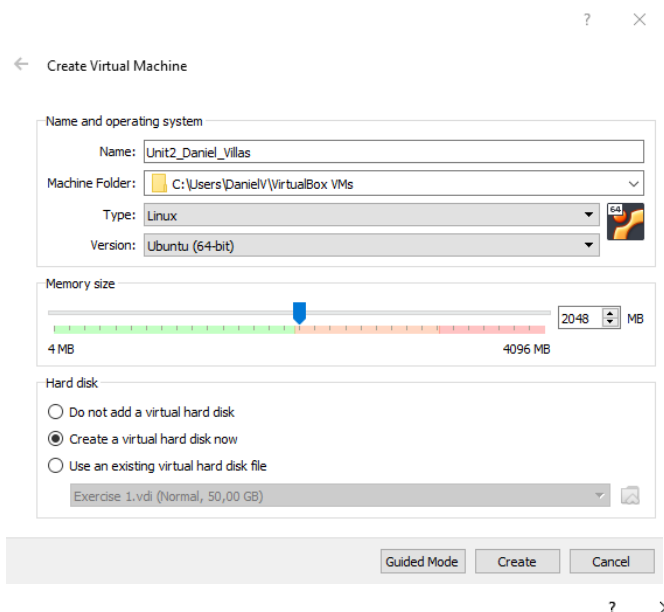


UNIT 2 ASSIGNMENT

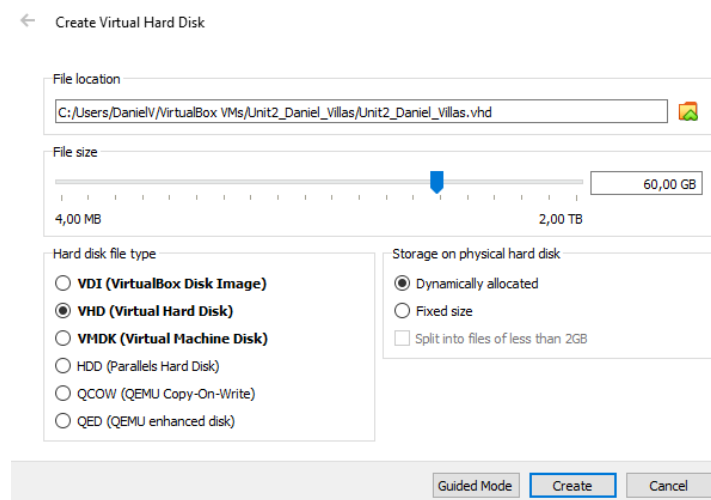
1. Install Ubuntu (I chose ubuntu 16 because of my computer's RAM) following this steps.

1. Create the virtual machine. Chose the type (Linux) and the version of the operative system (Ubuntu 16, 64-bit), the memory size (2 GB) and select "Create a virtual hard disk now". Finally, click in "Create".
2. Create Virtual Hard Disk. Select the file location (By default, it's automatically saved in a default folder), the file size (80 GB), chose the hard disk file type (Virtual Hard Disk) and the storage in the physical hard disk (dynamically allocated). Once, having finished this, there is created the virtual machine.

1

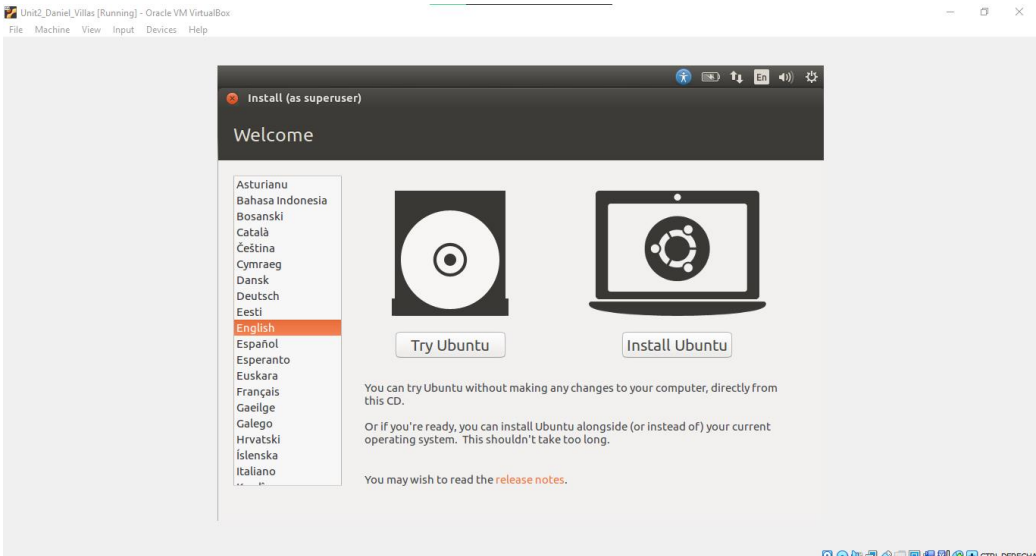


2

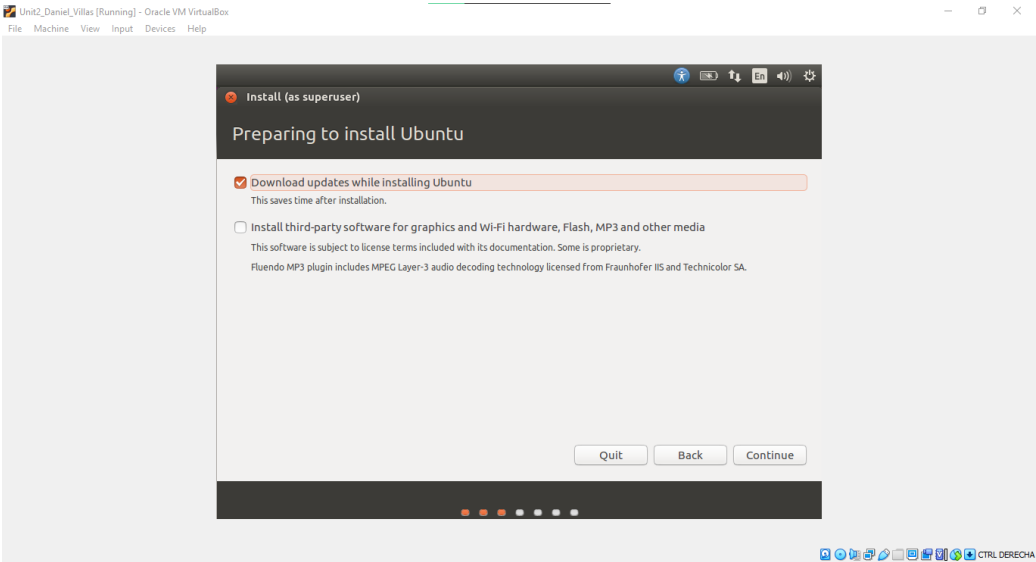


Now, start to configure ubuntu.

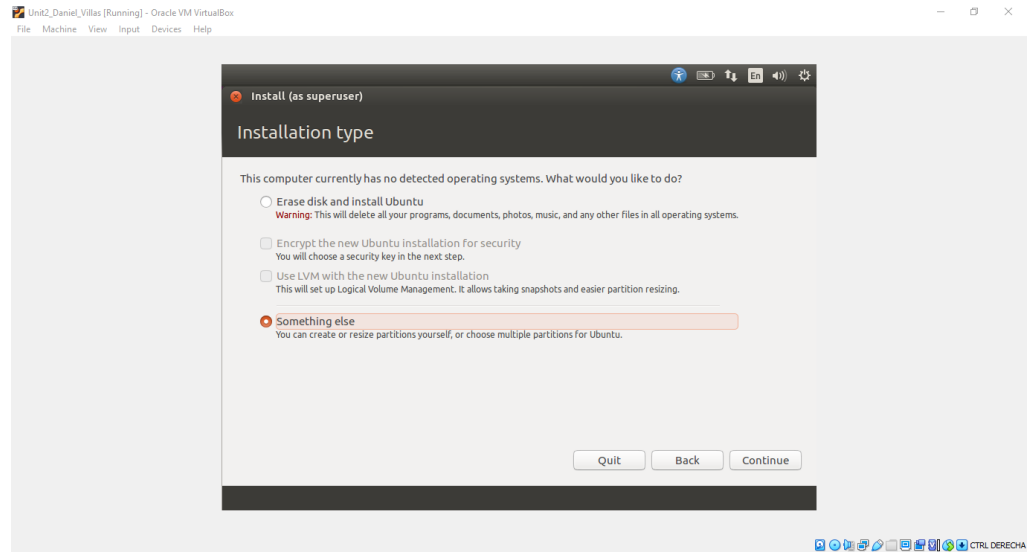
1. Click in “Install ubuntu”.



2. Select “Download updates while installing ubuntu”.

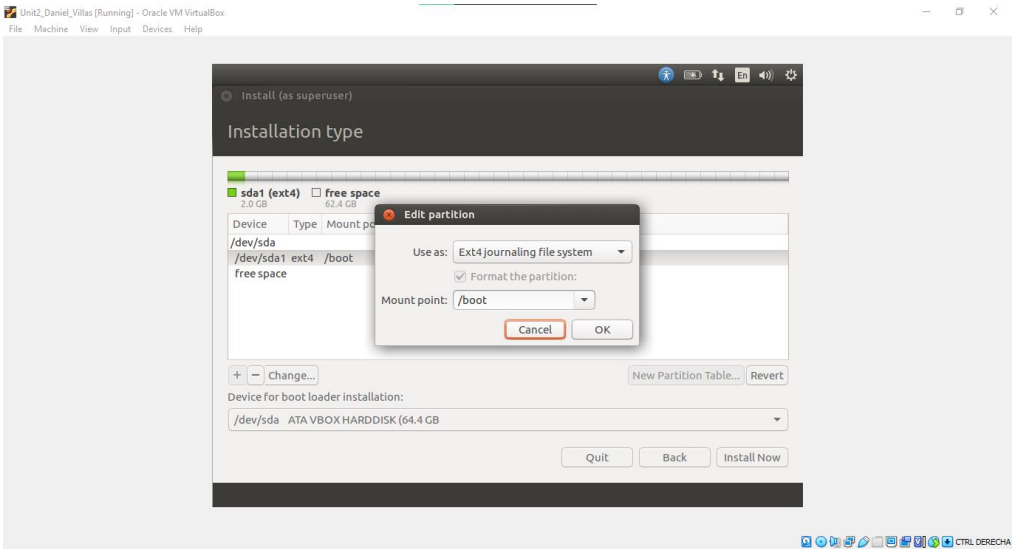


3. Chose “Something else”

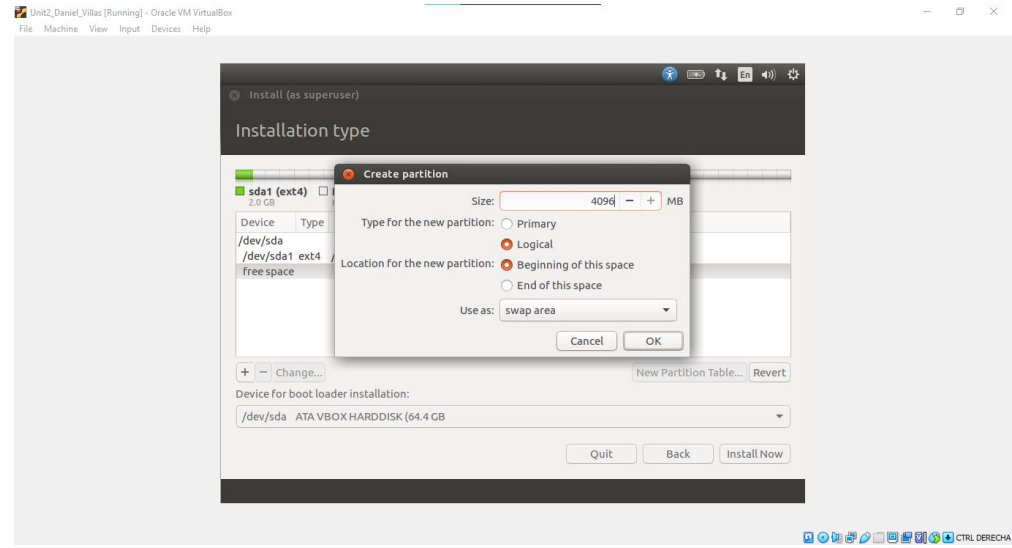


Time to create the partitions:

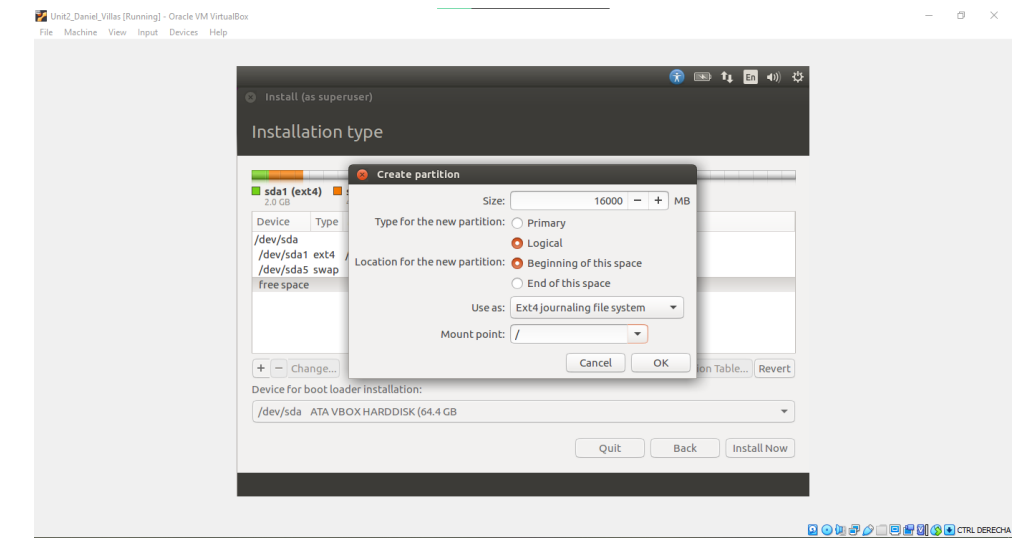
1. /boot with 1GB



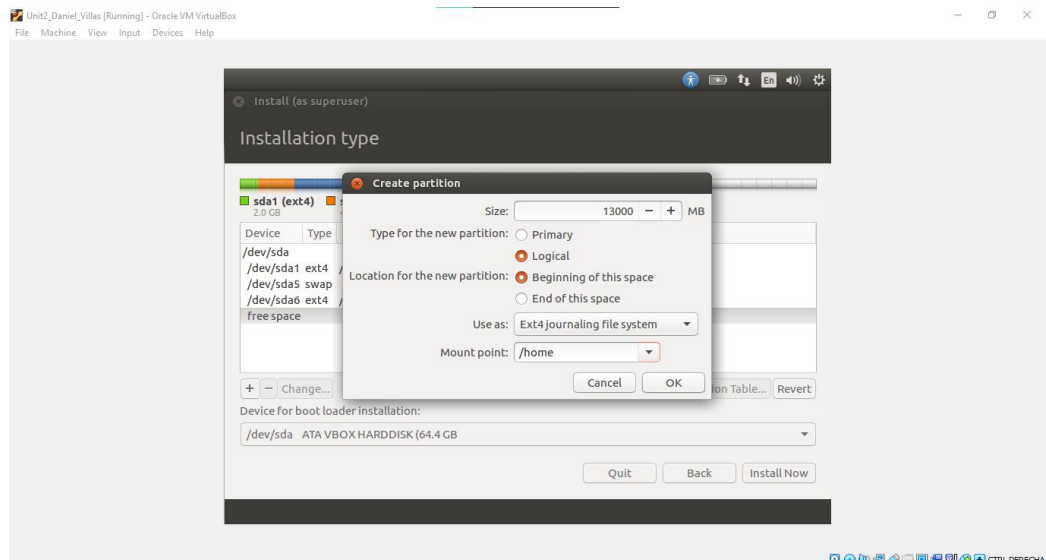
2. The swap area with 4GB



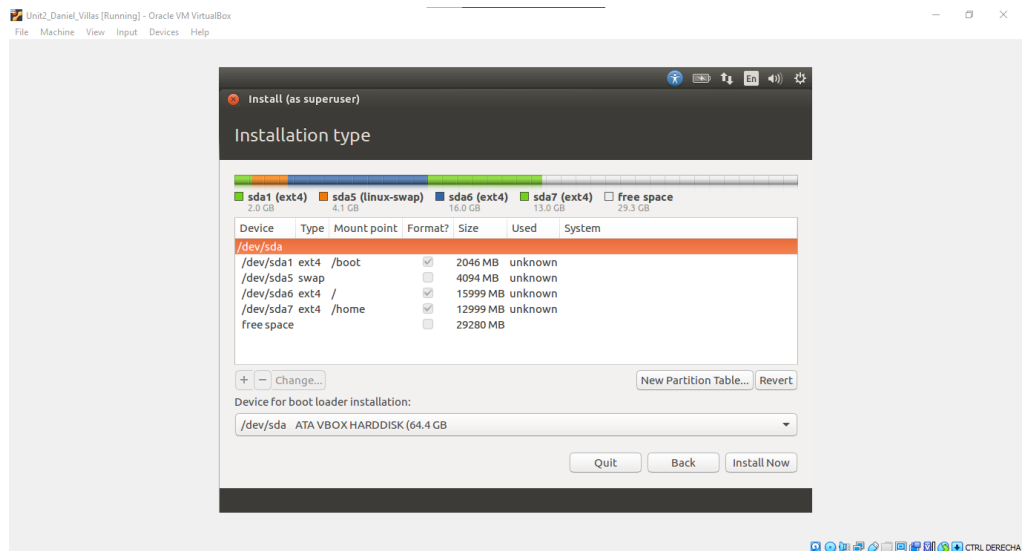
3. / with 16GB



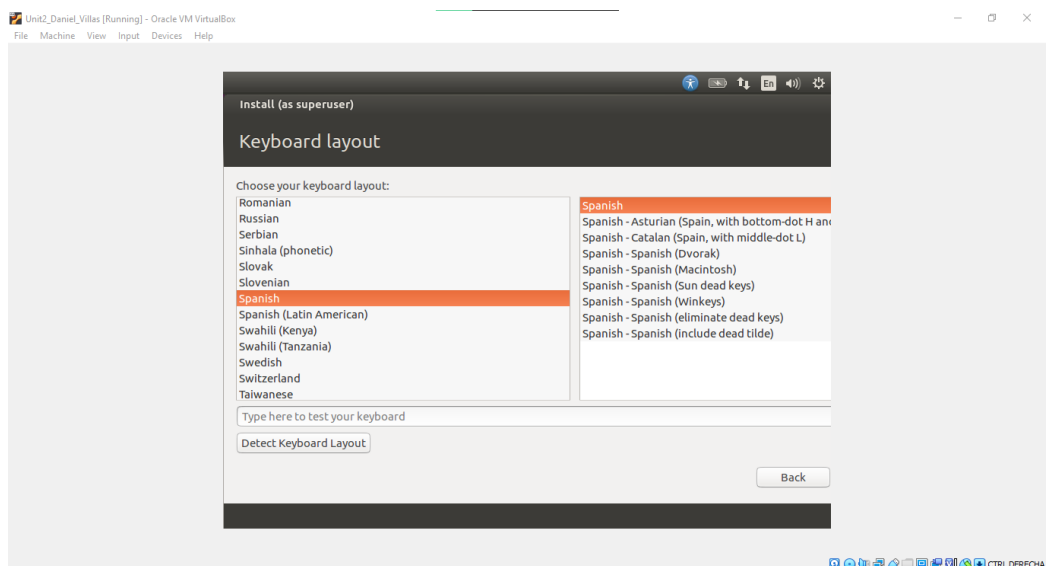
4. /home with 13 GB



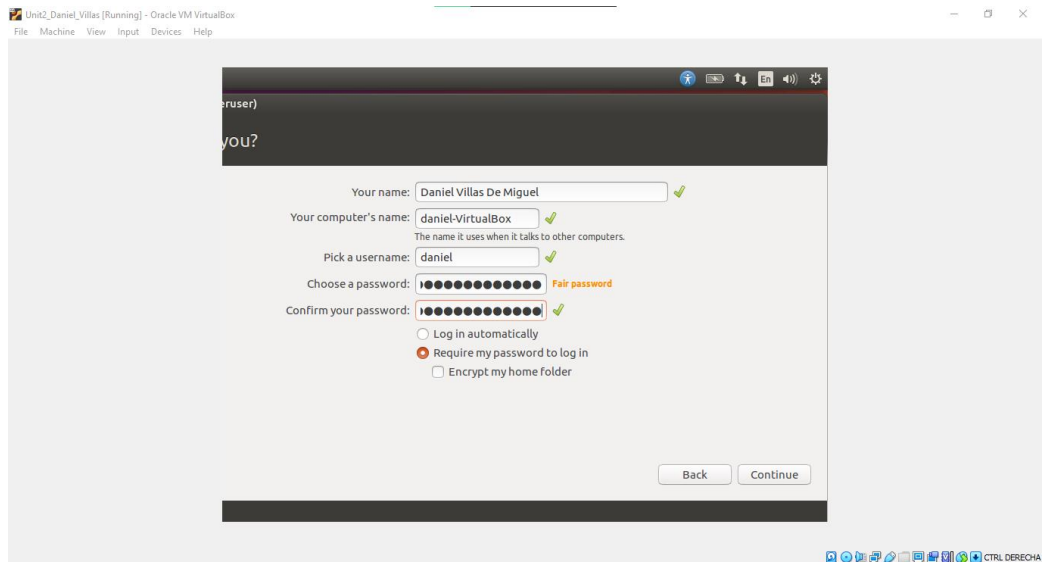
Now, there are all the partitions created, so only least click in “Install Now”.



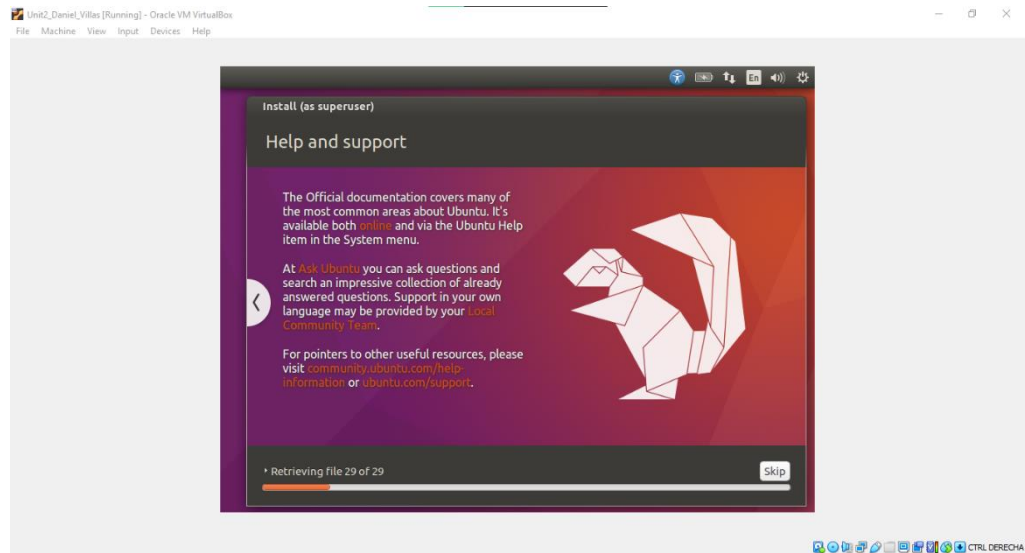
Then, select the keyboard layout, in this case in Spanish because of the special characters.



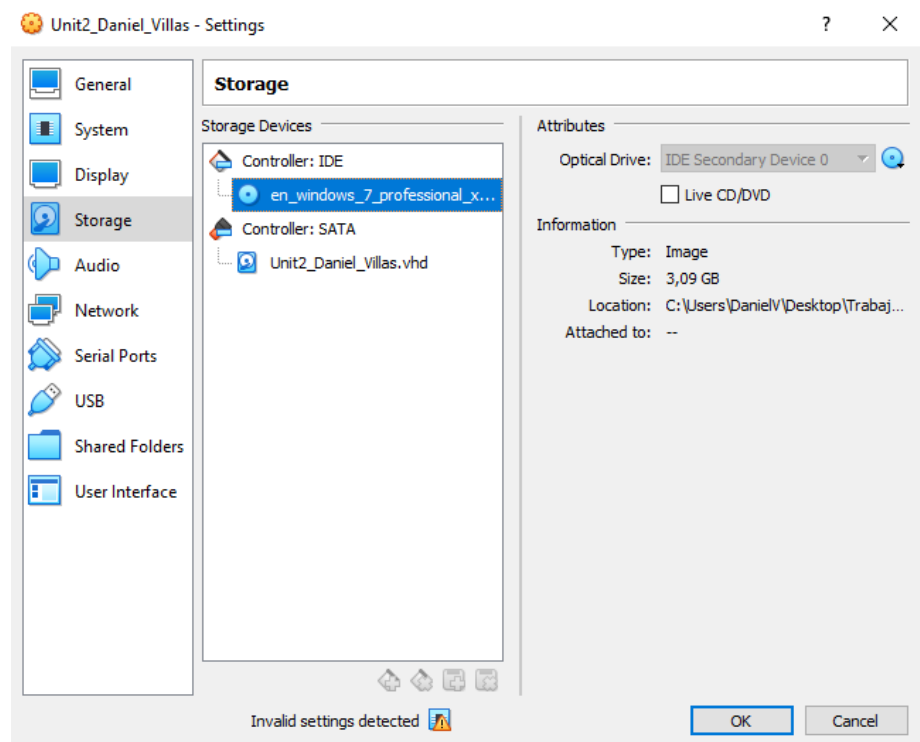
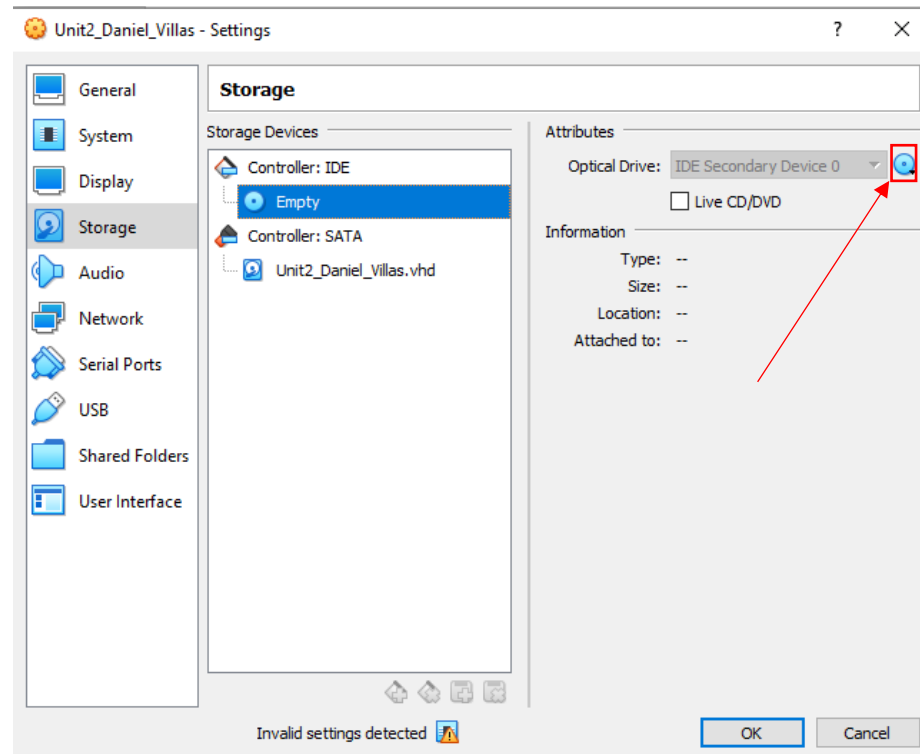
Afterwards, write the user's name, the computer's name, pick a username and chose a password.



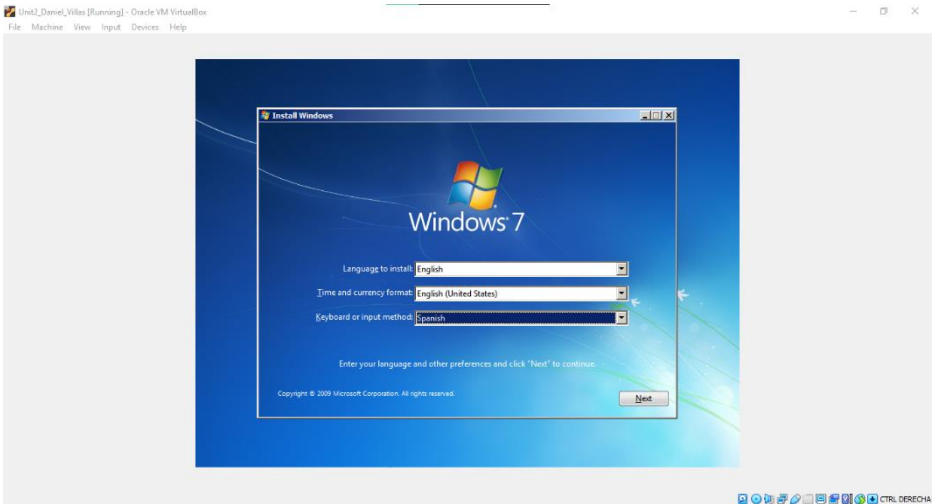
Finally, let the computer finish install Ubuntu.



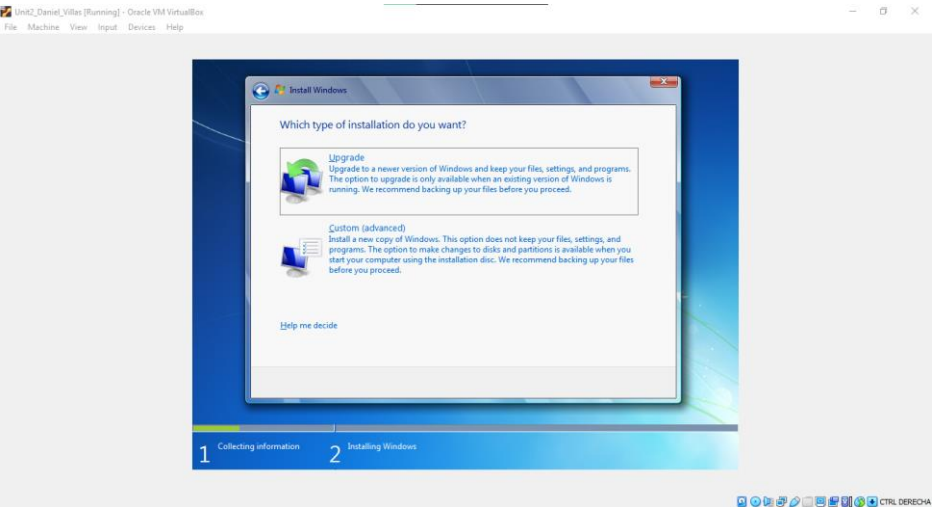
Then, to add another operative system, go again to the settings and go to “storage” select the empty controller IDE and chose the windows one.



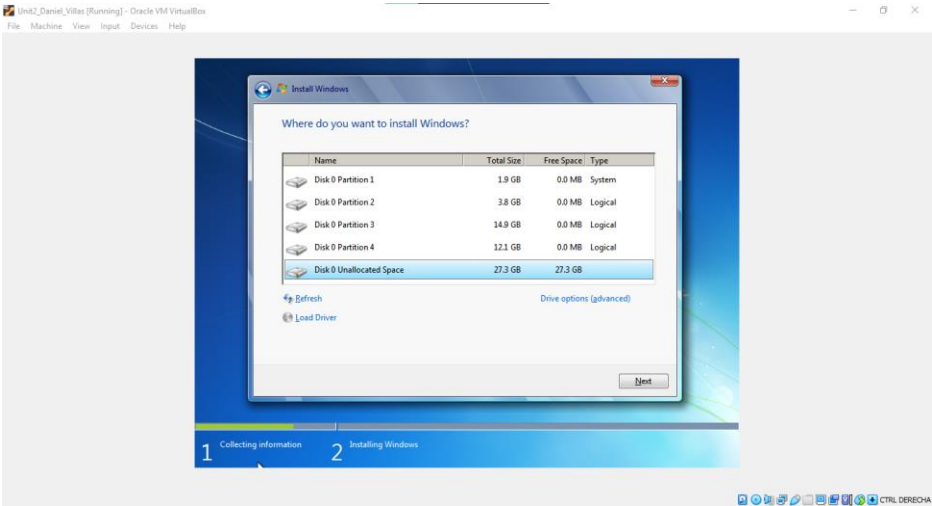
Select the language, the time and currency format and the keyboard layout.



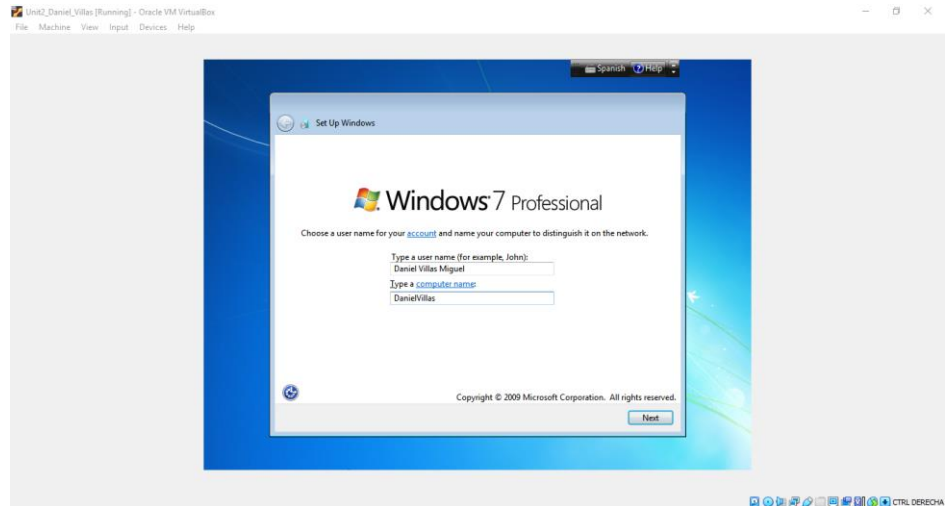
In the type of installation, chose “custom (advanced)”.



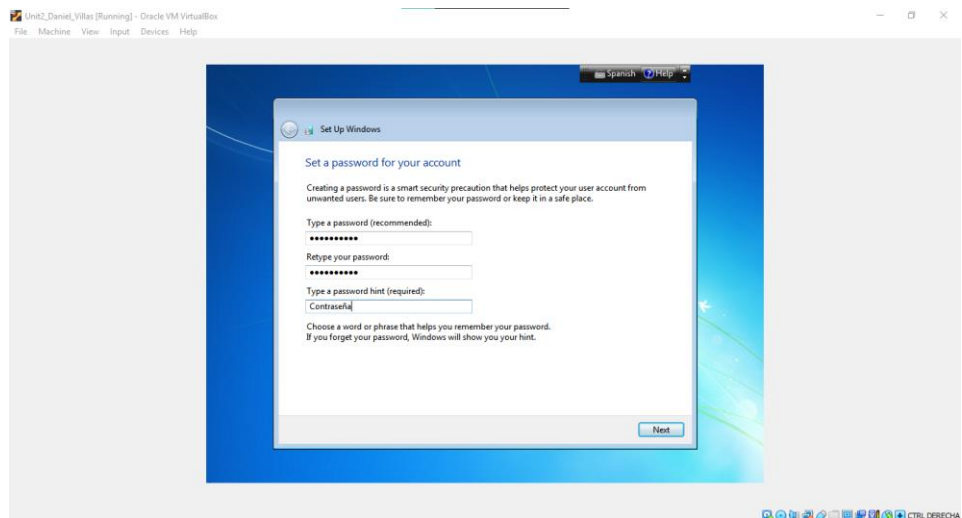
Now, select the disk with



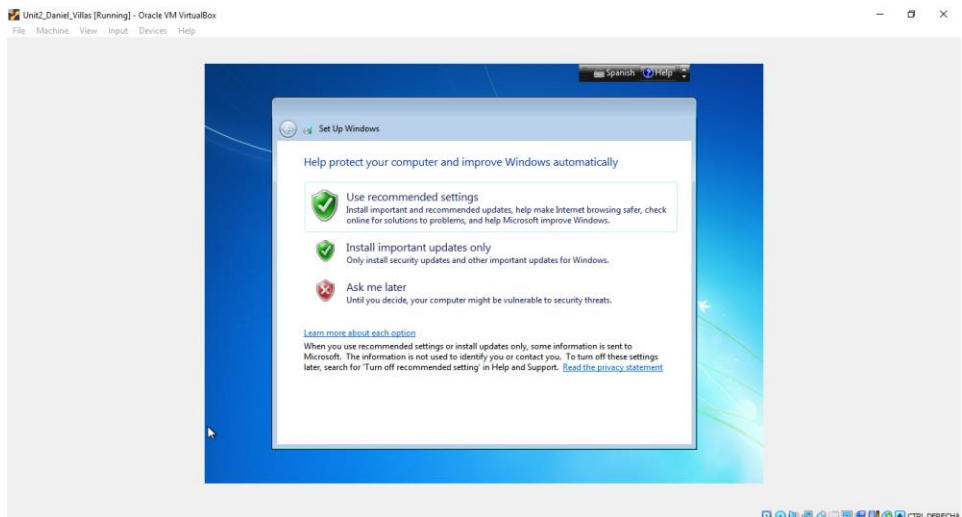
Select a username and a computer name.



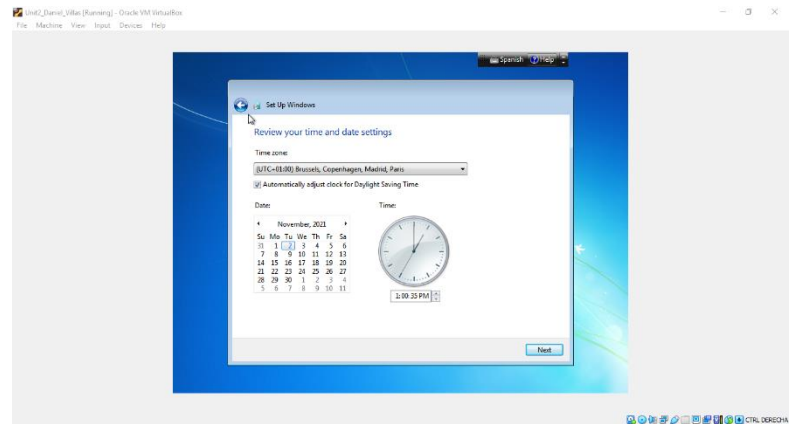
Type a password



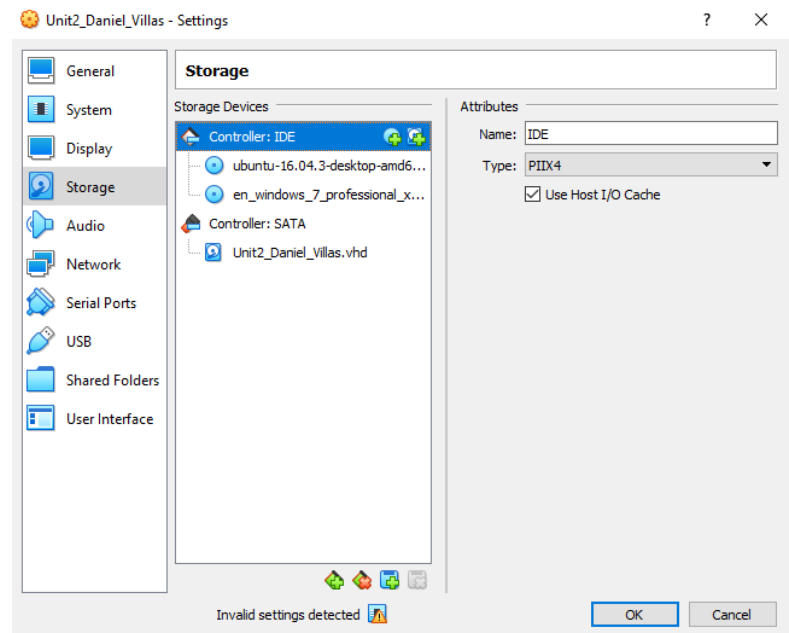
To help protect the computer, use the recommended settings.



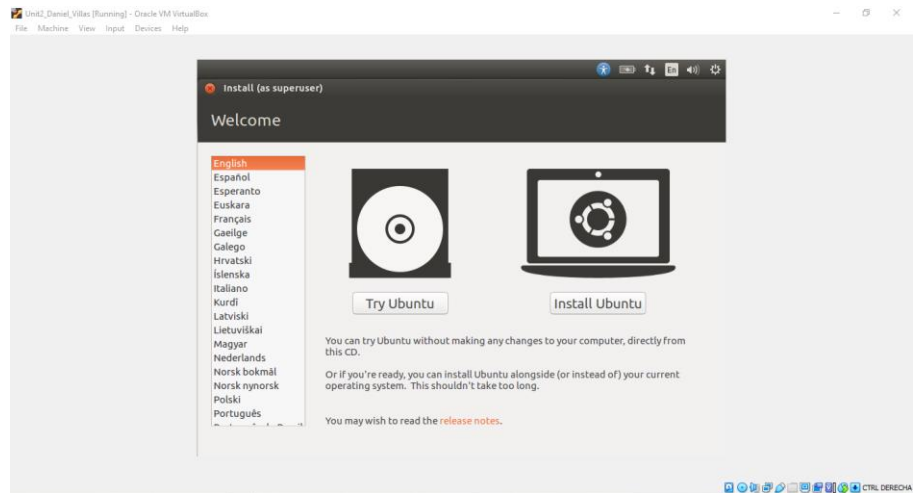
The time and date settings



Then, go back to the virtual machine settings, go to storage and add the other iso file.



Afterwards, select “try ubuntu”.



Once the virtual machine is booted with ubuntu, open the terminal to reinstall the GRUB 2. This are the commands:

-sudo mount /dev/sda6 /mnt

-sudo mount /dev/sda1 /mnt/boot

-for i in /dev /dev/pts /proc /sys /run; do sudo mount -B \$i /mnt\$i; done

-sudo chroot /mnt

-grub-install /dev/sda

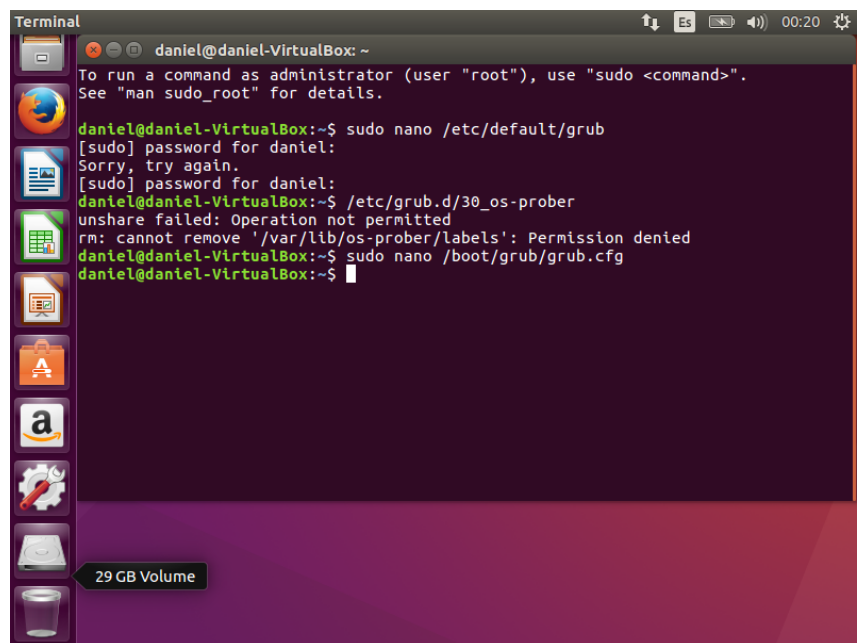
-update-grub

```
Terminal
ubuntu@ubuntu: ~
ubuntu@ubuntu:~$ sudo mount /dev/sda6 /mnt
mount: /dev/sda6 is already mounted or /mnt busy
/dev/sda6 is already mounted on /mnt
ubuntu@ubuntu:~$ sudo mount /dev/sda1 /mnt/boot
mount: /dev/sda1 is already mounted or /mnt/boot busy
/dev/sda1 is already mounted on /mnt
/dev/sda1 is already mounted on /mnt/boot
ubuntu@ubuntu:~$ for i in /dev /dev/pts /proc /sys /run; do sudo mount -B $i /mnt$i; done
bash: syntax error near unexpected token `inn'
ubuntu@ubuntu:~$ for i in /dev /dev/pts /proc /sys /run; do sudo mount -B $i /mnt$i; done
ubuntu@ubuntu:~$ sudo chroot /mnt
root@ubuntu:/# grub/install /dev/sda
bash: grub/install: No such file or directory
root@ubuntu:/# update/grub
bash: update/grub: No such file or directory
root@ubuntu:/# grub/install /dev/sda
bash: grub/install: No such file or directory
root@ubuntu:/# grub-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
root@ubuntu:/# update-grub
Generating grub configuration file ...
```

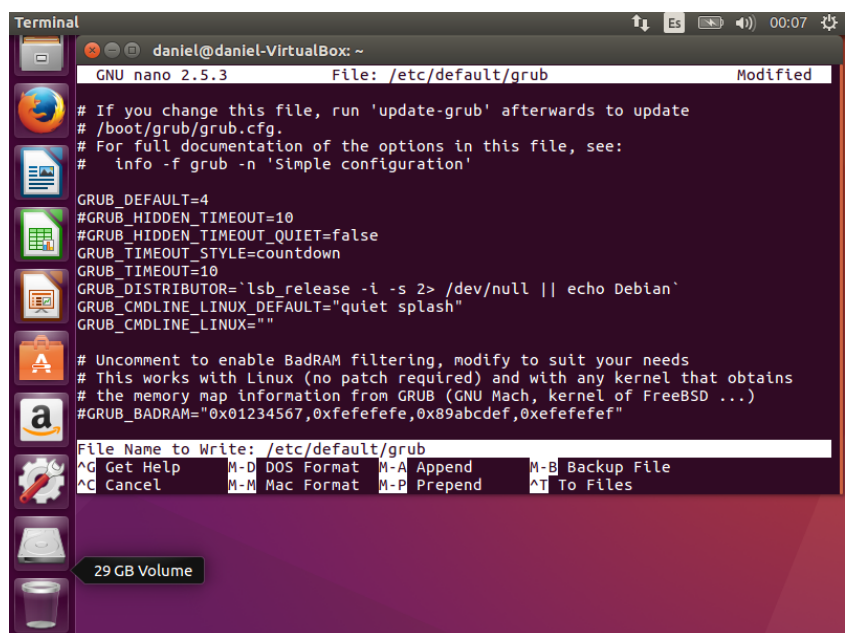
```
Terminal
ubuntu@ubuntu: ~
t$i; done
ubuntu@ubuntu:~$ sudo chroot /mnt
root@ubuntu:/# grub/install /dev/sda
bash: grub/install: No such file or directory
root@ubuntu:/# update/grub
bash: update/grub: No such file or directory
root@ubuntu:/# grub/install /dev/sda
bash: grub/install: No such file or directory
root@ubuntu:/# grub-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
root@ubuntu:/# update-grub
Generating grub configuration file ...
Warning: Setting GRUB_TIMEOUT to a non-zero value when GRUB_HIDDEN_TIMEOUT is set is no longer supported.
Found linux image: /boot/vmlinuz-4.10.0-28-generic
Found initrd image: /boot/initrd.img-4.10.0-28-generic
Found memtest86+ image: /memtest86+.elf
Found memtest86+ image: /memtest86+.bin
Found Windows 7 (loader) on /dev/sda3
done
root@ubuntu:/# exit
exit
ubuntu@ubuntu:~$
```

Then, write the command 'sudo nano /etc/default/grub' and type the following commands:

- GRUB_DEFAULT=4
- #GRUB_HIDDEN_TIMEOUT=10
- #GRUB_HIDDEN_TIMEOUT_QUIET=false
- GRUB_TIMEOUT_STYLE=countdown
- GRUB_TIMEOUT=10



A terminal window titled 'Terminal' showing a user named 'daniel' at 'daniel-VirtualBox'. The user enters 'sudo nano /etc/default/grub'. The terminal shows the password prompt, a failed attempt, and then the user enters '/etc/grub.d/30_os-prober'. This results in an error: 'unshare failed: Operation not permitted' and 'rm: cannot remove '/var/lib/os-prober/labels': Permission denied'. The user then enters 'sudo nano /boot/grub/grub.cfg'.



A terminal window titled 'Terminal' showing the 'nano' editor editing the file '/etc/default/grub'. The file content is as follows:

```
GNU nano 2.5.3 File: /etc/default/grub Modified

# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

GRUB_DEFAULT=4
#GRUB_HIDDEN_TIMEOUT=10
#GRUB_HIDDEN_TIMEOUT_QUIET=false
GRUB_TIMEOUT_STYLE=countdown
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
GRUB_CMDLINE_LINUX=""

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

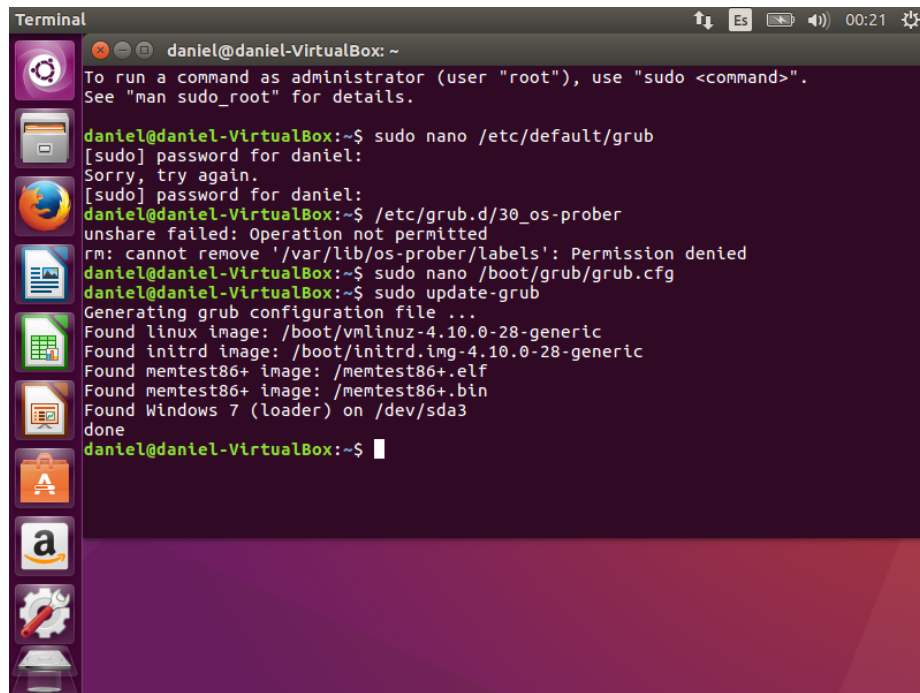
File Name to Write: /etc/default/grub
^G Get Help      ^M-D DOS Format  ^M-A Append      ^M-B Backup File
^C Cancel        ^M-M Mac Format  ^M-P Prepend     ^T To Files
```

Write the command ‘sudo nano /boot/grub/grub.cfg’, then write in the terminal the following lines:

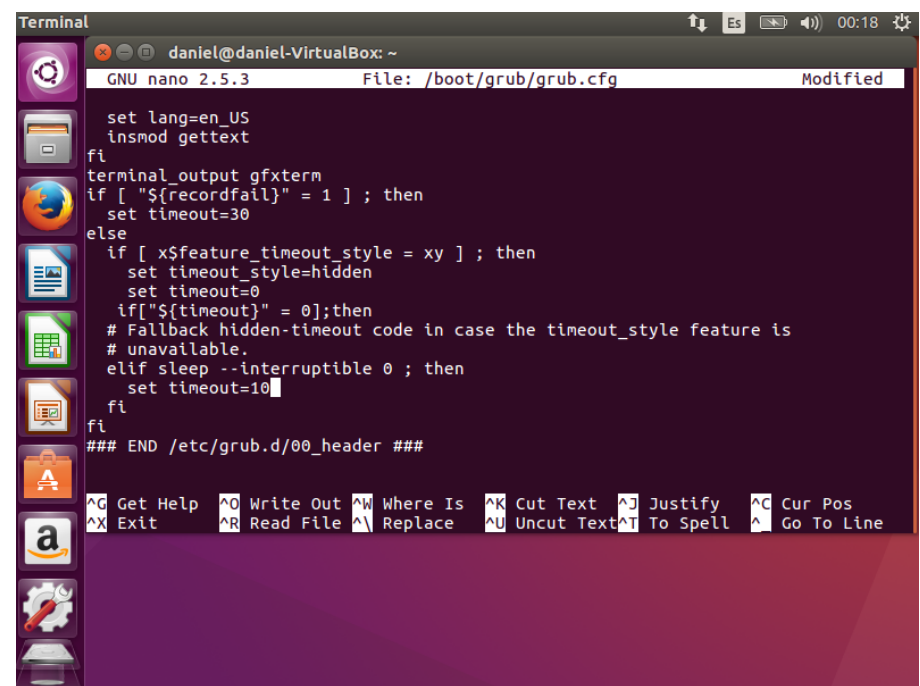
```
If["${timeout}" = 0]; then
```

```
    Set timeout=10
```

```
fi
```



A terminal window titled 'Terminal' showing a user named 'daniel' at 'daniel-VirtualBox'. The user enters the command 'sudo nano /etc/default/grub'. The terminal shows a password prompt, an error 'rm: cannot remove "/var/lib/os-prober/labels": Permission denied', and then the user enters 'sudo nano /boot/grub/grub.cfg'. The terminal shows the command 'sudo update-grub' being executed, which generates a GRUB configuration file and lists found images: 'linux image: /boot/vmlinuz-4.10.0-28-generic', 'initrd image: /boot/initrd.img-4.10.0-28-generic', 'memtest86+ image: /memtest86+.elf', and 'memtest86+ image: /memtest86+.bin'. The terminal ends with 'Found Windows 7 (loader) on /dev/sda3' and 'done'.



A terminal window titled 'Terminal' showing the 'GNU nano 2.5.3' editor editing the file '/boot/grub/grub.cfg'. The editor shows the following content:

```
set lang=en_US
insmod gettext
fi
terminal_output gfxterm
if [ "${recordfail}" = 1 ] ; then
    set timeout=30
else
    if [ x$feature_timeout_style = xy ] ; then
        set timeout_style=hidden
        set timeout=0
        if["${timeout}" = 0];then
            # Fallback hidden-timeout code in case the timeout_style feature is
            # unavailable.
            elif sleep --interruptible 0 ; then
                set timeout=10
            fi
        fi
    fi
fi
### END /etc/grub.d/00_header ###
```

The bottom of the terminal shows a list of nano editor shortcuts: ^G Get Help, ^O Write Out, ^W Where Is, ^R Cut Text, ^J Justify, ^C Cur Pos, ^X Exit, ^R Read File, ^\ Replace, ^U Uncut Text, ^T To Spell, ^_ Go To Line.

Finally, create the shared folder and the internet connection.

To create the shared folder go to “Shared Folders”, click the add folder icon, chose a folder in the computer and select “Auto-mount”.

To connect to internet, go to “Network” and set the option “Attached to” to “Bridged Adapter”.

Now, can see that the virtual machine is connected to internet correctly.

