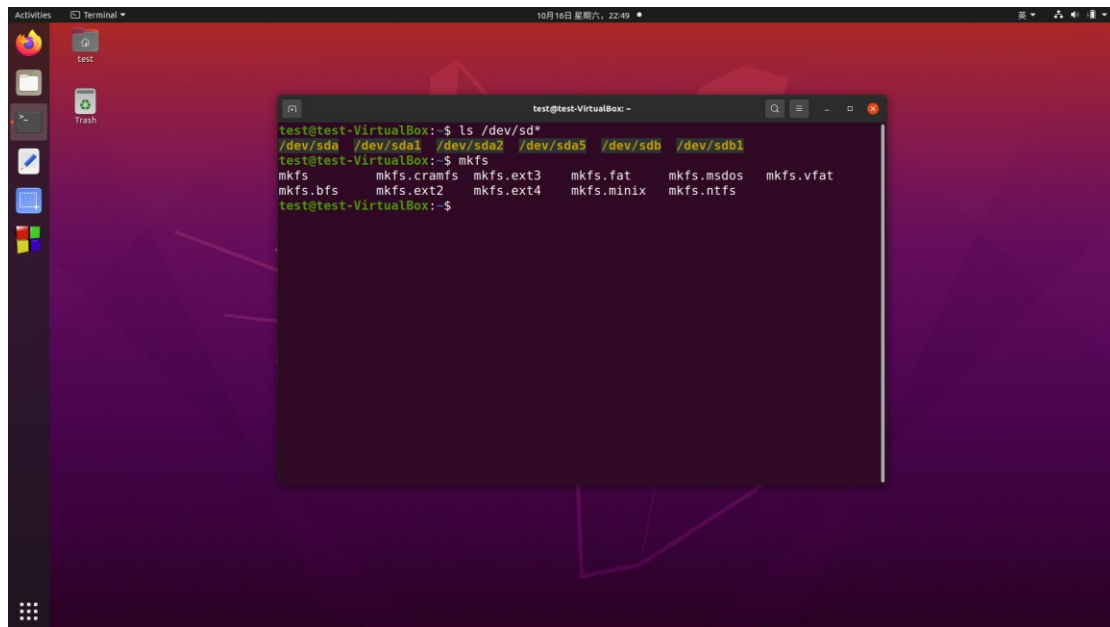
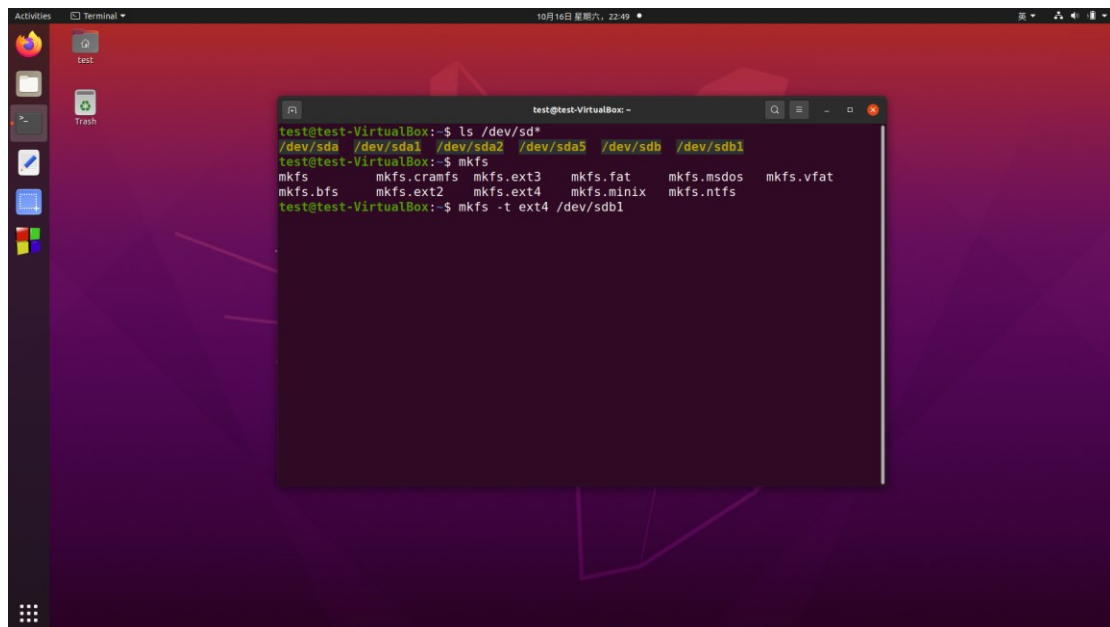


# Add a New Disk and Resize it In Linux

Format the new partition: make a new filesystem

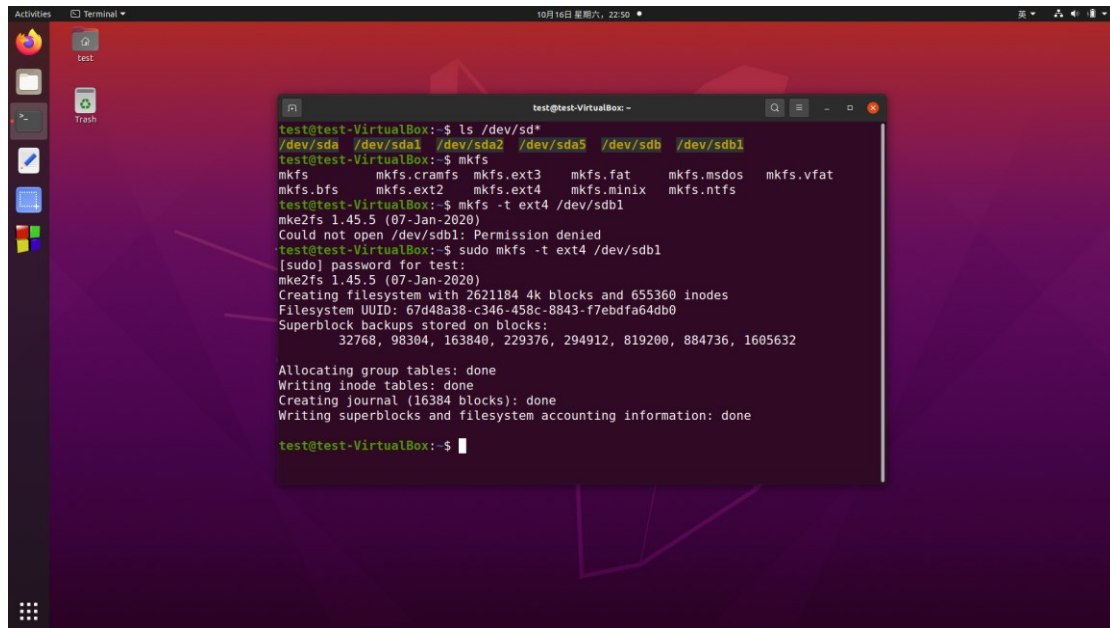
A terminal window titled 'test@test-VirtualBox: ~' is open on a Linux desktop. The user has run 'ls /dev/sd\*' and the output is: '/dev/sda /dev/sda1 /dev/sda2 /dev/sda5 /dev/sdb /dev/sdb1'. Then, the user has run 'mkfs' and the output shows a list of available filesystems: 'mkfs', 'mkfs.cramfs', 'mkfs.ext3', 'mkfs.fat', 'mkfs.msdos', 'mkfs.vfat', 'mkfs.bfs', 'mkfs.ext2', 'mkfs.ext4', 'mkfs.minix', and 'mkfs.ntfs'.

```
test@test-VirtualBox:~$ ls /dev/sd*
/dev/sda /dev/sda1 /dev/sda2 /dev/sda5 /dev/sdb /dev/sdb1
test@test-VirtualBox:~$ mkfs
mkfs      mkfs.cramfs  mkfs.ext3   mkfs.fat   mkfs.msdos  mkfs.vfat
mkfs.bfs  mkfs.ext2    mkfs.ext4   mkfs.minix mkfs.ntfs
```

The same terminal window is shown, but now the user has entered the command 'mkfs -t ext4 /dev/sdb1' to format the new partition.

```
test@test-VirtualBox:~$ ls /dev/sd*
/dev/sda /dev/sda1 /dev/sda2 /dev/sda5 /dev/sdb /dev/sdb1
test@test-VirtualBox:~$ mkfs
mkfs      mkfs.cramfs  mkfs.ext3   mkfs.fat   mkfs.msdos  mkfs.vfat
mkfs.bfs  mkfs.ext2    mkfs.ext4   mkfs.minix mkfs.ntfs
test@test-VirtualBox:~$ mkfs -t ext4 /dev/sdb1
```

`$sudo mkfs -t ext4 /dev/sdb1`

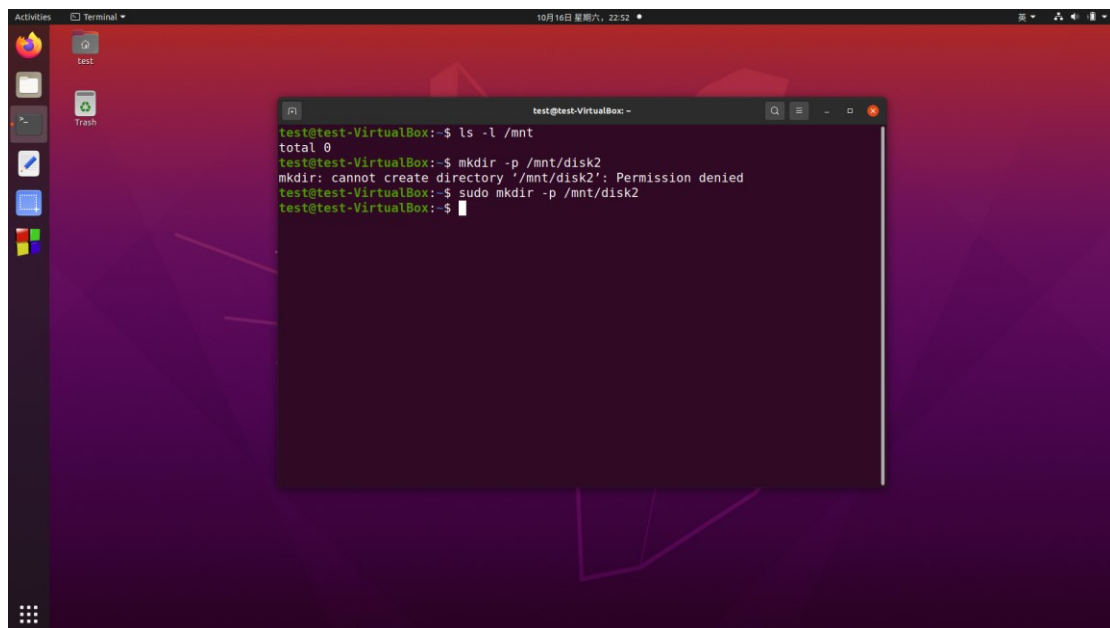
A terminal window titled 'test@test-VirtualBox: ~' is open on a Linux desktop. The user has run 'ls /dev/sd\*' showing various disk devices. Then they run 'mkfs' which lists available filesystems. They then run 'mkfs -t ext4 /dev/sdb1' but get a 'Permission denied' error. Finally, they run 'sudo mkfs -t ext4 /dev/sdb1' and successfully format the disk. The output shows the creation of a filesystem with 2621184 4k blocks and 655360 inodes, with a UUID of 67d48a38-c346-458c-8843-f7ebdfa64db0. The process concludes with 'Writing superblocks and filesystem accounting information: done'.

```
test@test-VirtualBox:~$ ls /dev/sd*
/dev/sda /dev/sda1 /dev/sda2 /dev/sda5 /dev/sdb /dev/sdb1
test@test-VirtualBox:~$ mkfs
mkfs      mkfs.cramfs  mkfs.ext3    mkfs.fat    mkfs.msdos  mkfs.vfat
mkfs.bfs  mkfs.ext2    mkfs.ext4    mkfs.minix  mkfs.ntfs
test@test-VirtualBox:~$ mkfs -t ext4 /dev/sdb1
mke2fs 1.45.5 (07-Jan-2020)
Could not open /dev/sdb1: Permission denied
test@test-VirtualBox:~$ sudo mkfs -t ext4 /dev/sdb1
[sudo] password for test:
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 2621184 4k blocks and 655360 inodes
Filesystem UUID: 67d48a38-c346-458c-8843-f7ebdfa64db0
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

test@test-VirtualBox:~$
```

mount the new disk to specified directory

A terminal window titled 'test@test-VirtualBox: ~' is open. The user runs 'ls -l /mnt' showing an empty directory. Then they run 'mkdir -p /mnt/disk2' but get a 'Permission denied' error. Finally, they run 'sudo mkdir -p /mnt/disk2' and successfully create the directory.

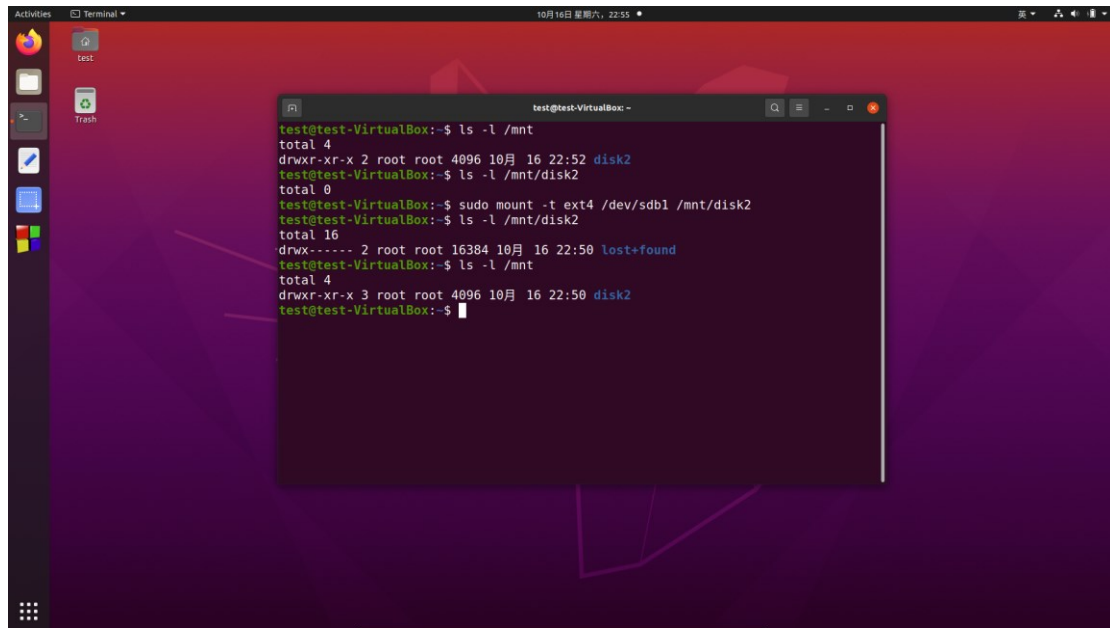
```
test@test-VirtualBox:~$ ls -l /mnt
total 0
test@test-VirtualBox:~$ mkdir -p /mnt/disk2
mkdir: cannot create directory '/mnt/disk2': Permission denied
test@test-VirtualBox:~$ sudo mkdir -p /mnt/disk2
test@test-VirtualBox:~$
```

sudo mkdir -p /mnt/disk2

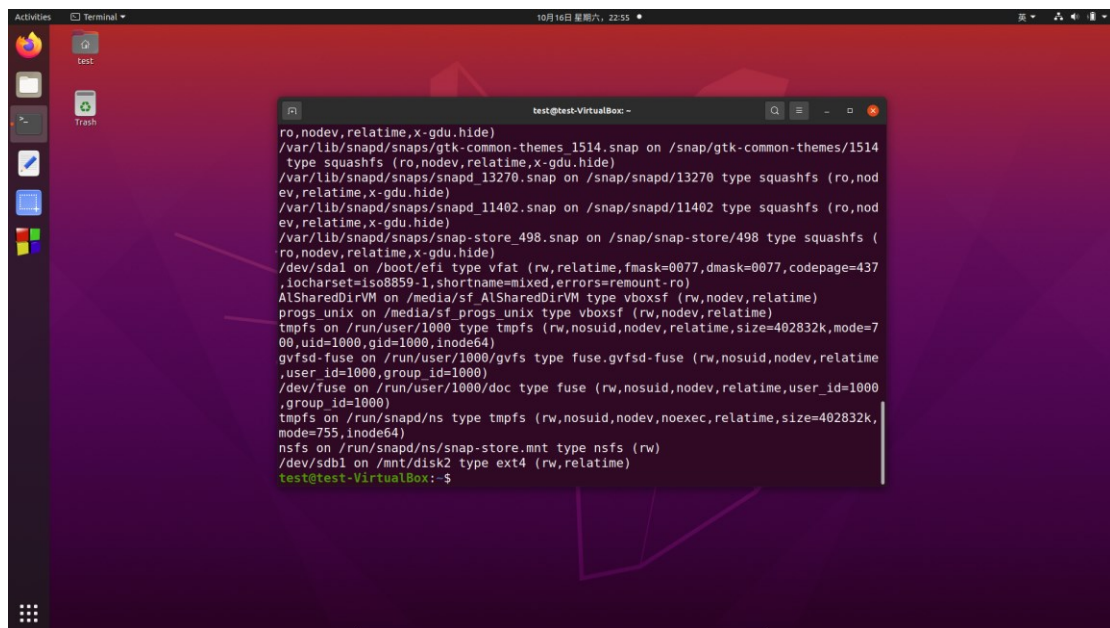
ls -l /mnt

ls -l /mnt/disk2

sudo mount -t ext4 /dev/sdb1 /mnt/disk2



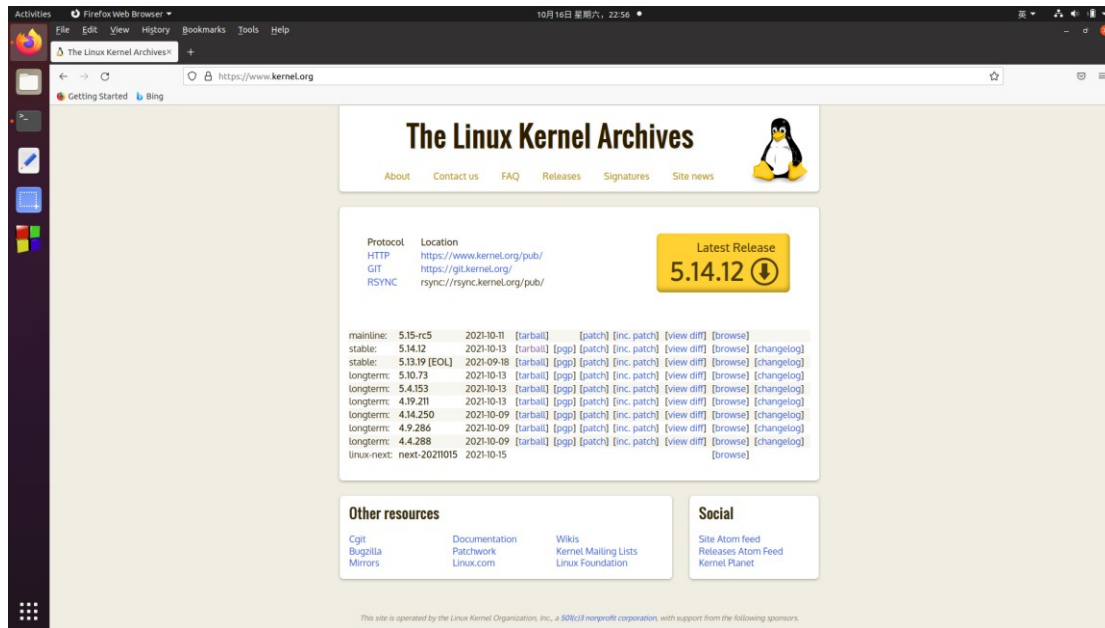
```
test@test-VirtualBox:~$ ls -l /mnt
total 4
drwxr-xr-x 2 root root 4096 10月 16 22:52 disk2
test@test-VirtualBox:~$ ls -l /mnt/disk2
total 0
test@test-VirtualBox:~$ sudo mount -t ext4 /dev/sdb1 /mnt/disk2
test@test-VirtualBox:~$ ls -l /mnt/disk2
total 16
drwx----- 2 root root 16384 10月 16 22:50 lost+found
test@test-VirtualBox:~$ ls -l /mnt
total 4
drwxr-xr-x 3 root root 4096 10月 16 22:50 disk2
test@test-VirtualBox:~$
```



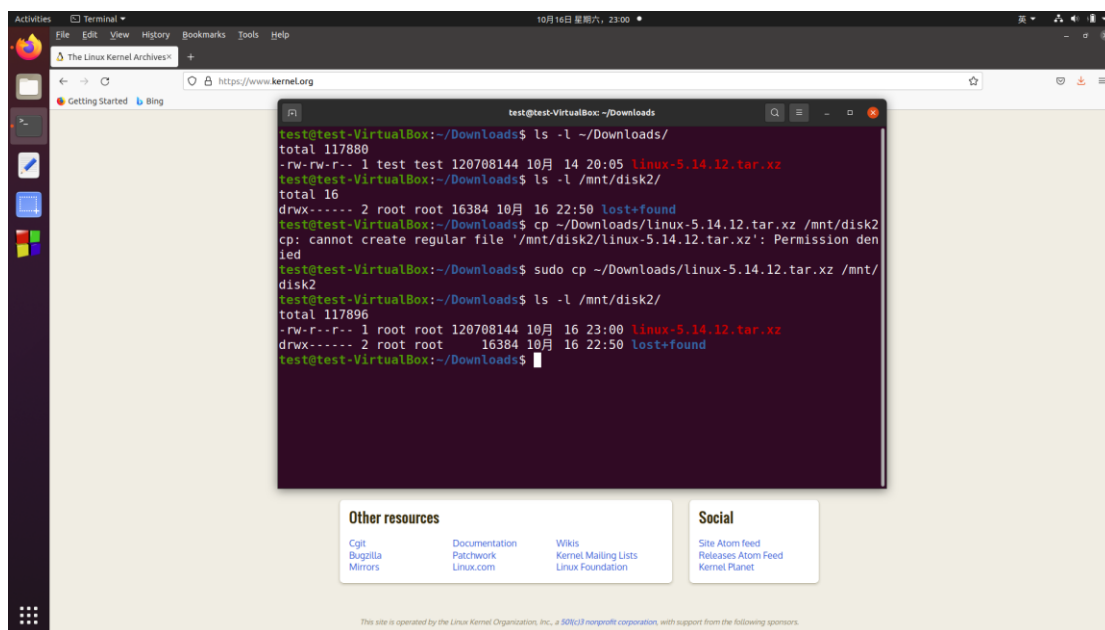
```
ro,nodev,relatime,x-gdu.hide)
/var/lib/snapd/snaps/gtk-common-themes_1514.snap on /snap/gtk-common-themes/1514
type squashfs (ro,nodev,relatime,x-gdu.hide)
/var/lib/snapd/snaps/snapd_13270.snap on /snap/snapd/13270 type squashfs (ro,nodev,relatime,x-gdu.hide)
/var/lib/snapd/snaps/snapd_11402.snap on /snap/snapd/11402 type squashfs (ro,nodev,relatime,x-gdu.hide)
/var/lib/snapd/snaps/snap-store_498.snap on /snap/snap-store/498 type squashfs (ro,nodev,relatime,x-gdu.hide)
/dev/sdal on /boot/efi type vfat (rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=iso8859-1,shortname=mixed,errors=remount-ro)
ALSharedDirVM on /media/sf_ALSharedDirVM type vboxsf (rw,nodev,relatime)
progs_unix on /media/sf_progs_unix type vboxsf (rw,nodev,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=402832k,mode=700,uid=1000,gid=1000,inode64)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
/dev/fuse on /run/user/1000/doc type fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
tmpfs on /run/snapd/ns type tmpfs (rw,nosuid,nodev,noexec,relatime,size=402832k,mode=755,inode64)
nsfs on /run/snapd/ns/snap-store.mnt type nsfs (rw)
/dev/sdb1 on /mnt/disk2 type ext4 (rw,relatime)
test@test-VirtualBox:~$
```

use it

download the latest linux kernel

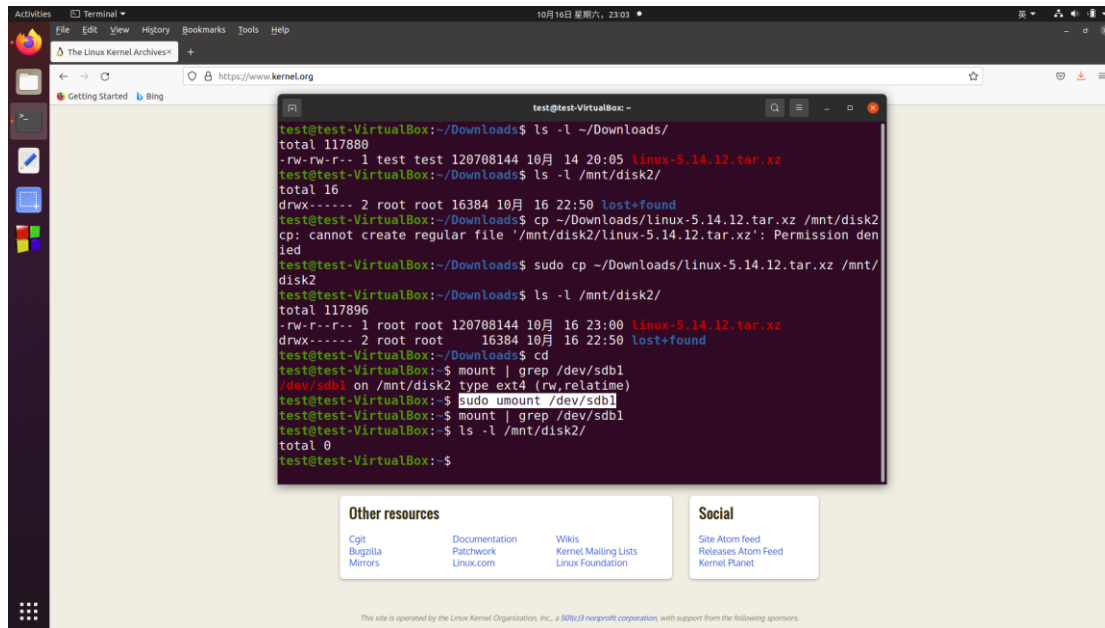


```
sudo cp ~/Downloads/linux-5.14.12.tar.xz /mnt/disk2
```



umount the new partition

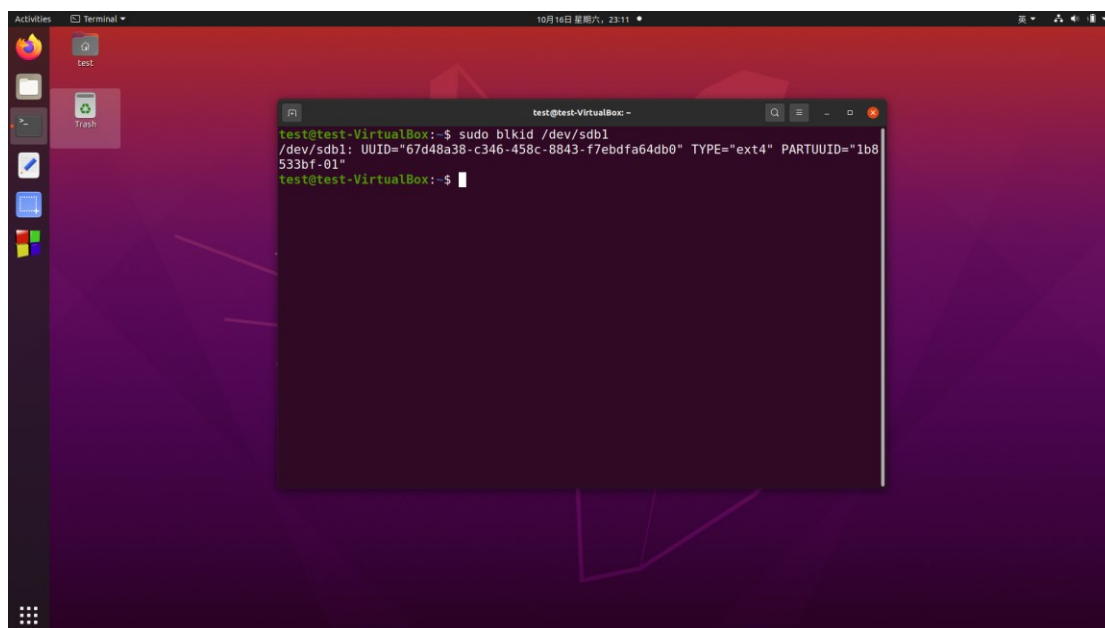
```
sudo umount /dev/sdb1
```



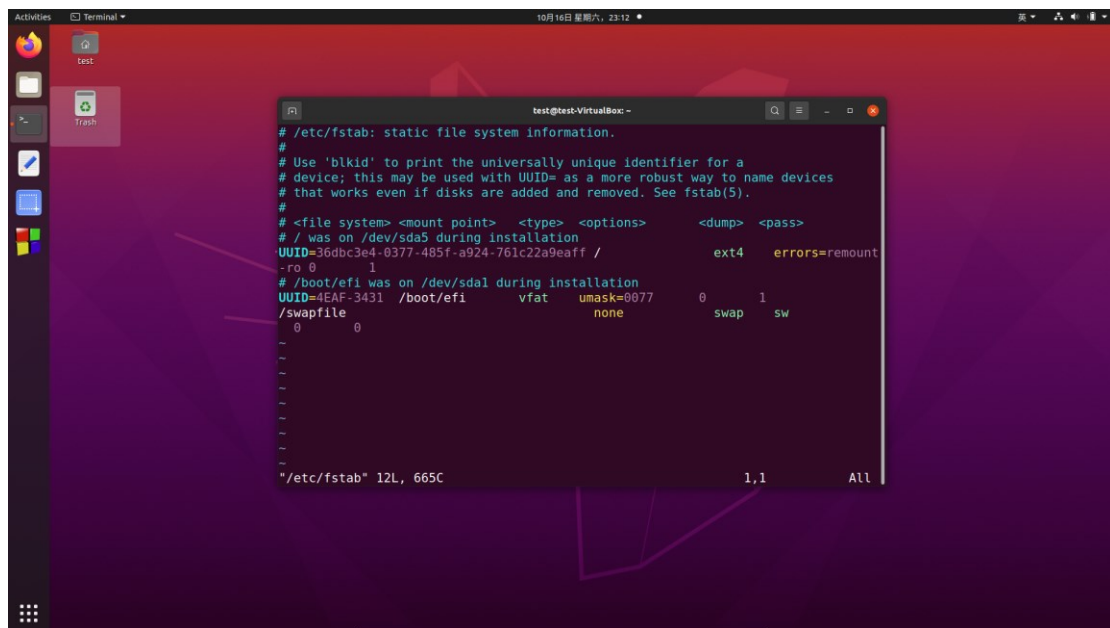
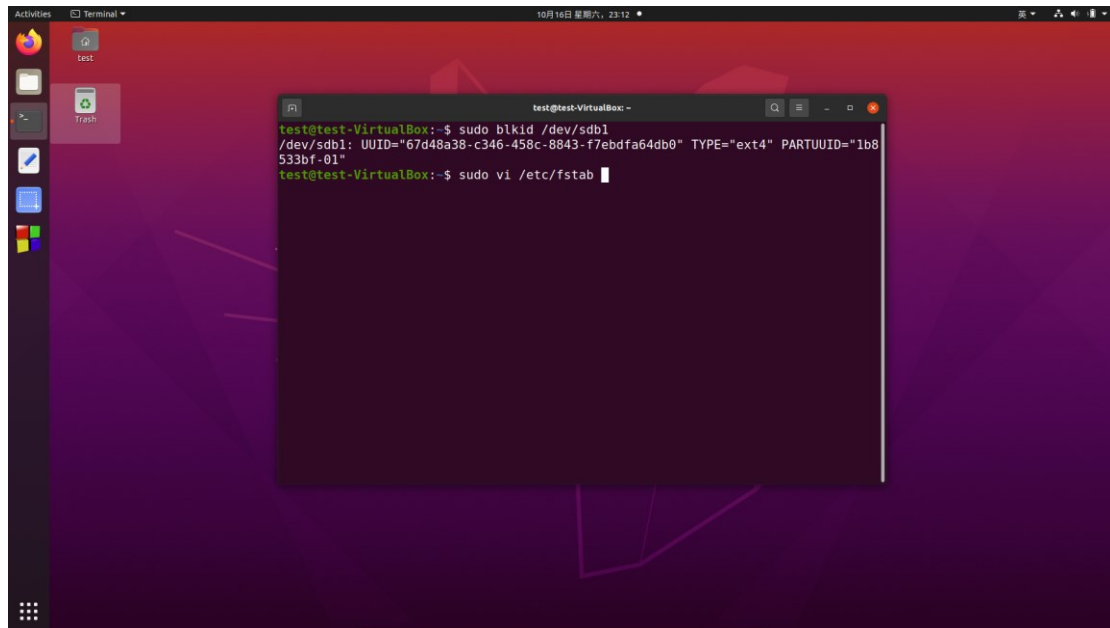
## auto mount

\$ sudo blkid /dev/sdb1

/dev/sdb1:           UUID="67d48a38-c346-458c-8843-f7ebdfa64db0"           TYPE="ext4"  
PARTUUID="1b8533bf-01"  
\$



sudo vi /etc/fstab

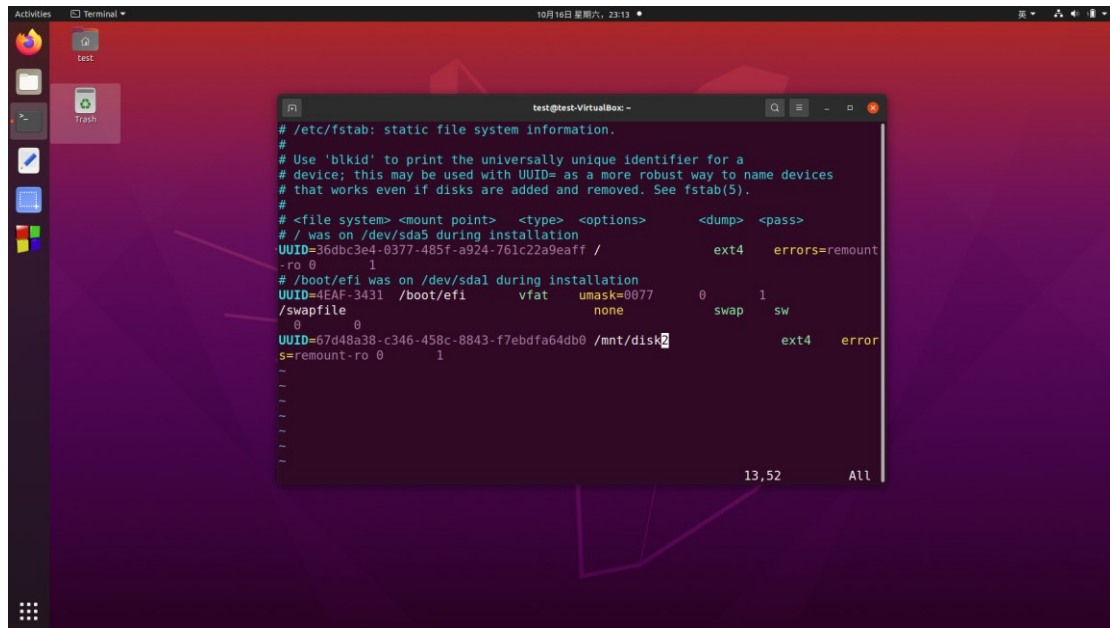


Add a new line:

UUID=67d48a38-c346-458c-8843-f7ebdfa64db0 /mnt/disk2

ext4

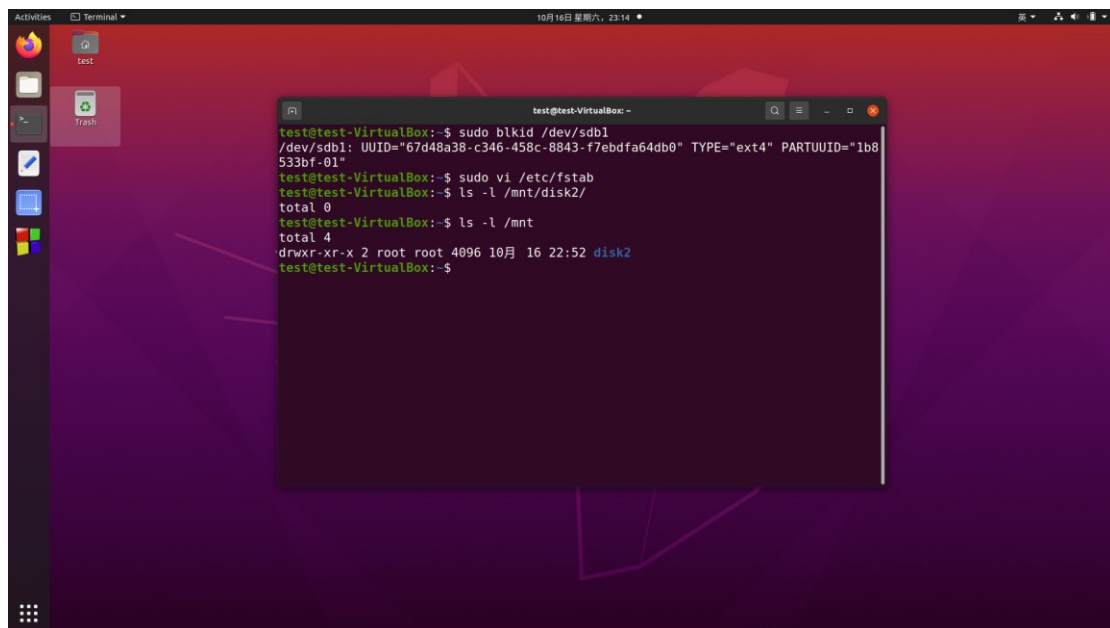
errors=remount-ro 0 1



The screenshot shows a Linux desktop with a terminal window open. The terminal displays the contents of the `/etc/fstab` file, which is a static file system information file. The file contains entries for the root filesystem, boot/efi, swapfile, and a disk mounted at `/mnt/disk2`. The entries are as follows:

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda5 during installation
UUID=36dbc3e4-0377-485f-a924-761c22a9eaff / ext4 errors=remount-ro 0 1
# /boot/efi was on /dev/sda1 during installation
UUID=4EAF-3431 /boot/efi vfat umask=0077 0 1
/swapfile none swap sw 0 0
UUID=67d48a38-c346-458c-8843-f7ebdfa64db0 /mnt/disk2 ext4 error 0 1
s=remount-ro 0 1
```

Auto remount:

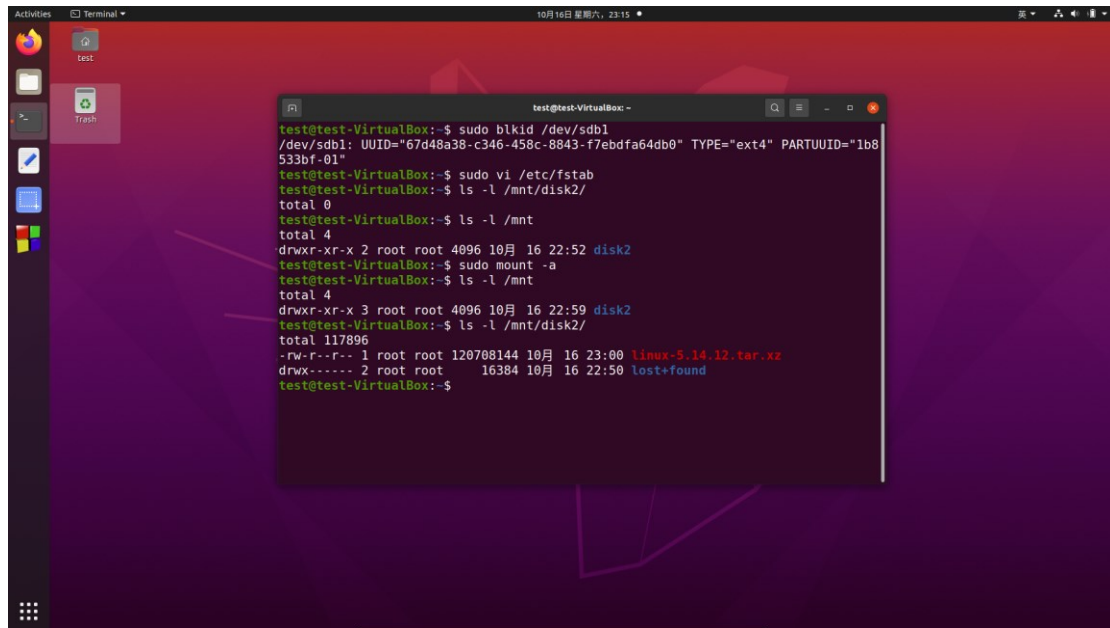


The screenshot shows a Linux desktop with a terminal window open. The terminal displays the following commands and their output:

```
test@test-VirtualBox:~$ sudo blkid /dev/sdb1
/dev/sdb1: UUID="67d48a38-c346-458c-8843-f7ebdfa64db0" TYPE="ext4" PARTUUID="1b8533bf-01"
test@test-VirtualBox:~$ sudo vi /etc/fstab
test@test-VirtualBox:~$ ls -l /mnt/disk2/
total 0
test@test-VirtualBox:~$ ls -l /mnt
total 4
drwxr-xr-x 2 root root 4096 10月 16 22:52 disk2
test@test-VirtualBox:~$
```

\$ sudo mount -a

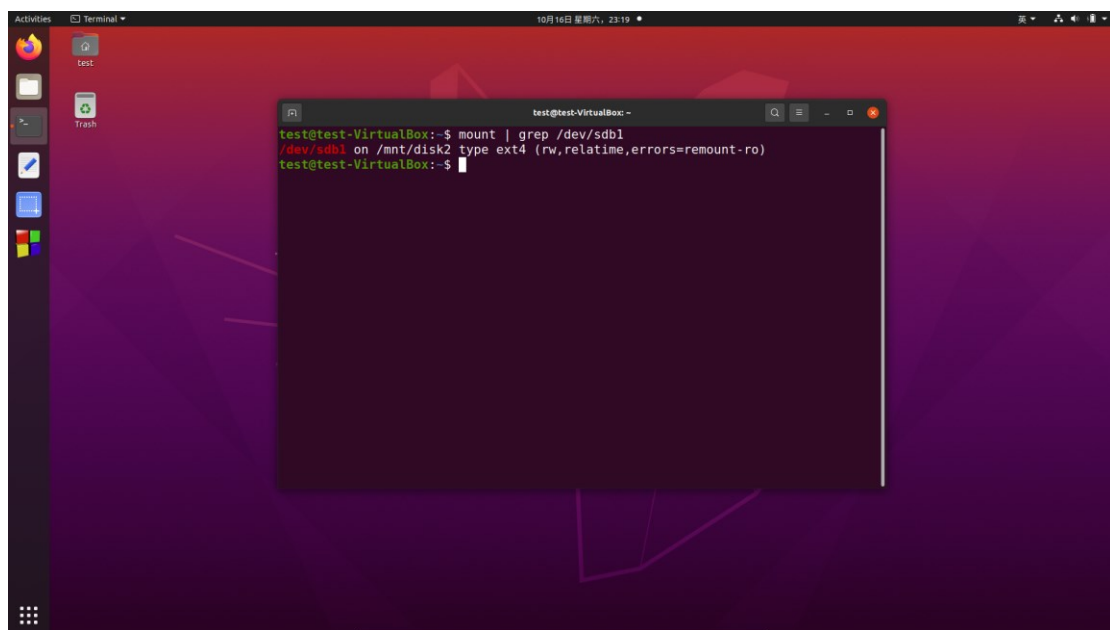




A terminal window on a Linux desktop environment. The user is performing several commands to set up a disk. The commands and their outputs are as follows:

```
test@test-VirtualBox:~$ sudo blkid /dev/sdb1
/dev/sdb1: UUID="67d48a38-c346-458c-8843-f7ebdfa64db0" TYPE="ext4" PARTUUID="1b8533bf-01"
test@test-VirtualBox:~$ sudo vi /etc/fstab
test@test-VirtualBox:~$ ls -l /mnt/disk2/
total 0
test@test-VirtualBox:~$ ls -l /mnt
total 4
drwxr-xr-x 2 root root 4096 10月 16 22:52 disk2
test@test-VirtualBox:~$ sudo mount -a
test@test-VirtualBox:~$ ls -l /mnt
total 4
drwxr-xr-x 3 root root 4096 10月 16 22:59 disk2
test@test-VirtualBox:~$ ls -l /mnt/disk2/
total 117896
-rw-r--r-- 1 root root 120708144 10月 16 23:00 linux-5.14.12.tar.xz
drwx----- 2 root root 16384 10月 16 22:50 lost+found
test@test-VirtualBox:~$
```

Reboot and check it again:  
\$ mount | grep /dev/sdb1



A terminal window showing the output of the command `mount | grep /dev/sdb1`. The output is:

```
test@test-VirtualBox:~$ mount | grep /dev/sdb1
/dev/sdb1 on /mnt/disk2 type ext4 (rw,relatime,errors=remount-ro)
test@test-VirtualBox:~$
```

End.