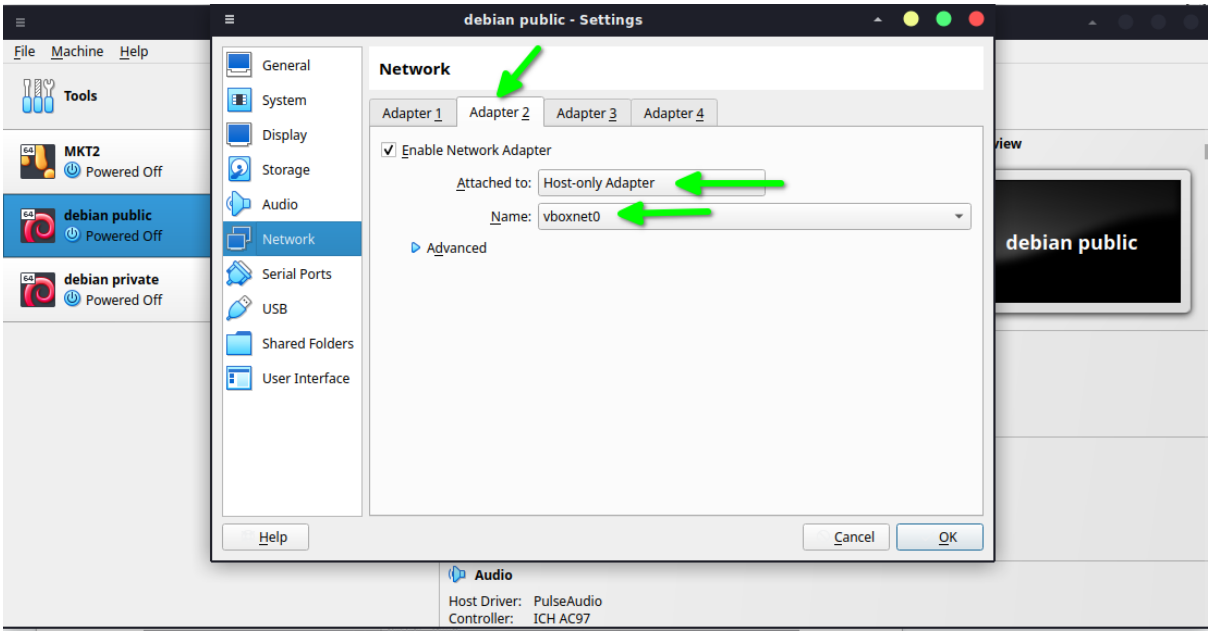




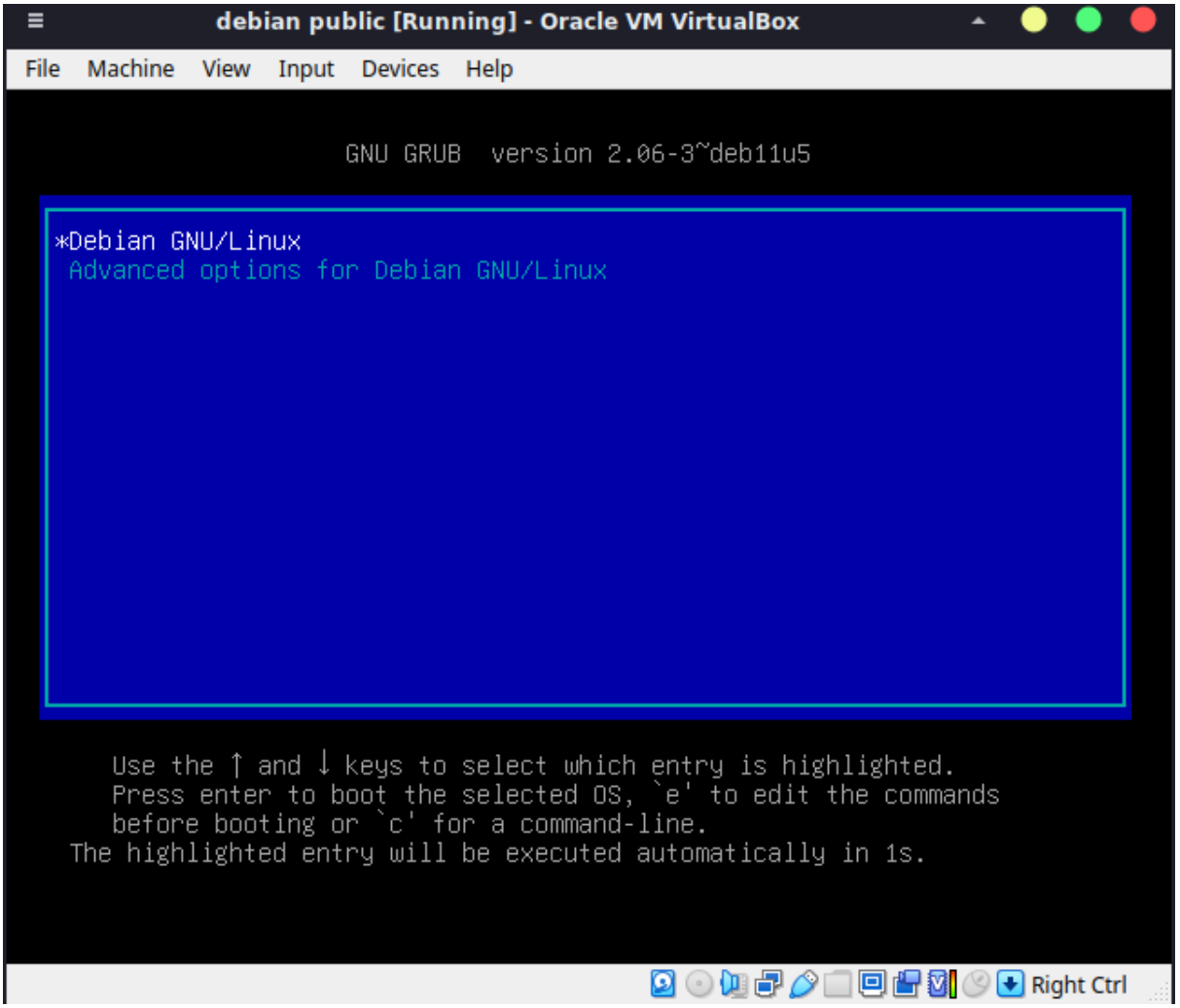
Configurar la red, debe realizarse en ambas máquinas virtuales

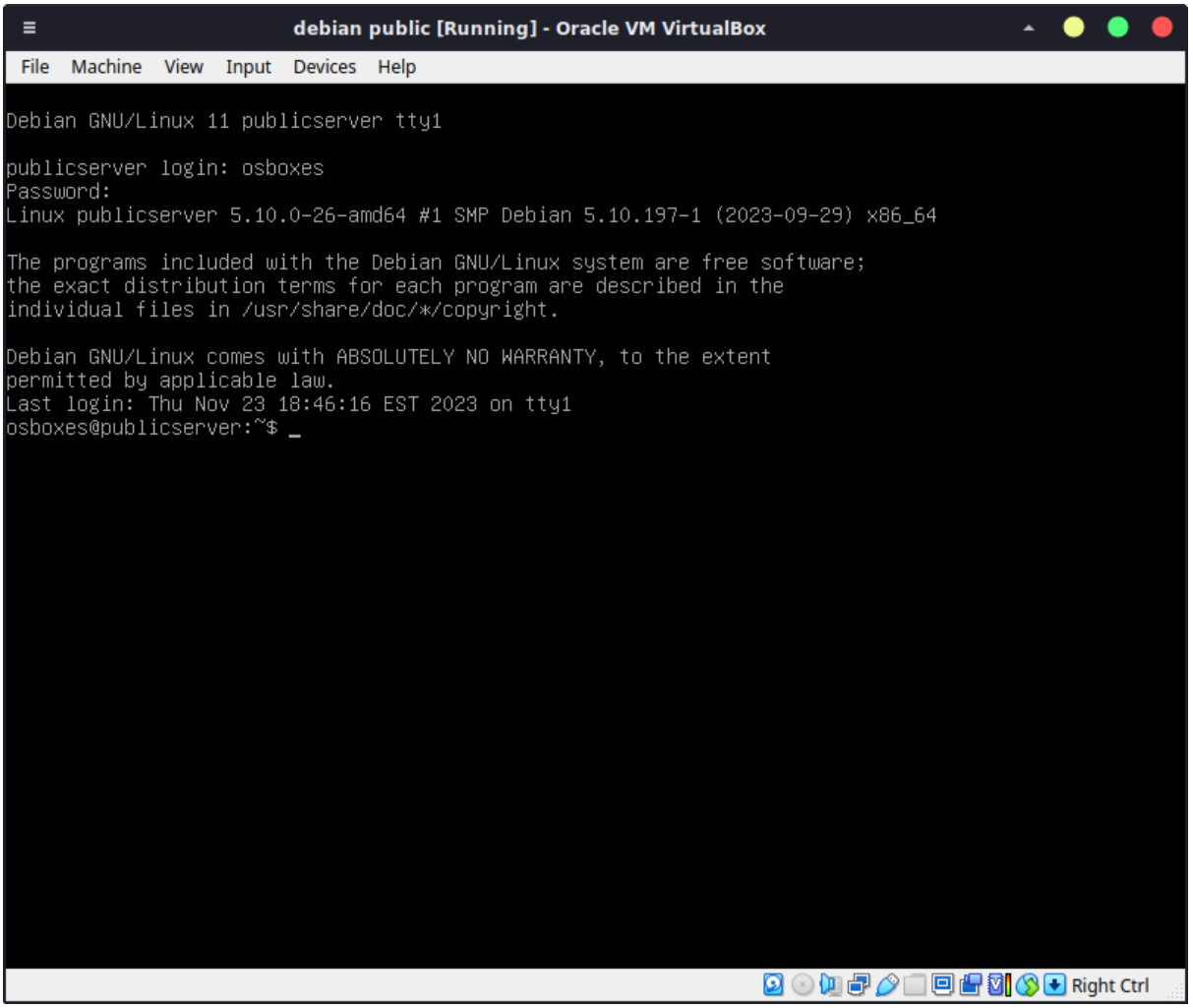

debian public - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
User Interface
General
Basic
Advanced
Description
Disk Encryption
Name: debian public
Type: Linux
Version: Debian (64-bit)
Help
Cancel
OK
File Machine Help
Tools
MKT2
Powered Off
debian public
Powered Off
debian private
Powered Off
debian public
Audio
Host Driver: PulseAudio
Controller: ICH AC97


debian public - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
User Interface
Network
Adapter 1
Adapter 2
Adapter 3
Adapter 4
Enable Network Adapter
Attached to: Bridged Adapter
Name: wlp3s0
Advanced
Help
Cancel
OK
File Machine Help
Tools
MKT2
Powered Off
debian public
Powered Off
debian private
Powered Off
debian public
Audio
Host Driver: PulseAudio
Controller: ICH AC97



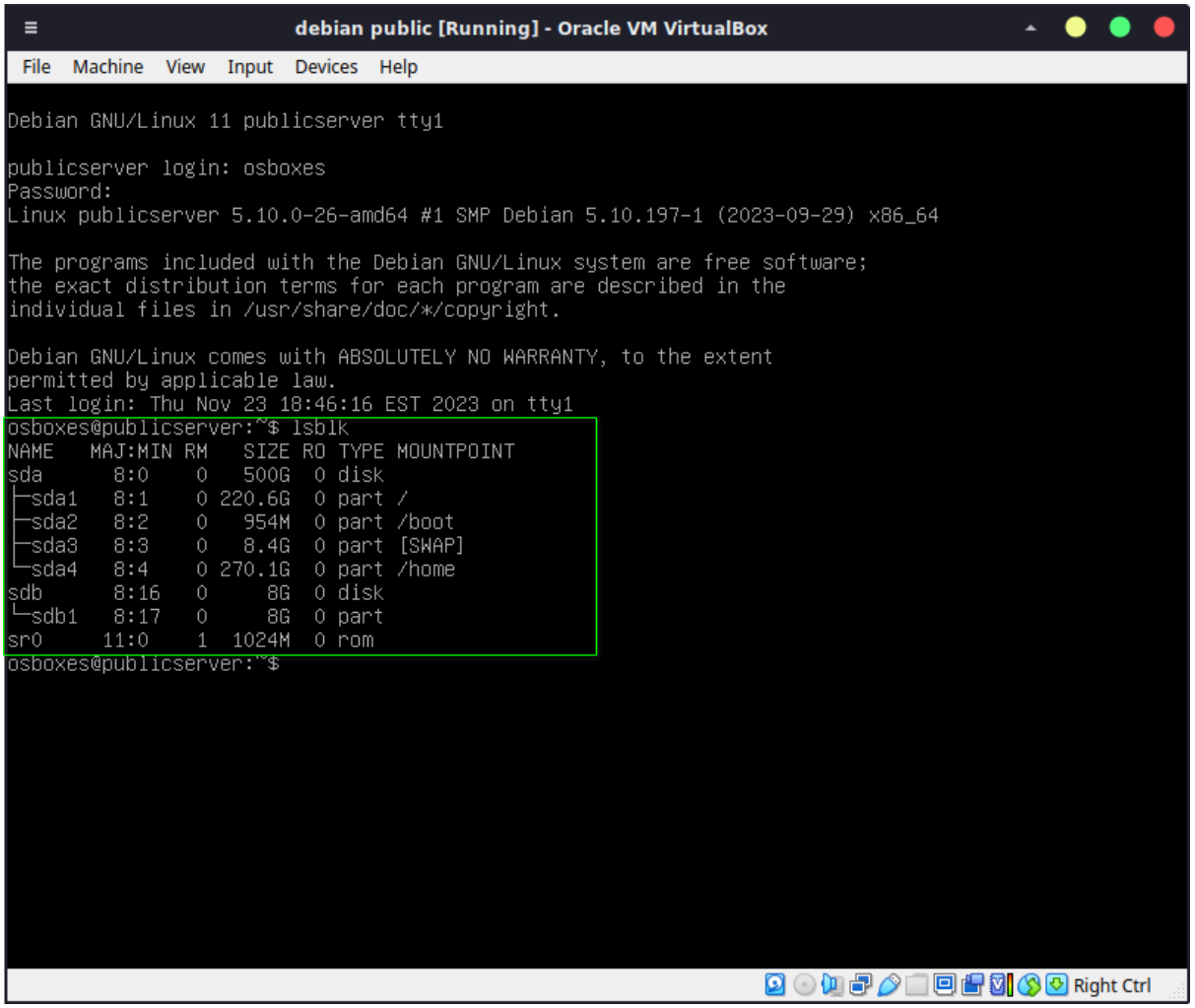
Ingresamos a la maquina virtual debian public
User: osboxes
Password: osboxes.org



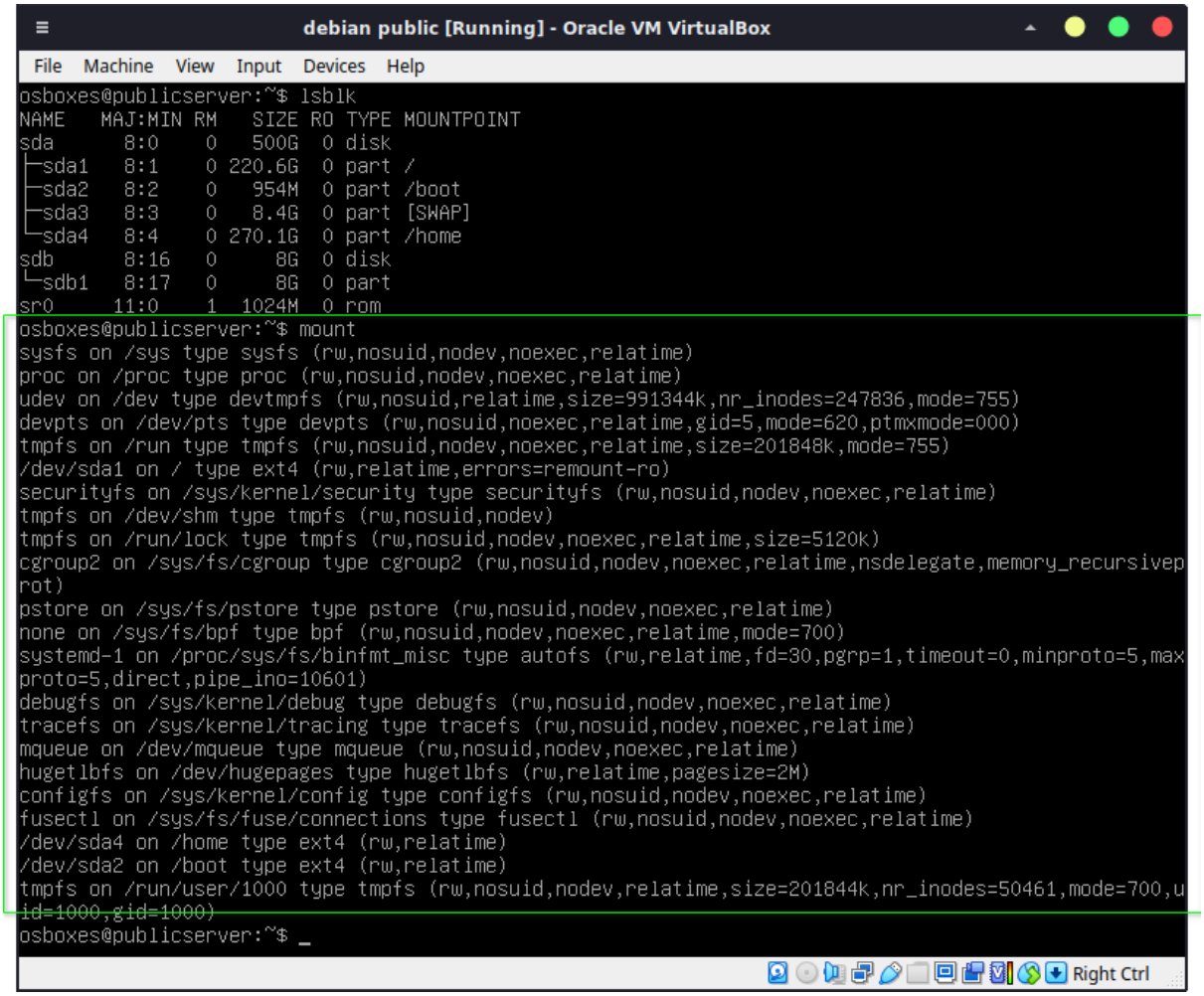

debian public [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Debian GNU/Linux 11 publicserver tty1
publicserver login: osboxes
Password:
Linux publicserver 5.10.0-26-amd64 #1 SMP Debian 5.10.197-1 (2023-09-29) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov 23 18:46:16 EST 2023 on tty1
osboxes@publicserver:~$ _
Right Ctrl

Paso 2

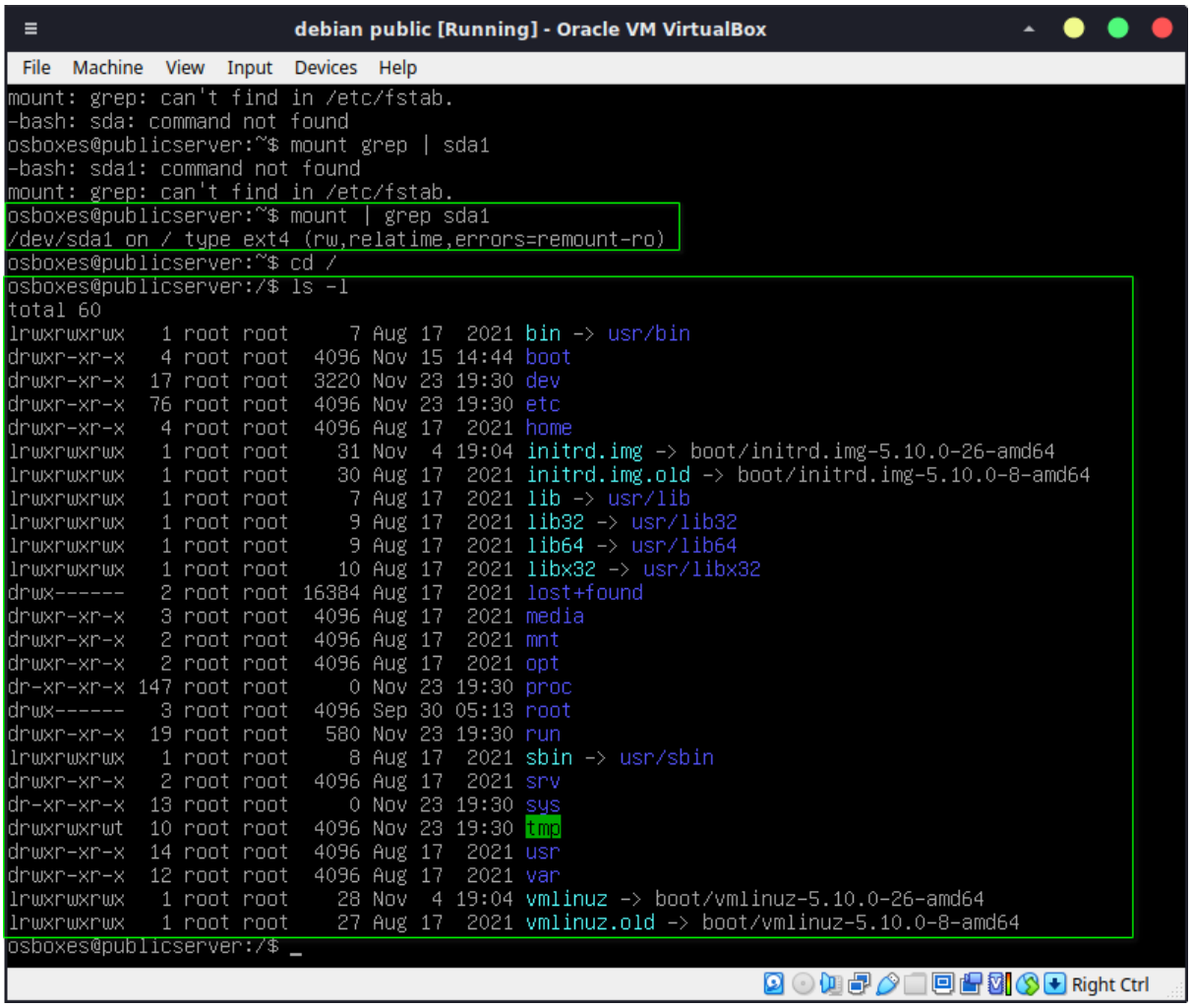
2a


debian public [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Debian GNU/Linux 11 publicserver tty1
publicserver login: osboxes
Password:
Linux publicserver 5.10.0-26-amd64 #1 SMP Debian 5.10.197-1 (2023-09-29) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov 23 18:46:16 EST 2023 on tty1
osboxes@publicserver:~$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 500G 0 disk
├─sda1 8:1 0 220.6G 0 part /
├─sda2 8:2 0 954M 0 part /boot
├─sda3 8:3 0 8.4G 0 part [SWAP]
└─sda4 8:4 0 270.1G 0 part /home
sdb 8:16 0 8G 0 disk
└─sdb1 8:17 0 8G 0 part
sr0 11:0 1 1024M 0 rom
osboxes@publicserver:~$
Right Ctrl

2b


debian public [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@publicserver:~$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 500G 0 disk
├─sda1 8:1 0 220.6G 0 part /
├─sda2 8:2 0 954M 0 part /boot
├─sda3 8:3 0 8.4G 0 part [SWAP]
└─sda4 8:4 0 270.1G 0 part /home
sdb 8:16 0 8G 0 disk
└─sdb1 8:17 0 8G 0 part
sr0 11:0 1 1024M 0 rom
osboxes@publicserver:~$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=991344k,nr_inodes=247836,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=201848k,mode=755)
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursivep
rot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
none on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=30,pgrp=1,timeout=0,minproto=5,max
proto=5,direct,pipe_ino=10601)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
/dev/sda4 on /home type ext4 (rw,relatime)
/dev/sda2 on /boot type ext4 (rw,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=201844k,nr_inodes=50461,mode=700,u
id=1000,gid=1000)
osboxes@publicserver:~$
Right Ctrl

2c y 2d



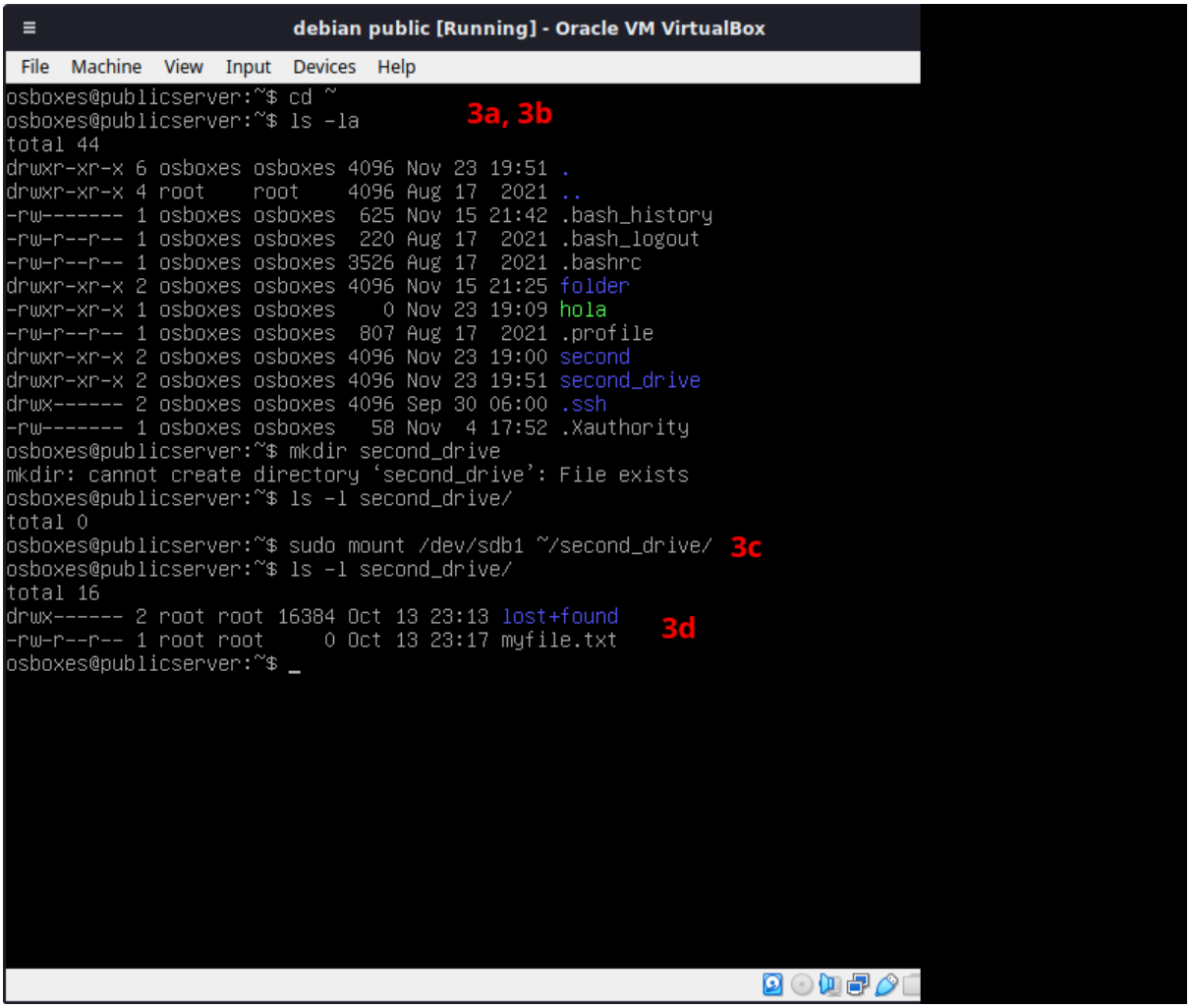
2d ¿Cuál es el significado de la salida? ¿Dónde se almacenan físicamente los archivos de la lista?

2d. ¿Por qué no se muestra /dev/sdb1 en la salida de arriba?

sdb1 no aparece porque no está montado.

Paso 3

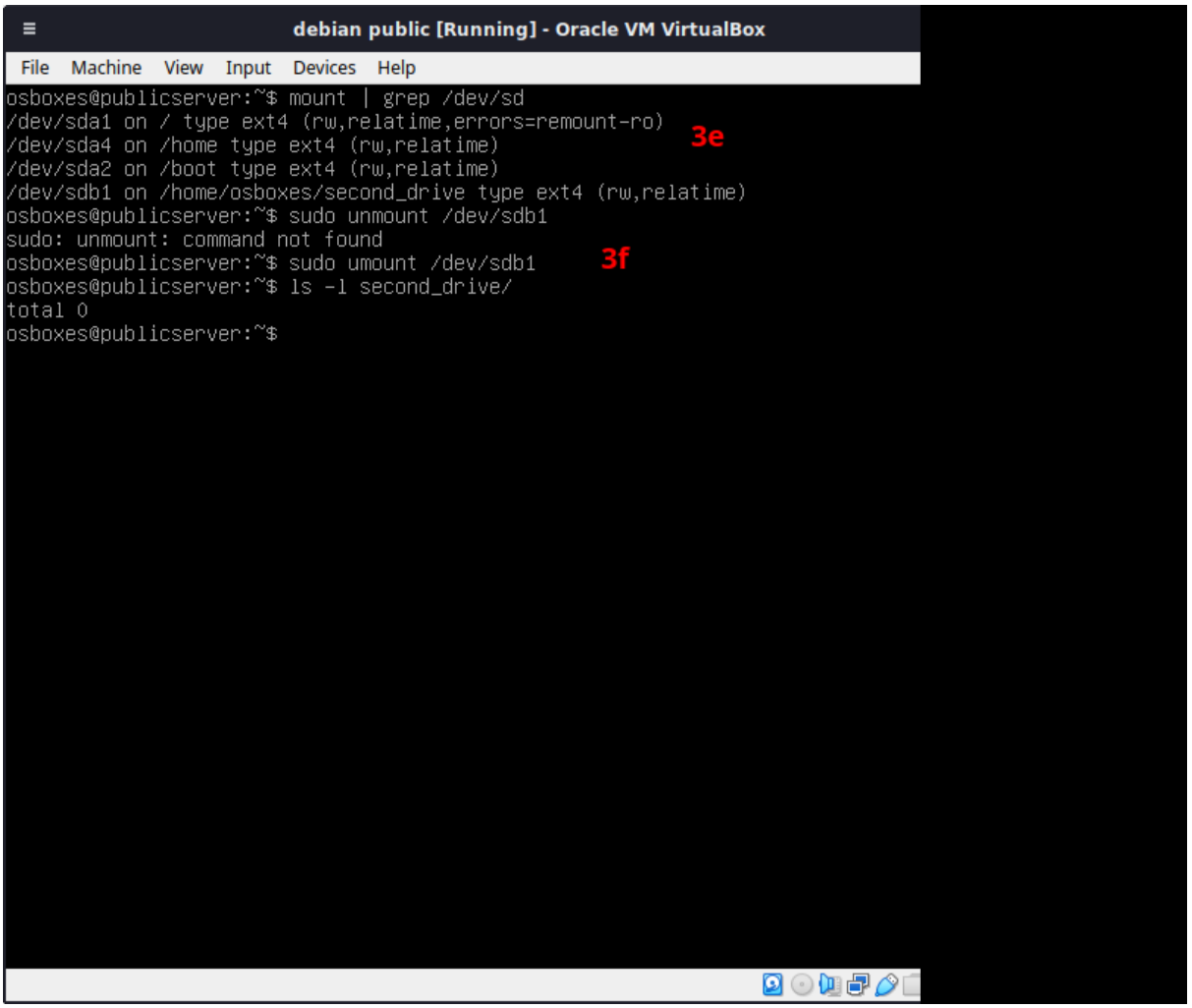
3a, 3b, 3c y 3d



3d. ¿Por qué ya no está vacío el directorio? ¿Dónde se almacenan físicamente los archivos de la lista?

Ya no está vacío porque se monto el disco /dev/sdb1 en el directorio second_drive

3e y 3f



3g. Investigar: Es posible asignar permisos específicos al montar un sistema de archivos ? ¿Puede dar un ejemplo ?

Si, se puede.

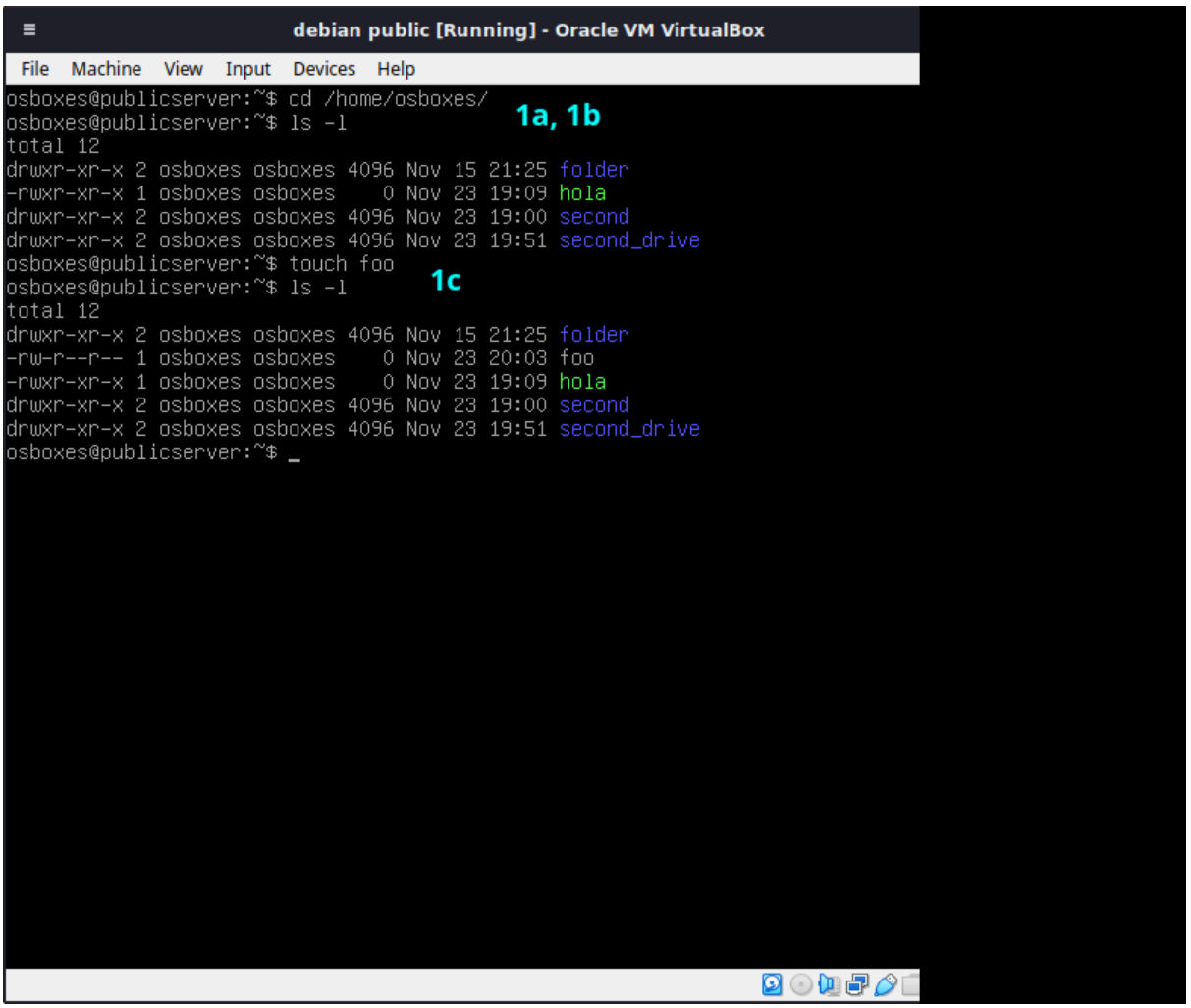
Ejemplo:

```
sudo nano /etc/fstab
# <file system> <mount point> <type> <options> <dump> <pass>
```

Parte 2: Permisos de archivo

Paso 1

1a, 1b y 1c



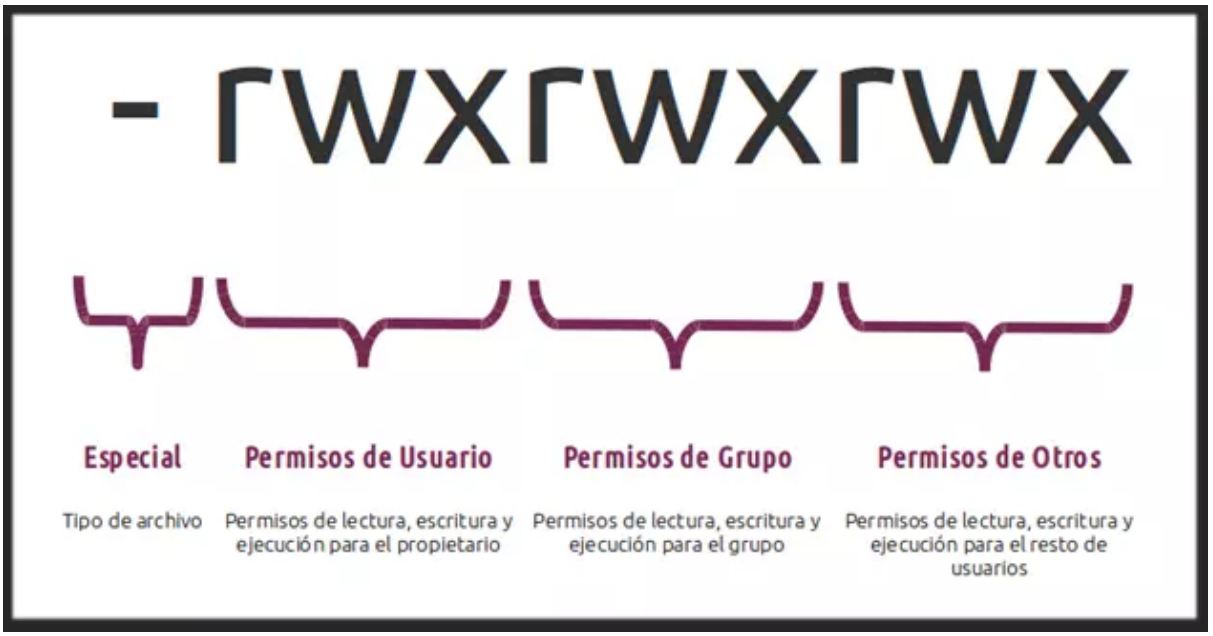
1c. Consideren el archivo foo a modo de ejemplo. ¿Quién es el propietario del archivo? ¿Y del grupo?

El owner del archivo es osboxes y el del grupo es osboxes

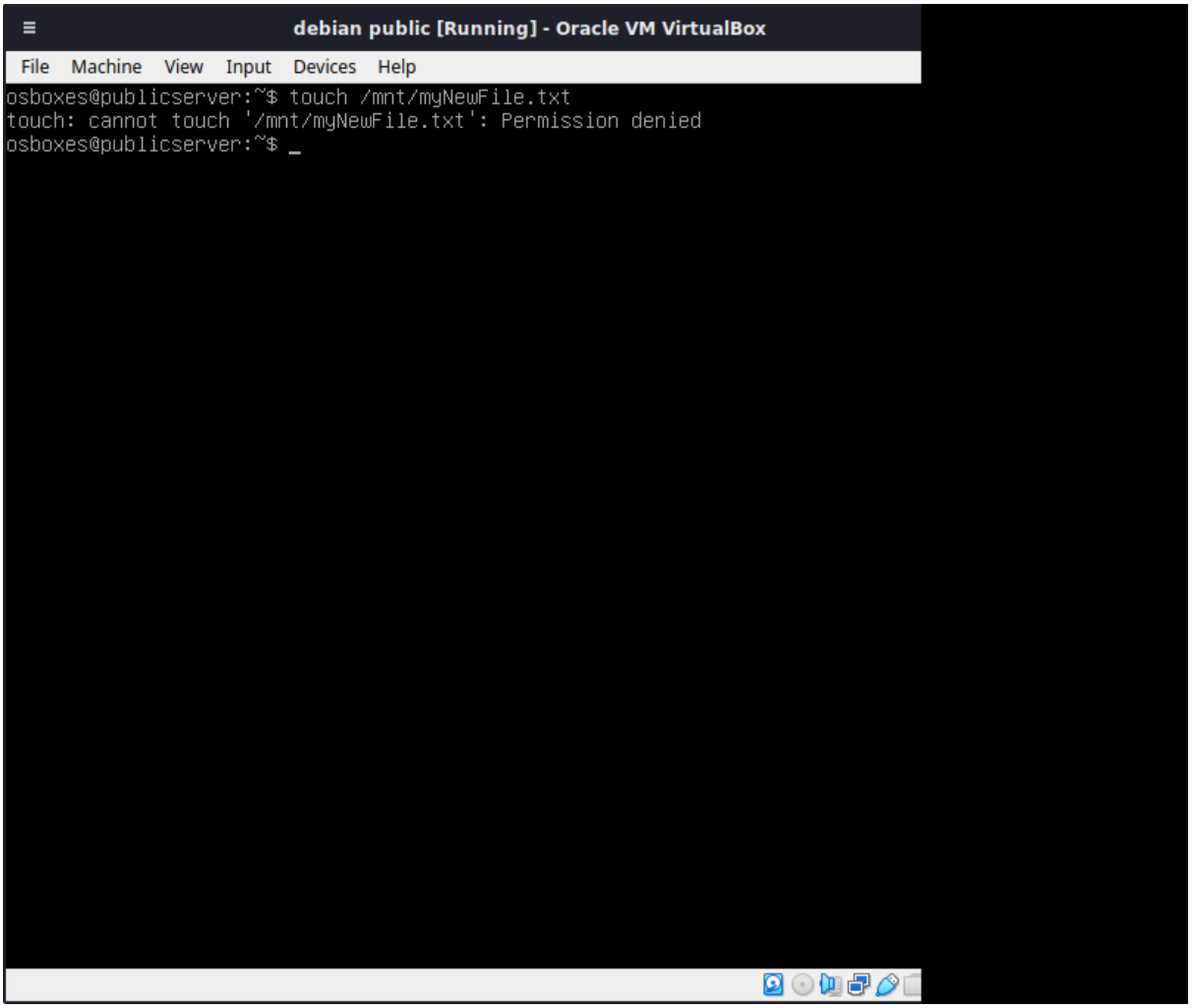
Los permisos para foo son –rw-r--r--. ¿Qué significa esto?

Tipo de archivo	Permisos de usuario (owner)	Permisos de grupo	Permisos de otros
-	rw-	r - -	r - -

file	r: lectura w: escritura	r: lectura	r: lectura

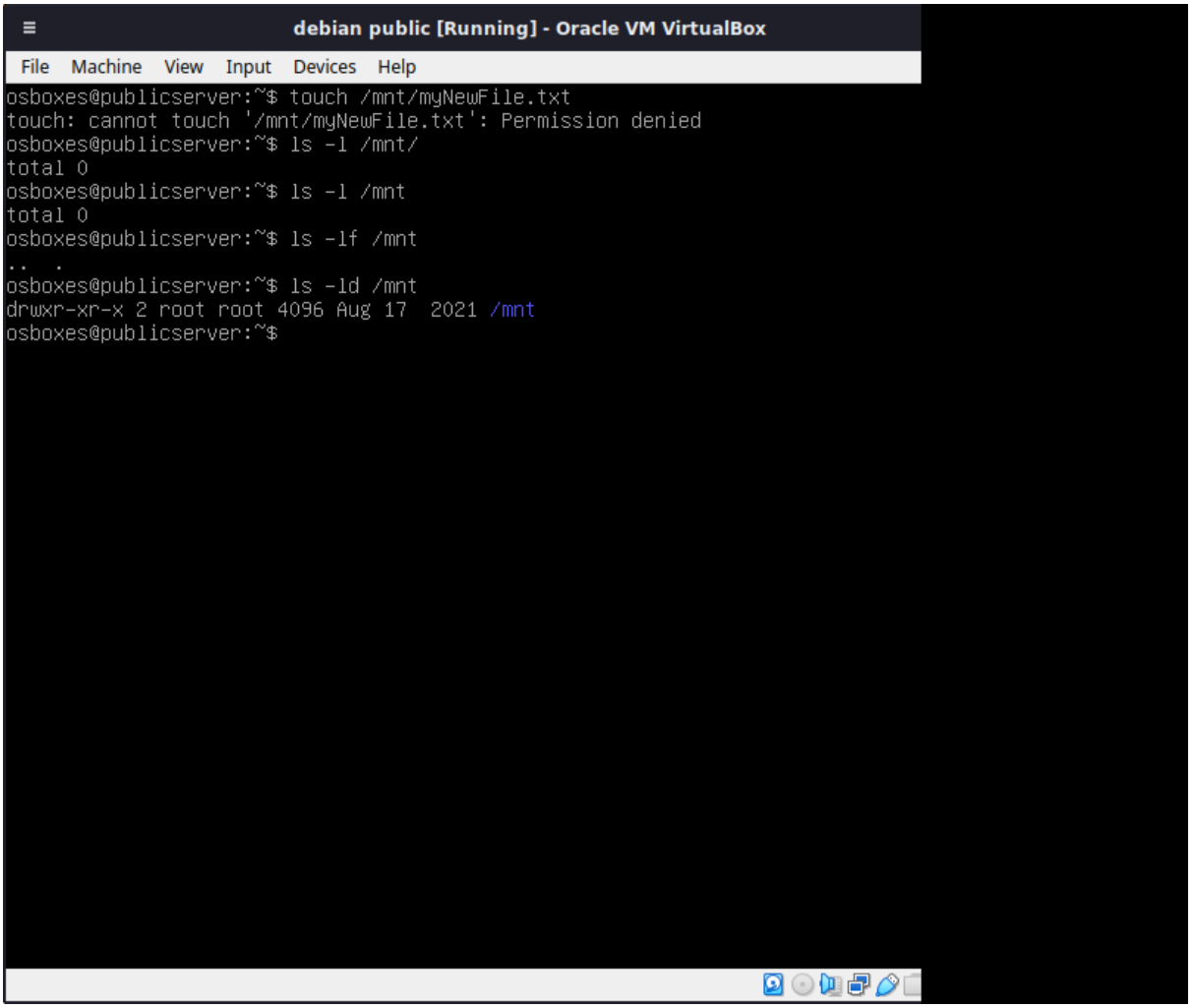


1d



1d. ¿Por qué no se creó el archivo?

Genere una lista con los permisos, la propiedad y el contenido del directorio /mnt y explique qué sucedió. Cuando se agrega la opción -d, muestra el permiso del directorio principal. Registren las respuestas en las siguientes líneas.

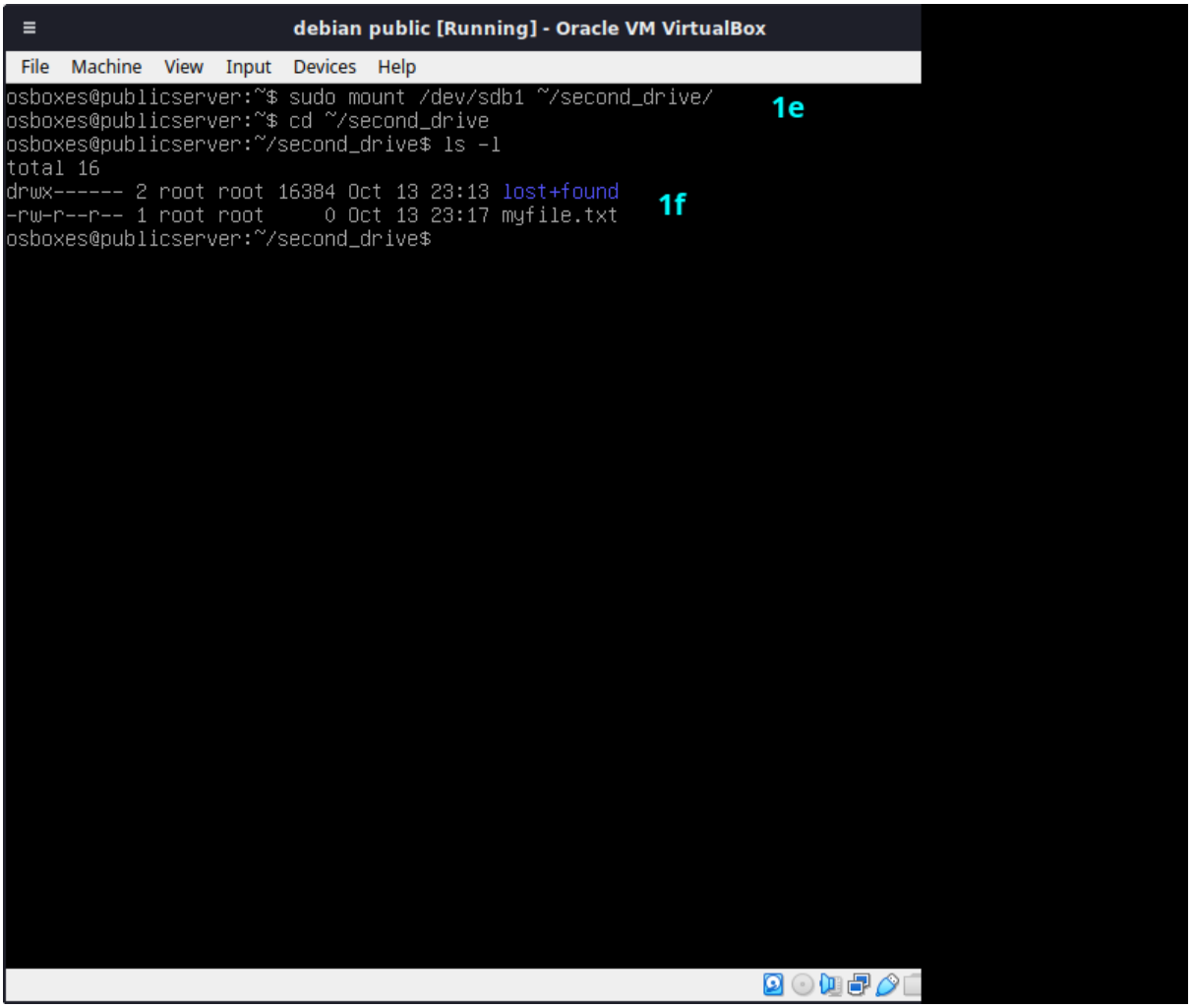


[osboxes@publicserver ~]$ ls -ld /mnt
drwxr-xr-x 2 root root 4096 Jan 5 2018 /mnt

¿Qué se puede hacer para que el comando touch que se muestra arriba tenga éxito?

Utilizar **sudo**.
Cambiar el owner.
Agregar permisos al archivo para que pueda ejecutar.

1e y 1f

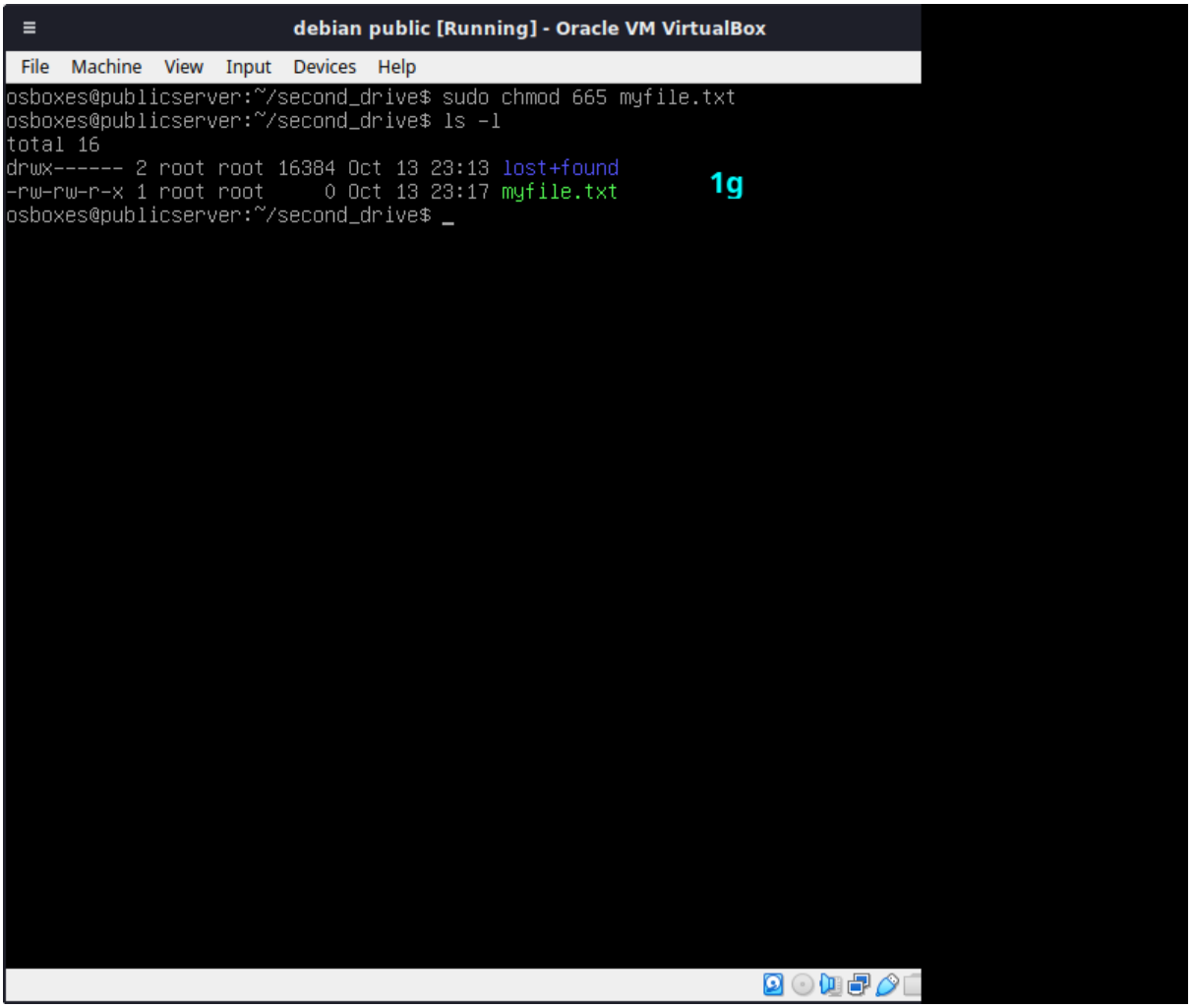


¿Cuáles son los permisos del archivo myFile.txt? Aquí.

```
[osboxes@publicserver ~]$ cd ~/second_drive
[osboxes@publicserver second_drive]$ ls -l
total 20
drwx------ 2 root root 16384 Mar 3 10:59 lost+found
-rw-r--r-- 1 root root 183 Mar 3 15:42 myFile.txt
```

Tipo de archivo	Permisos de usuario (owner)	Permisos de grupo	Permisos de otros
-	rw-	r - -	r - -
file	r: lectura w: escritura	r: lectura	r: lectura

1g



1g ¿Cambiaron los permisos? ¿Cuáles son los permisos de myFile.txt?

Si, cambiaron los permisos del archivo myfile.txt. Los permisos son:

Tipo de archivo	Permisos de usuario (owner)	Permisos de grupo	Permisos de otros
-	rw-	r - -	r - -
file	r: lectura w: escritura	r: lectura	r: lectura

El comando **chmod** toma permisos en formato octal. De esa manera, el desglose del 665 es el siguiente:

6 en octal es 110 en binario. Suponiendo que cada posición de los permisos de un archivo puede ser 1 o 0, 110 significa rw- (leer=1, escribir=1 y ejecutar=0).

Por lo tanto, el comando **chmod 665 myFile.txt** cambia los permisos a:

Owner: rw- (6 in octal or 110 in binary)

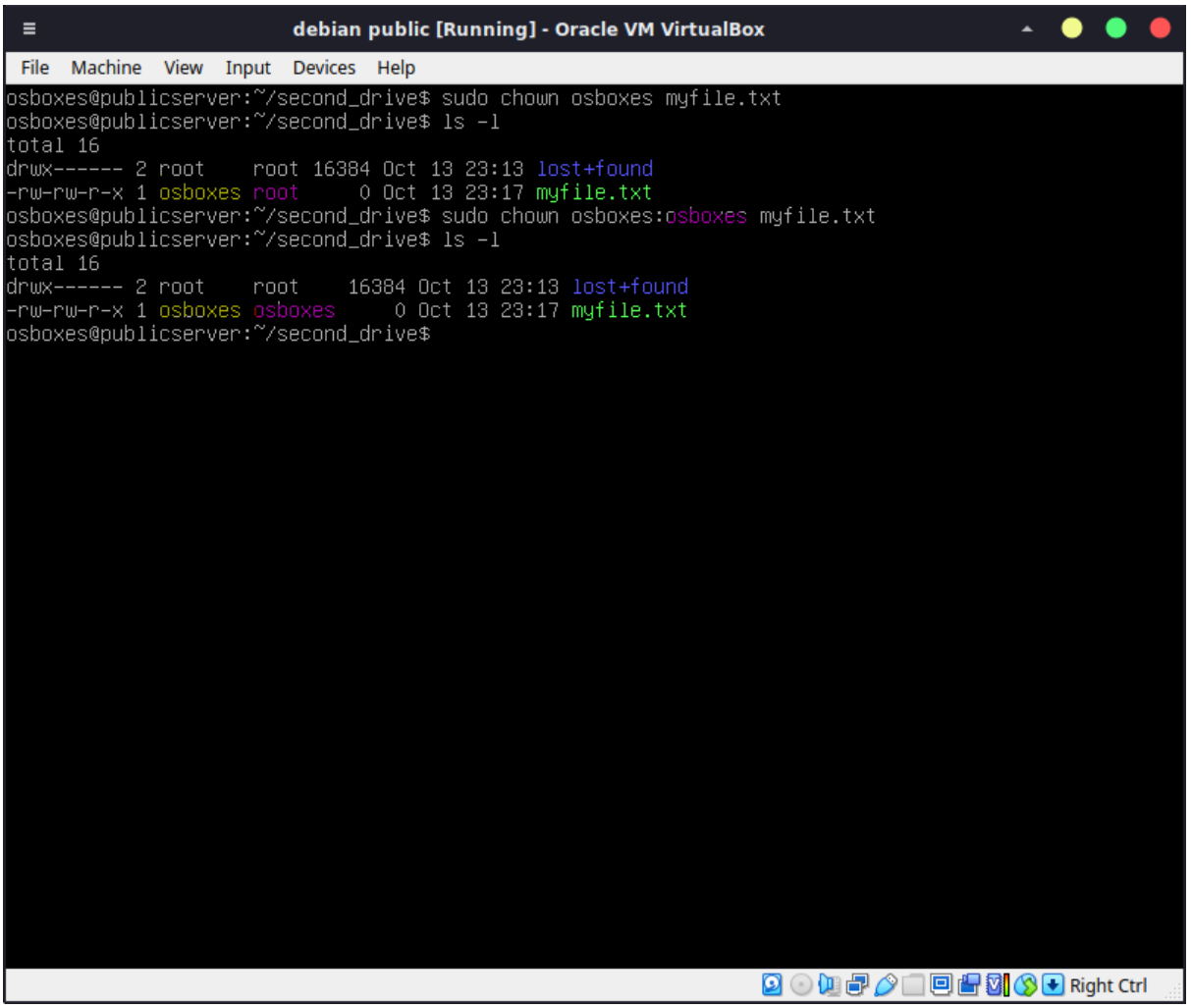
Group: rw- (6 in octal or 110 in binary)

Other: r-x (5 in octal or 101 in binary)

¿Qué comando cambiaría los permisos de myFile.txt a rwxrwxrwx, con lo que se otorgaría acceso total al archivo a cualquier usuario del sistema?

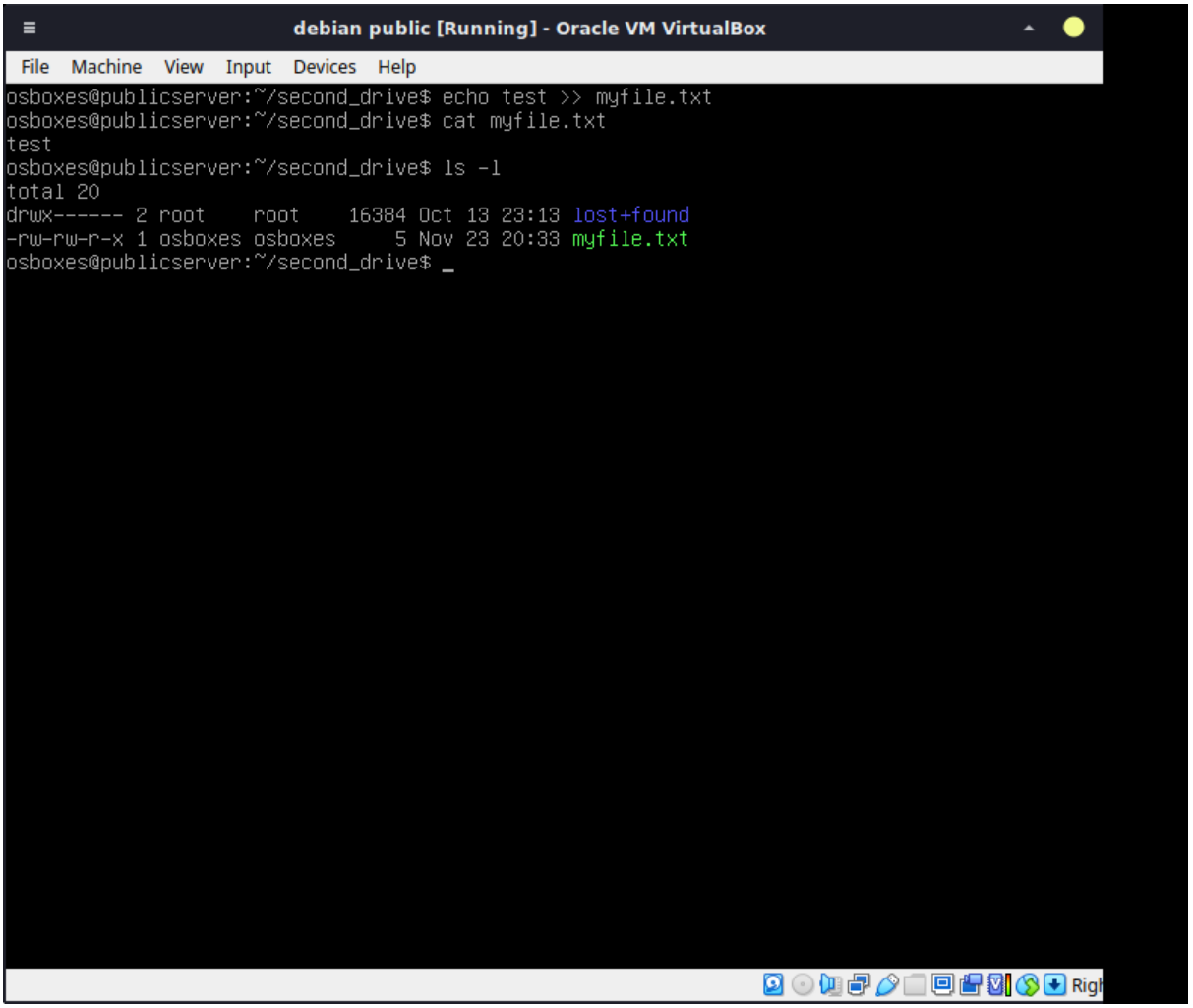
Se necesitaría el comando: **chmod 777 myFile.txt**

1h

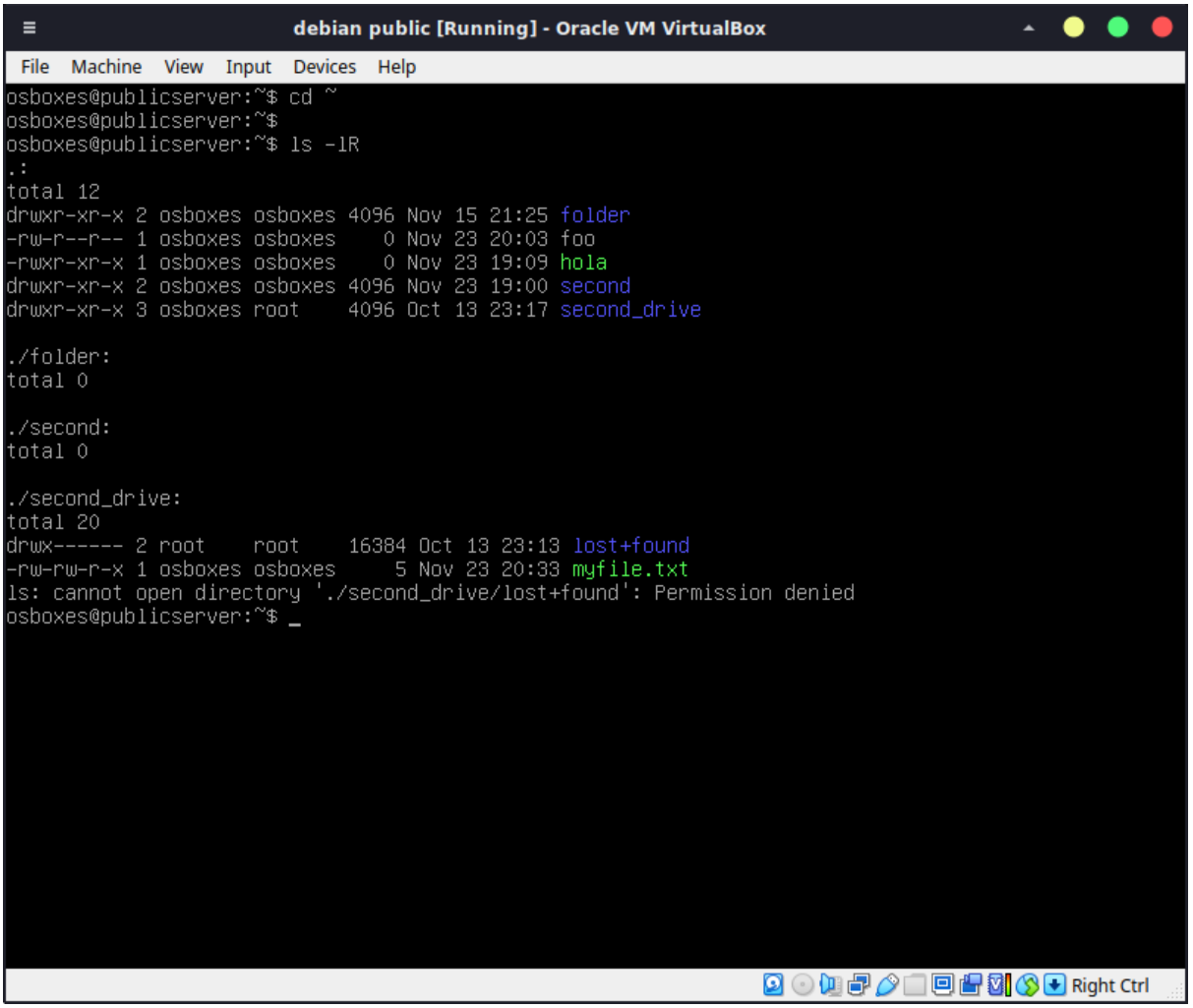


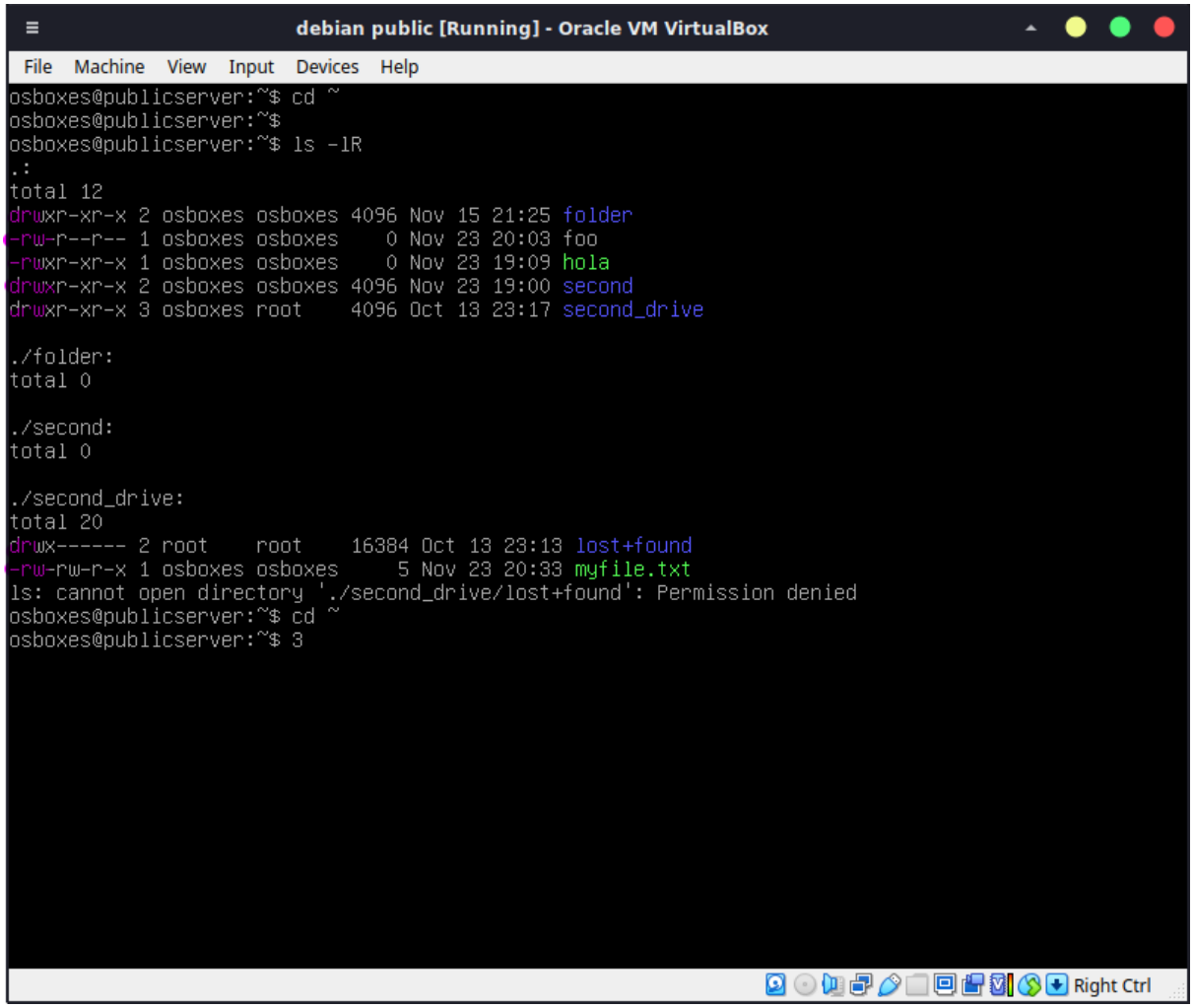
1i ¿Fue exitosa la operación? Explique.

Si, fue exitosa porque


debian public [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@publicserver:~/second_drive$ echo test >> myfile.txt
osboxes@publicserver:~/second_drive$ cat myfile.txt
test
osboxes@publicserver:~/second_drive$ ls -l
total 20
drwx------ 2 root root 16384 Oct 13 23:13 lost+found
-rw-rw-r-x 1 osboxes osboxes 5 Nov 23 20:33 myfile.txt
osboxes@publicserver:~/second_drive$

Paso 2


debian public [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@publicserver:~$ cd ~
osboxes@publicserver:~$
osboxes@publicserver:~$ ls -lR
.:
total 12
drwxr-xr-x 2 osboxes osboxes 4096 Nov 15 21:25 folder
-rw-r--r-- 1 osboxes osboxes   0 Nov 23 20:03 foo
-rwxr-xr-x 1 osboxes osboxes   0 Nov 23 19:09 hola
drwxr-xr-x 2 osboxes osboxes 4096 Nov 23 19:00 second
drwxr-xr-x 3 osboxes root   4096 Oct 13 23:17 second_drive
./folder:
total 0
./second:
total 0
./second_drive:
total 20
drwx------ 2 root   root   16384 Oct 13 23:13 lost+found
-rw-rw-r-x 1 osboxes osboxes   5 Nov 23 20:33 myfile.txt
ls: cannot open directory './second_drive/lost+found': Permission denied
osboxes@publicserver:~$
Right Ctrl



Comparen los permisos del directorio con los del archivo. ¿Cuál es la diferencia entre la parte inicial de ambos?

Los directorios siempre empiezan con **d**, mientras que los archivos no.