

UNIVERSIDAD DE GUADALAJARA

**CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E
INGENIERÍAS**

INGENIERÍA EN COMPUTACIÓN(INCO)



MATERIA: Sistemas Operativos.

SECCIÓN: D07.

DOCENTE: Lupercio Coronel, Ramiro.

Integrantes:

Izmael Guzmán Murguía.
Maria Fernanda Quintero Suniga.
Andrea Margarita Aguilar Garcia.

Actividad 1: Linux-Ubuntu.

FECHA DE ENTREGA: 4 de febrero de 2022.

Índice

Introducción	3
¿Qué es LILO?	4
LILO y el proceso de arranque de x86	4
Desarrollo	5
Instalación de VirtualBox(Desde Parrot)	5
Creación de máquina virtual (Ubuntu)	6
Proceso de instalación de Ubuntu	7
Uso de lilo	9
Percances para el manejo de lilo	16
Control de procesos	17
Funcionamiento del Monitor del Sistema de linux	17
Manejo de interrupciones	22
Percances	22
Conclusión.....	23
Bibliografía.....	24

Introducción

Para esta actividad se presentará un pequeño tutorial donde se muestra el uso de lilo desde su proceso de instalación, configuración y uso, así como algunos problemas comunes que se suelen presentar y por supuesto su solución. Por otra parte se presentan también diversas herramientas para el control, visualización e interrupción de procesos y uso de recursos siendo estas herramientas ejecutadas en modo terminal casi en su totalidad, este tutorial se presentará desde una maquina virtual pero al usar VirtualBox las diferencias que se presentarán en comparación con un ordenador real serán nimias, además este tutorial pretende ser lo más completo posible por lo que se iniciara desde la instalación de VirtualBox, pasando por la creación de la máquina virtual, el procesos de instalación de la distro mostrando únicamente lo que se considera más importante ya que en general este proceso suele ser muy intuitivo y por supuesto el uso de las herramientas.

¿Que es LILO?

LILO (Linux LOader), desarrollado por John Coffman bajo licencia BSD, es el primer gestor de arranque del sistema GNU/Linux. Su configuración se encuentra en el archivo de texto `/etc/lilo.conf`. Ha ido dejando de forma progresiva su lugar a GRUB (GRand Unified Bootloader).

LILO es un gestor de arranque lo que quiere decir que es un programa que permite cargar un sistema operativo a la vez. Puede ofrecernos un menú para seleccionar una opción del sistema a arrancar.

LILO es un acrónimo de Linux LOader (cargador) y ha sido usado para arrancar linux en sistemas x86 por muchos años. Aunque ahora GRUB es el gestor de arranque por defecto, algunos prefieren usar LILO porque les es más familiar y otros porque GRUB puede causar problemas al arrancar determinado tipo de hardware.

LILO y el proceso de arranque de x86

LILO se carga asimismo en la memoria casi de forma idéntica a GRUB, con la diferencia de que es un gestor de dos etapas solamente.

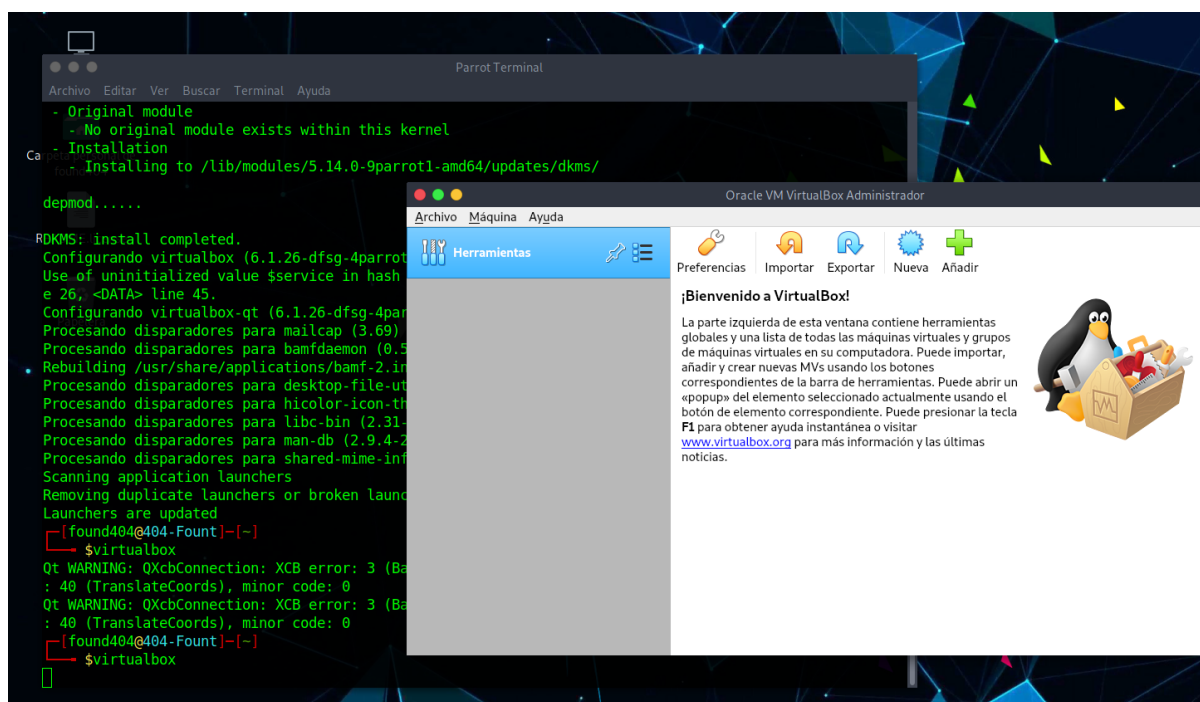
La etapa 1 o el gestor de arranque primario se lee en la memoria por la BIOS desde el MBR. El gestor de arranque primario existe en menos de 512 bytes de espacio en disco dentro del MBR. Su función es cargar la etapa 2 del gestor de arranque y pasarle la información de la geometría del disco.

La etapa 2 o el gestor de arranque secundario se lee en memoria.. El gestor de arranque secundario visualiza la pantalla inicial de Red Hat Enterprise Linux. Esta pantalla le permite seleccionar el sistema operativo o el kernel de Linux que desee arrancar.

La etapa 2 lee el sistema operativo o el kernel y lleva a cabo `initrd` en memoria. Una vez que LILO determina qué sistema operativo iniciar, éste lo carga en la memoria y transfiere el control de la máquina a ese sistema operativo.

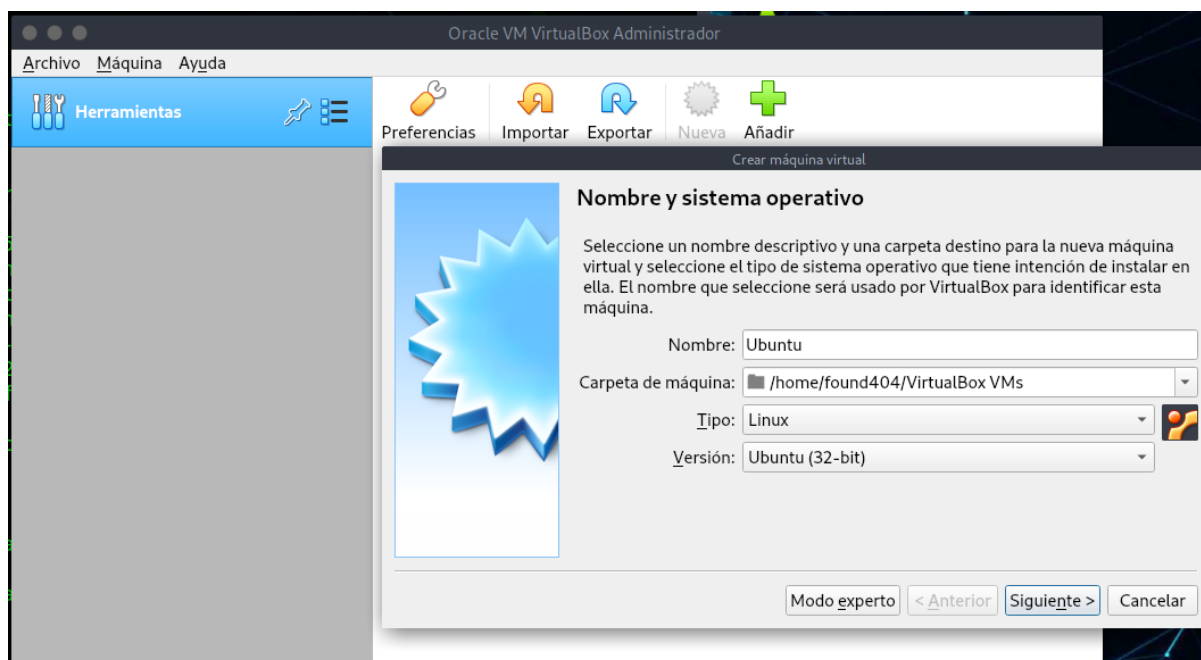
Una vez que la etapa 2 está en memoria, LILO visualiza la pantalla inicial de Red Hat Enterprise Linux con los diferentes sistemas operativos o kernels que han sido configurados para arrancar. Por defecto, si Red Hat Enterprise Linux es el único sistema instalado, linux será la única opción disponible. Si el sistema tiene múltiples procesadores habrá una opción `linux-up` para el kernel del procesador único y una opción `linux` para los kernel de múltiples procesadores (SMP). Si LILO está configurado para arrancar otros sistemas operativos, estas entradas de arranque también aparecerán en pantalla.

1. Instalamos VirtualBox con el comando `sudo apt-get install virtualbox...`

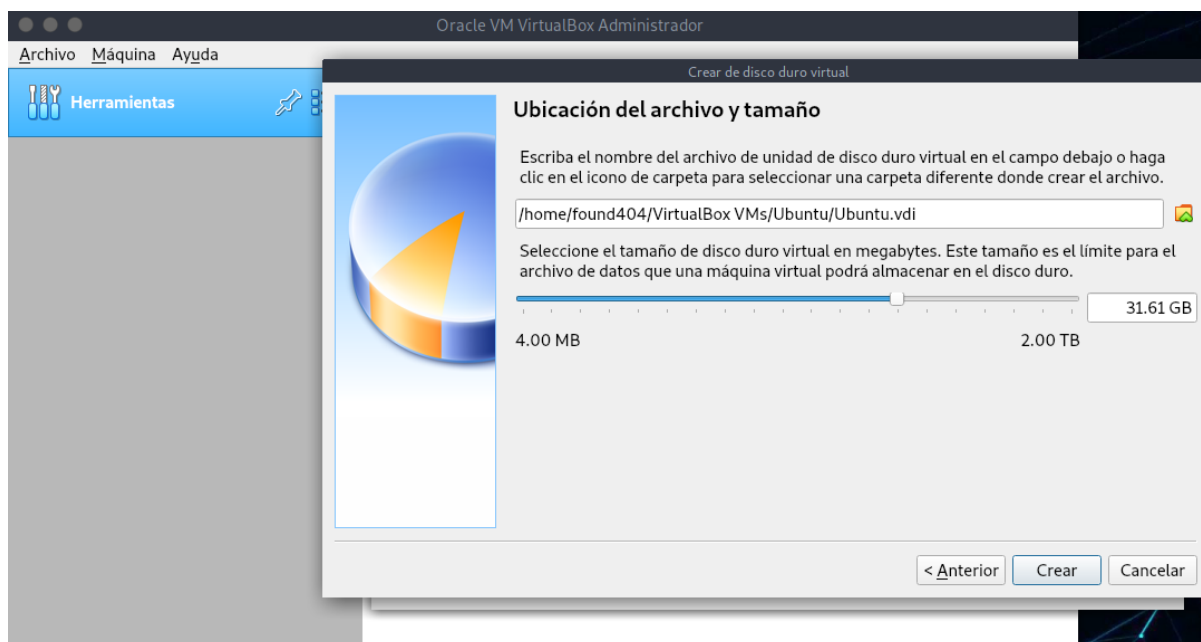


Creación de máquina virtual (Ubuntu):

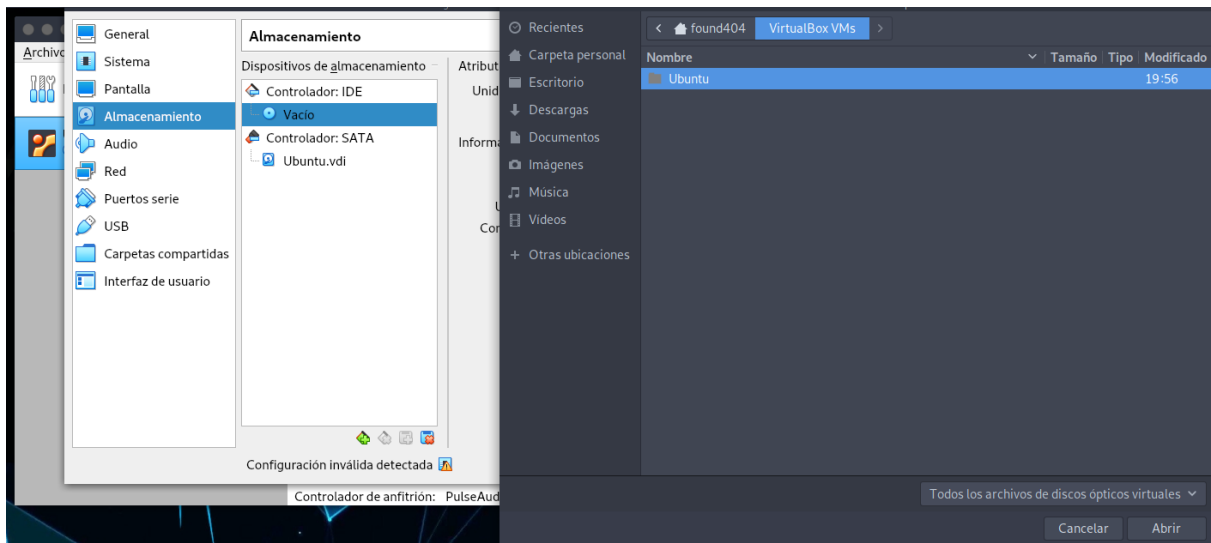
3. Crear nueva máquina virtual Ubuntu de 32 bits.



4. Asignación de almacenamiento.



5. Configuración del disco de arranque.

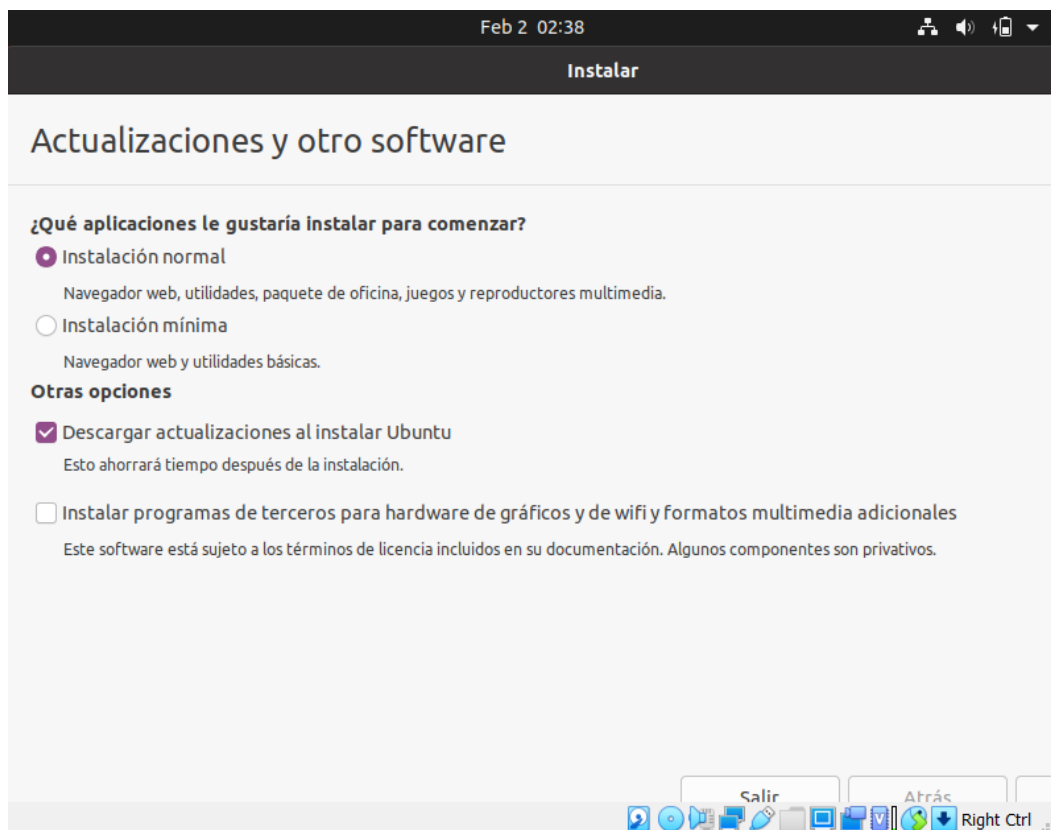


Proceso de instalación de Ubuntu.

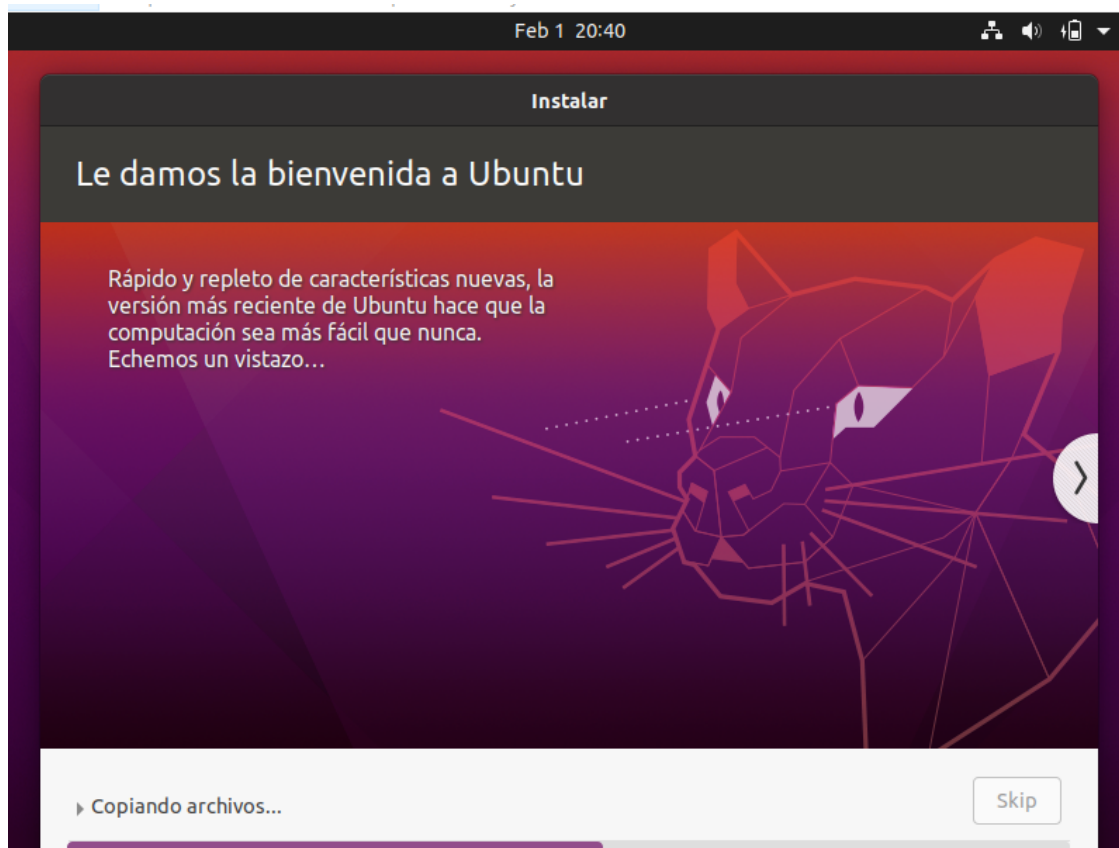
6. Inicio de la máquina virtual.



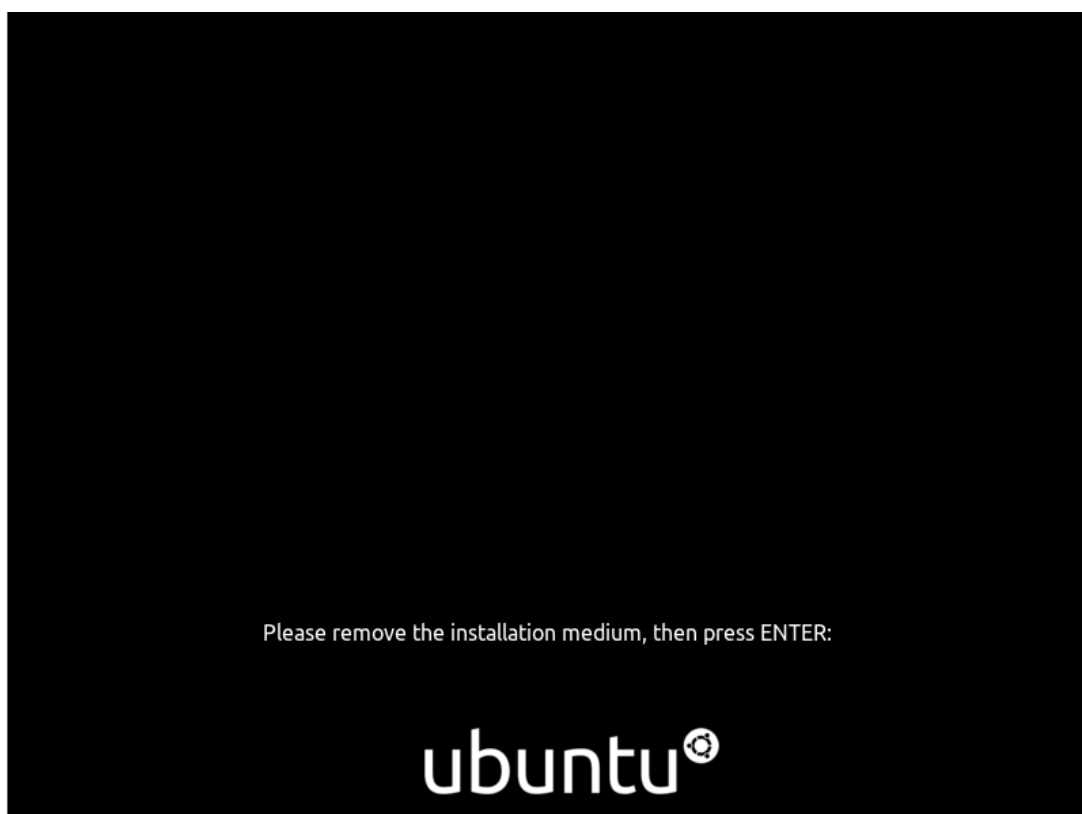
7. Instalación del SO.



8. Proceso de instalación.



9. Instalación terminada, solicitud de reinicio.



Uso de lilo.

10. Comprobación de lilo.

```
example@example-VirtualBox: ~  
(kazam:4065): Gtk-WARNING **: 21:38:00.937: Can't set a parent on widget which has a parent  
(kazam:4065): Gtk-WARNING **: 21:38:00.942: Can't set a parent on widget which has a parent  
^CTraceback (most recent call last):  
  File "/usr/bin/kazam", line 153, in <module>  
    Gtk.main()  
  File "/usr/lib/python3/dist-packages/gi/overrides/Gtk.py", line 1632, in main  
    return _Gtk_main(*args, **kwargs)  
  File "/usr/lib/python3.8/contextlib.py", line 120, in __exit__  
    next(self.gen)  
  File "/usr/lib/python3/dist-packages/gi/_ossighelper.py", line 251, in register_sigint_fallback  
    signal.default_int_handler(signal.SIGINT, None)  
KeyboardInterrupt  
  
example@example-VirtualBox:~$ lilo  
  
No se ha encontrado la orden «lilo», pero se puede instalar con:  
  
sudo apt install lilo  
  
example@example-VirtualBox:~$
```

11. Instalación de lilo con apt.

```

example@example-VirtualBox: ~
sudo apt install lilo

example@example-VirtualBox:~$ sudo apt-get install lilo
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  lilo
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 263 kB de archivos.
Se utilizarán 711 kB de espacio de disco adicional después de esta operación.
Des:1 http://mx.archive.ubuntu.com/ubuntu/focal/main amd64 lilo amd64 1:24.2-5 [
263 kB]
Descargados 263 kB en 1s (335 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete lilo previamente no seleccionado.
(Leyendo la base de datos ... 186526 ficheros o directorios instalados actualmen
te.)
Preparando para desempaquetar .../lilo_1%3a24.2-5_amd64.deb ...
Desempaquetando lilo (1:24.2-5) ...
Configurando lilo (1:24.2-5) ...
Procesando disparadores para man-db (2.9.1-1) ...
example@example-VirtualBox:~$

```

12. Podremos ver las operaciones que es capaz de hacer lilo utilizando **lilo -h**.

```

example@example-VirtualBox: ~
example@example-VirtualBox:~$ lilo -h
usage: lilo [ -C config_file ] [ -q [ -m map_file ] [ -v N | -v ... ]
      lilo [ -C config_file ] [ -b boot_device ] [ -c ] [ -g | -l | -L ]
      [ -F ] [ -i boot_loader ] [ -m map_file ] [ -d delay ]
      [ -v N | -v ... ] [ -t ] [ -s save_file | -S save_file ]
      [ -p ] [ -P fix | -P ignore ] [ -r root_dir ] [ -w | -w+ ]
      lilo [ -C config_file ] [ -m map_file ] -R [ word ... ]
      lilo [ -C config_file ] -I name [ options ]
      lilo [ -C config_file ] [ -s save_file ] -u | -U [ boot_device ]
      lilo -H                      install only to active discs (RAID-1)
      lilo -A /dev/XXX [ N ]        inquire/activate a partition
      lilo -M /dev/XXX [ mbr | ext ] install master boot record
      lilo -T help                  list additional options
      lilo -X                      internal compile-time options
      lilo -V [ -v ]               version information

example@example-VirtualBox:~$

```

13. Alternativamente podemos obtener la misma información con **lilo --version**.

```
example@example-VirtualBox: ~
example@example-VirtualBox:~$ lilo --version
usage: lilo [ -C config_file ] -q [ -m map_file ] [ -v N | -v ... ]
      lilo [ -C config_file ] [ -b boot_device ] [ -c ] [ -g | -l | -L ]
      [ -F ] [ -i boot_loader ] [ -m map_file ] [ -d delay ]
      [ -v N | -v ... ] [ -t ] [ -s save_file | -S save_file ]
      [ -p ][ -P fix | -P ignore ] [ -r root_dir ] [ -w | -w+ ]
      lilo [ -C config_file ] [ -m map_file ] -R [ word ... ]
      lilo [ -C config_file ] -I name [ options ]
      lilo [ -C config_file ] [ -s save_file ] -u | -U [ boot_device ]
      lilo -H                                install only to active discs (RAID-1)
      lilo -A /dev/XXX [ N ]                 inquire/activate a partition
      lilo -M /dev/XXX [ mbr | ext ]         install master boot record
      lilo -T help                           list additional options
      lilo -X                                internal compile-time options
      lilo -V [ -v ]                         version information

example@example-VirtualBox:~$
```

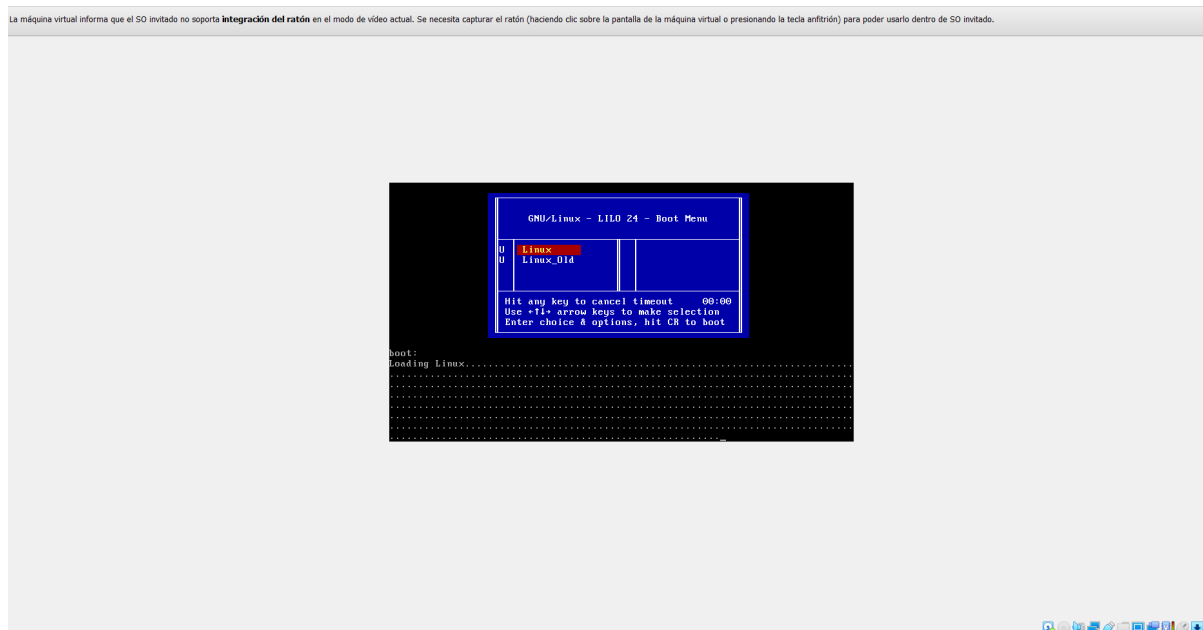
14. Ejecutamos `sudo liloconfig` para crear los ficheros.

```
example@example-VirtualBox: ~
example@example-VirtualBox:~$ /sbin/lilo
Fatal: Cannot open: /etc/lilo.conf
example@example-VirtualBox:~$ /etc/lilo.conf
bash: /etc/lilo.conf: No existe el archivo o el directorio
example@example-VirtualBox:~$ sudo liloconfig
2 images '/boot/vmlinuz*' found.
New file created as: /etc/lilo.conf
Now you must execute '/sbin/lilo' to activate this new configuration!
```

15. Reiniciamos la máquina virtual.

```
example@example-VirtualBox: ~
example@example-VirtualBox:~$ reboot
```

16. Podemos verificar que funciona correctamente al iniciar el sistema operativo si nos muestra el menú boot.



17. Permiso denegado por restricción de archivo.

```

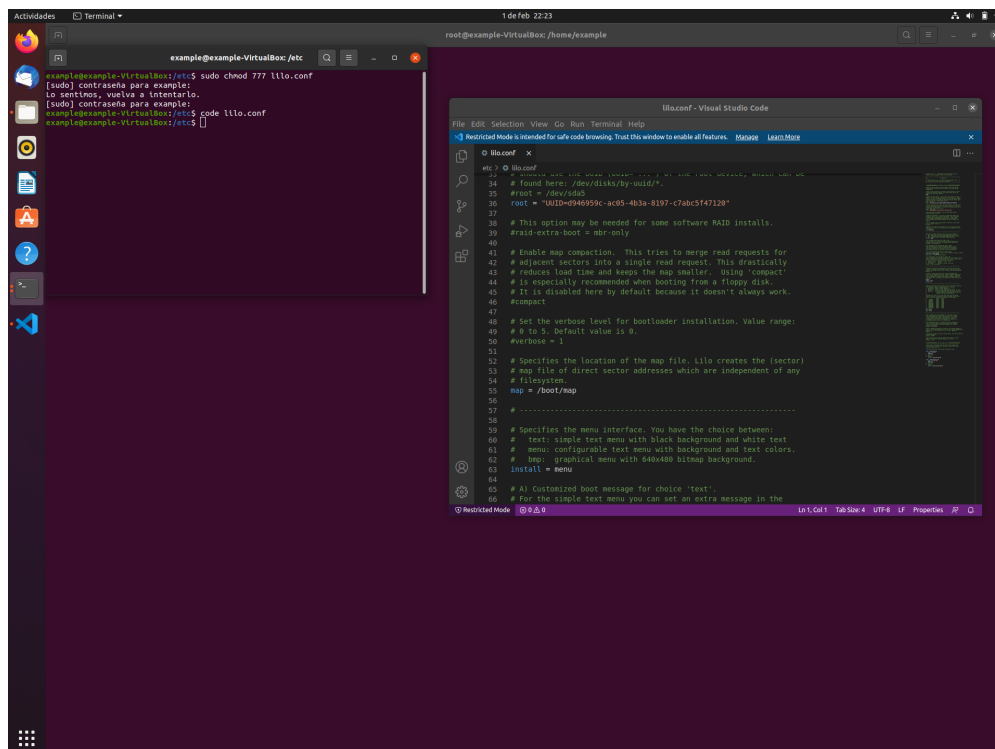
root@example-VirtualBox: /home/example

example@example-VirtualBox:~$ lilo --version
usage: lilo [ -C config_file ] -q [ -m map_file ] [ -v N | -v ... ]
      lilo [ -C config_file ] [ -b boot_device ] [ -c ] [ -g | -l | -L ]
      [ -F ] [ -i boot_loader ] [ -m map_file ] [ -d delay ]
      [ -v N | -v ... ] [ -t ] [ -s save_file | -S save_file ]
      [ -p ] [ -P fix | -P ignore ] [ -r root_dir ] [ -w | -w+ ]
      lilo [ -C config_file ] [ -m map_file ] -R [ word ... ]
      lilo [ -C config_file ] -I name [ options ]
      lilo [ -C config_file ] [ -s save_file ] -u | -U [ boot_device ]
      lilo -H                               install only to active discs (RAID-1)
      lilo -A /dev/XXX [ N ]                 inquire/activate a partition
      lilo -M /dev/XXX [ mbr | ext ]         install master boot record
      lilo -T help                           list additional options
      lilo -X                               internal compile-time options
      lilo -V [ -v ]                         version information

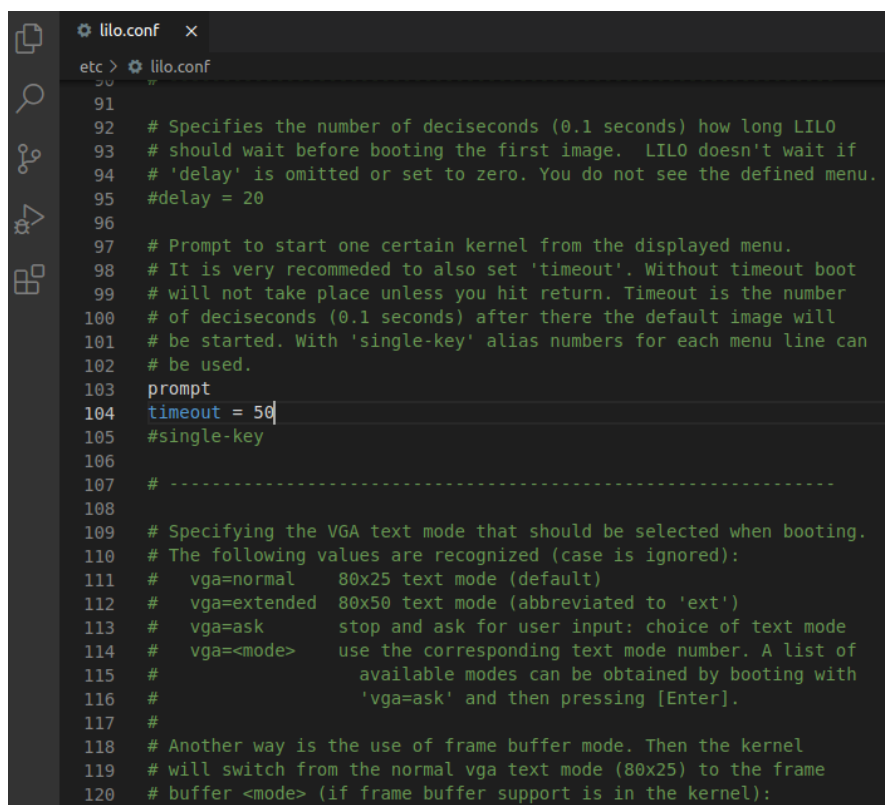
example@example-VirtualBox:~$ sudo su
[sudo] contraseña para example:
root@example-VirtualBox:/home/example# /etc/lilo.conf
bash: /etc/lilo.conf: Permiso denegado
root@example-VirtualBox:/home/example#

```

18. Concedemos todos los permisos con **sudo chmod 777 lilo.conf**, modificamos el archivo **lilo.conf** con visual studio code.



19. Modificación de gestor de arranque lilo a 5 segundos (por mitad de como era originalmente).



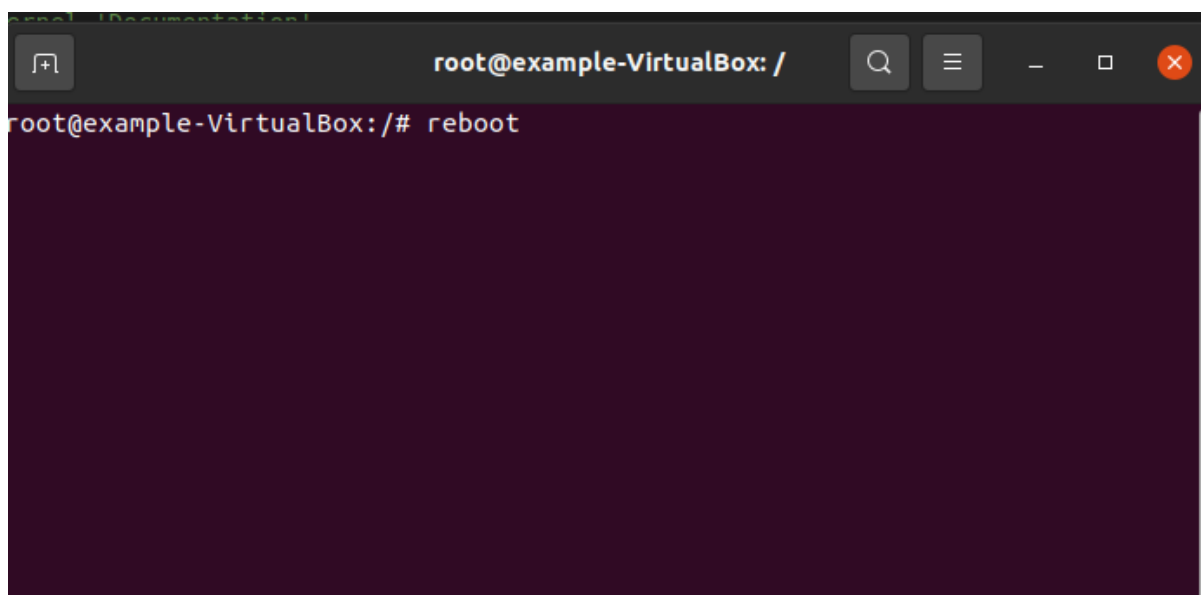
20. Modificación de imágenes del sistema operativo.

```
162
163
164 # ##### LILO per-image section #####
165
166 # Each image is configured with the linux kernel (=image) and
167 # usually with the initrd file. Configure all GNU/Linux systems
168 # on other partitions, too.
169
170 # Warning: labels must be less than 16 characters long.
171
172 image = /boot/vmlinuz
173 |   label = "Linux"
174 |   read-only
175 |   # restricted
176 |   # alias = 1
177 |   # optional
178 |   initrd = /boot/initrd.img
179
180 image = /boot/vmlinuz.old
181 |   label = "Linux Old"
182 |   read-only
183 |   # restricted
184 |   # alias = 2
185 |   # optional
186 |   initrd = /boot/initrd.img.old
187 other = /dev/hda1
188 |   label = "Windows"
189
190
```

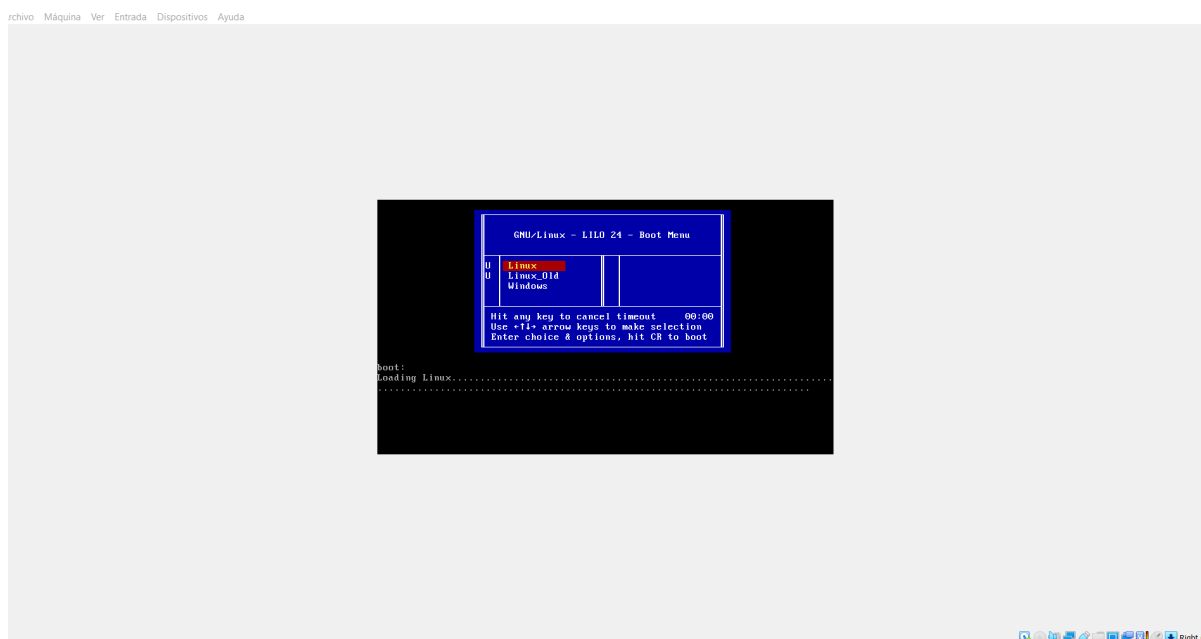
21. Aplicamos los cambios.

```
root@example-VirtualBox: /
root@example-VirtualBox: /# /sbin/lilo
Warning: /etc/lilo.conf should be writable only for root
Added Linux + *
Added Linux_Old +
Warning: 'other = /boot/vmlinuz.old' specifies a file that is longer
than a single sector.
Added Windows
2 warnings were issued.
root@example-VirtualBox: /#
```

22. Reiniciamos.



23. Ahora es posible visualizar la nueva configuración.



Percances para el manejo de lilo:

El único problema que se presentó al momento de realizar esta parte de la actividad fue el permiso denegado para abrir el archivo lilo.conf incluso para el superusuario o root, para solucionarlo fue necesario darle todos los permisos posibles al archivo mediante **chmod 777** en modo superusuario, esto facilitó el abrir el archivo sin ningún tipo de problema y además poder editarlo.

```

root@example-VirtualBox: /home/example

example@example-VirtualBox:~$ lilo --version
usage: lilo [ -C config_file ] -q [ -m map_file ] [ -v N | -v ... ]
      lilo [ -C config_file ] [ -b boot_device ] [ -c ] [ -g | -l | -L ]
      [ -F ] [ -i boot_loader ] [ -m map_file ] [ -d delay ]
      [ -v N | -v ... ] [ -t ] [ -s save_file | -S save_file ]
      [ -p ][ -P fix | -P ignore ] [ -r root_dir ] [ -w | -w+ ]
      lilo [ -C config_file ] [ -m map_file ] -R [ word ... ]
      lilo [ -C config_file ] -I name [ options ]
      lilo [ -C config_file ] [ -s save_file ] -u | -U [ boot_device ]
      lilo -H                      install only to active discs (RAID-1)
      lilo -A /dev/XXX [ N ]        inquire/activate a partition
      lilo -M /dev/XXX [ mbr | ext ] install master boot record
      lilo -T help                  list additional options
      lilo -X                      internal compile-time options
      lilo -V [ -v ]               version information

example@example-VirtualBox:~$ sudo su
[sudo] contraseña para example:
root@example-VirtualBox:/home/example# /etc/lilo.conf
bash: /etc/lilo.conf: Permiso denegado
root@example-VirtualBox:/home/example#

```


Control de procesos:

Funcionamiento del Monitor del Sistema de linux:

Linux como se sabe es un sistema operativo multitarea y multiusuario, esto quiere decir que diferentes procesos pueden operar simultáneamente sin interferirse unos con los otros, esto implica que cada proceso tiene la ilusión de que es el único proceso del sistema y que tiene acceso a todos los recursos del sistema operativo. Una de las utilidades más prácticas que traen los sistemas operativos hoy en día es el administrador de tareas que además incorpora otra ventana donde podemos visualizar el uso de los recursos como la memoria, red y CPUs, podemos mencionar además que esta interfaz nos permite controlar los procesos cambiándoles el grado de prioridad, terminando y demás operaciones que nos facilitan el uso del ordenador, es así que esta herramienta es muy útil pero no imprescindible ya que podemos hacer lo mismo e incluso mostrar más detalles desde la terminal y por supuesto manejar los procesos y recursos de una manera más personalizada ajustándose más a lo que necesitamos.

1. Para mostrar un listado dinámico de procesos, nombres usuarios o recursos que se están usando en el momento de la petición se utiliza **top**.

```

top - 21:03:02 up 3:40, 1 user, load average: 0.50, 1.17, 1.68
Tasks: 197 total, 2 running, 195 sleeping, 0 stopped, 0 zombie
%Cpu(s): 17.6 us, 4.1 sy, 0.0 ni, 78.1 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st
MiB Mem : 5882.0 total, 163.7 free, 2140.0 used, 3578.3 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used, 3363.1 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 812  root        20   0 1360500 202088 76276 S 20.9   3.4   17:27.93 Xorg
37109 found404  20   0 2888384 398096 110224 S 12.3   6.6   7:10.28 Web Content
36874 found404  20   0 3806432 619436 202432 S  8.3  10.3  28:50.49 firefox-esr
38401 found404  20   0 1176344 105556 64892 S  6.6   1.8   0:10.13 kazam
37429 found404  20   0 2477024 142552 98236 R  5.6   2.4   0:27.03 Web Content
37811 found404  20   0 481484 53364 35796 S  2.6   0.9   0:07.74 mate-terminal
 1171 found404  20   0 477012 30468 13924 S  2.3   0.5   1:32.47 marco
 1220 found404  20   0 465716 60008 38792 S  2.3   1.0   0:27.25 wnck-applet
37206 found404  20   0 3012572 343260 148092 S  1.0   5.7   3:48.04 Web Content
37840 found404  20   0 10228 4048 3308 R  1.0   0.1   0:06.61 top
  786 root        20   0 1684460 35004 4224 S  0.7   0.6   1:05.84 opensnitchd
37061 found404  20   0 3083352 349996 151124 S  0.7   5.8   5:24.10 Web Content
  13  root        20   0 0 0 0 I  0.3   0.0   0:10.04 rcu_sched
 1163 found404  20   0 889072 32460 16520 S  0.3   0.5   0:13.45 mate-settings-d
 1165 found404  20   0 165820 3532 2744 S  0.3   0.1   0:04.48 at-spi2-registr
 1185 found404  20   0 472656 24924 11408 S  0.3   0.4   0:07.72 mate-panel
 1203 found404  20   0 915668 72224 48192 S  0.3   1.2   0:16.51 caja
 1237 found404  20   0 471404 36068 22904 S  0.3   0.6   0:03.84 mate-power-mana
 1277 found404  20   0 401648 19128 7588 S  0.3   0.3   0:03.08 clock-applet
 1278 found404  20   0 345464 13548 4988 S  0.3   0.2   0:17.62 mate-multiloa-
37942 found404  20   0 2384480 70184 57420 S  0.3   1.2   0:00.19 Web Content
38028 root        20   0 0 0 0 I  0.3   0.0   0:00.54 kworker/1:2-events
  1  root        20   0 165312 8012 5012 S  0.0   0.1   0:07.35 systemd
  2  root        20   0 0 0 0 S  0.0   0.0   0:00.02 kthreadd
  3  root        0 -20 0 0 0 I  0.0   0.0   0:00.00 rcu_gp
  4  root        0 -20 0 0 0 I  0.0   0.0   0:00.00 rcu_par_gp
  6  root        0 -20 0 0 0 I  0.0   0.0   0:00.00 kworker/0:0H-events_highpri
  9  root        0 -20 0 0 0 I  0.0   0.0   0:00.00 mm_percpu_wq
 10  root        20   0 0 0 0 S  0.0   0.0   0:00.00 rcu_tasks_rude
 11  root        20   0 0 0 0 S  0.0   0.0   0:00.00 rcu_tasks_trace
 12  root        20   0 0 0 0 S  0.0   0.0   0:00.62 ksoftirqd/0
 14  root        rt   0 0 0 0 S  0.0   0.0   0:00.20 migration/0

```

2. **htop** funciona muy similar al anterior sin embargo ahora se proporciona un menú de configuración al estilo de las aplicaciones DOS.

```

0[|||||] 22.1% Tasks: 89, 422 thr; 2 running
1[|||||] 32.7% Load average: 0.89 1.11 1.55
2[|||||] 30.5% Uptime: 03:44:14
Mem[|||||] 2.44G/5.74G
Swp[|||||] 0K/0K

PID USER      PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
36874 found404 20    0 3717M  707M  198M  S  24.6 12.0 30:17.79 firefox-esr
812  root      20    0 1393M  210M  89168  R  20.6 3.6 17:48.26 /usr/lib/xorg/Xorg :0 -seat seat0 -auth /var/run/lightdm/r
38401 found404 20    0 1223M  122M  74144  S  8.6 2.1 0:11.89 /usr/bin/python3 /usr/bin/kazam
38519 found404 20    0 8940  4784  3504  R  7.3 0.1 0:01.30 htop
37061 found404 20    0 3033M  397M  146M  S  6.6 6.8 6:03.92 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 4 -
1220 found404 20    0 455M  64988 43140  S  4.6 1.1 0:27.96 /usr/lib/mate-panel/wnck-applet
36983 found404 20    0 33.2G  195M  79900  S  4.6 3.3 0:46.99 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 2 -
37429 found404 20    0 2424M  145M  101M  S  4.0 2.5 0:37.76 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 8 -
1171 found404 20    0 465M  30468 13924  S  2.7 0.5 1:33.86 marco
36880 found404 20    0 3717M  707M  198M  S  2.7 12.0 0:49.38 firefox-esr
939  root      20    0 1393M  210M  89168  S  2.0 3.6 0:59.16 /usr/lib/xorg/Xorg :0 -seat seat0 -auth /var/run/lightdm/r
36906 found404 20    0 3717M  707M  198M  S  2.0 12.0 10:27.65 firefox-esr
37811 found404 20    0 470M  53344 35764  S  2.0 0.9 0:09.61 mate-terminal
1163 found404 20    0 868M  31968 16020  S  1.3 0.5 0:13.81 /usr/bin/mate-settings-daemon
1278 found404 20    0 337M  16268 7708  S  1.3 0.3 0:18.13 /usr/lib/mate-applets/mate-multiloader-applet
36883 found404 20    0 3717M  707M  198M  S  1.3 12.0 1:14.07 firefox-esr
36919 found404 20    0 3717M  707M  198M  S  1.3 12.0 0:27.08 firefox-esr
37206 found404 20    0 2941M  337M  144M  S  1.3 5.7 3:51.14 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 6 -
38058 found404 20    0 3717M  707M  198M  S  1.3 12.0 0:45.58 firefox-esr
689  root      20    0 231M  4536  3616  S  0.7 0.1 0:00.55 /usr/libexec/accounts-daemon
911  root      20    0 1644M  34996 4224  S  0.7 0.6 0:09.89 /usr/bin/opensnitchd -rules-path /etc/opensnitchd/rules
1165 found404 20    0 161M  3532  2744  S  0.7 0.1 0:04.61 /usr/libexec/at-spi2-registryd --use-gnome-session
1185 found404 20    0 462M  33572 19472  S  0.7 0.6 0:08.29 mate-panel
1203 found404 20    0 894M  73132 50552  S  0.7 1.2 0:18.47 /usr/bin/caja
36905 found404 20    0 3717M  707M  198M  S  0.7 12.0 0:22.93 firefox-esr
36974 found404 20    0 3717M  707M  198M  S  0.7 12.0 0:13.87 firefox-esr
36987 found404 20    0 33.2G  195M  79900  S  0.7 3.3 0:06.49 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 2 -
37065 found404 20    0 3033M  397M  146M  S  0.7 6.8 0:22.75 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 4 -
37066 found404 20    0 3033M  397M  146M  S  0.7 6.8 0:02.53 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 4 -
37068 found404 20    0 3033M  397M  146M  S  0.7 6.8 0:08.62 /usr/lib/firefox-esr/firefox-esr -contentproc -childID 4 -
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit

```

3. Para detener procesos en linux de manera rápida y forzada podemos utilizar **kill** proceso por ejemplo **kill kazam** provocará un cierre inmediato.

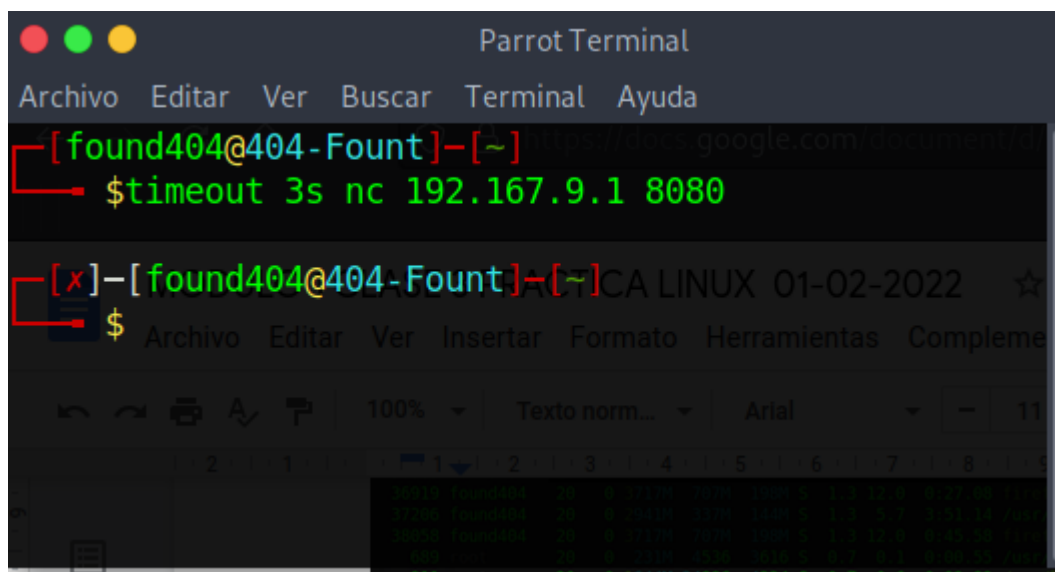
```

Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda

[found404@404-Fount]~$ htop
[x]~[found404@404-Fount]~$ kill kazam
[found404@404-Fount]~$

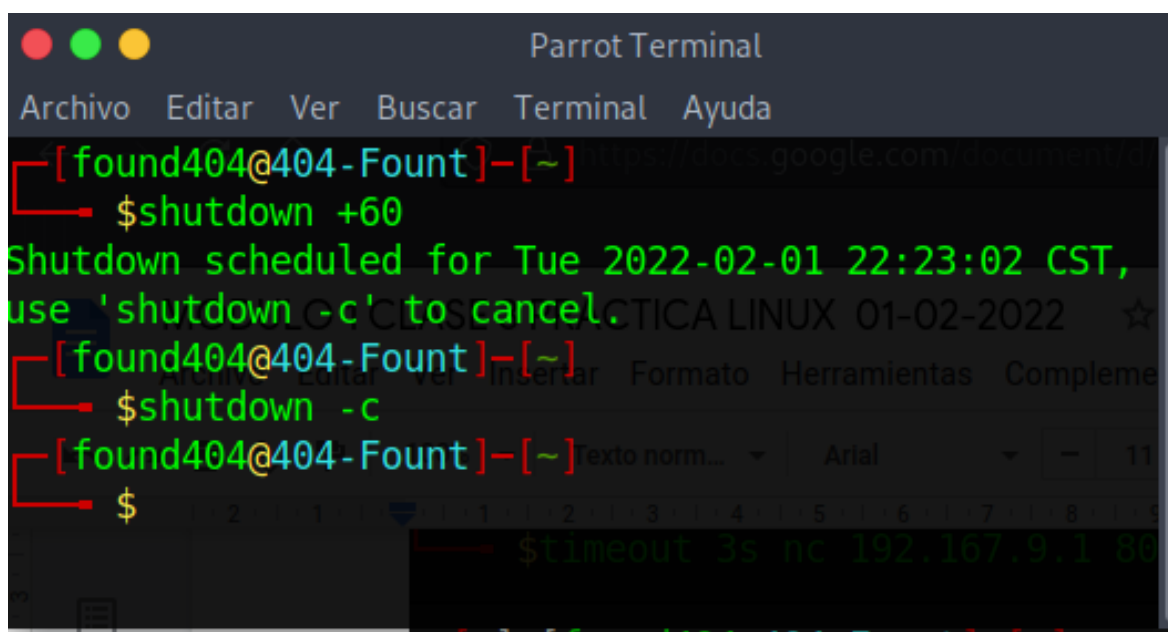
```

4. **timeout** permite correr un comando con un límite de tiempo como se puede mostrar a continuación.



The screenshot shows a Parrot Terminal window with the title 'Parrot Terminal'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal prompt is '[found404@404-Fount]~'. The user has entered the command `$timeout 3s nc 192.167.9.1 8080`. Below the command, there is a line indicating a connection attempt: `[x]-[found404@404-Fount]~`. The terminal also shows a status bar at the bottom with '100%' zoom, 'Texto norm...' font style, 'Arial' font, and a line number '11'.

5. **Shutdown** interrumpe el suministro de energía a los componentes centrales del sistema de forma controlada, los programas de usuario se cierran los programas y protocolos activos se guardan en el disco duro, se guardan las configuraciones de usuario y de más, se puede cancelar con **shutdown -c**.



The screenshot shows a Parrot Terminal window with the title 'Parrot Terminal'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal prompt is '[found404@404-Fount]~'. The user has entered the command `$shutdown +60`. The terminal output shows: `Shutdown scheduled for Tue 2022-02-01 22:23:02 CST, use 'shutdown -c' to cancel.`. Below this, the user has entered the command `$shutdown -c`. The terminal also shows a status bar at the bottom with '100%' zoom, 'Texto norm...' font style, 'Arial' font, and a line number '11'.

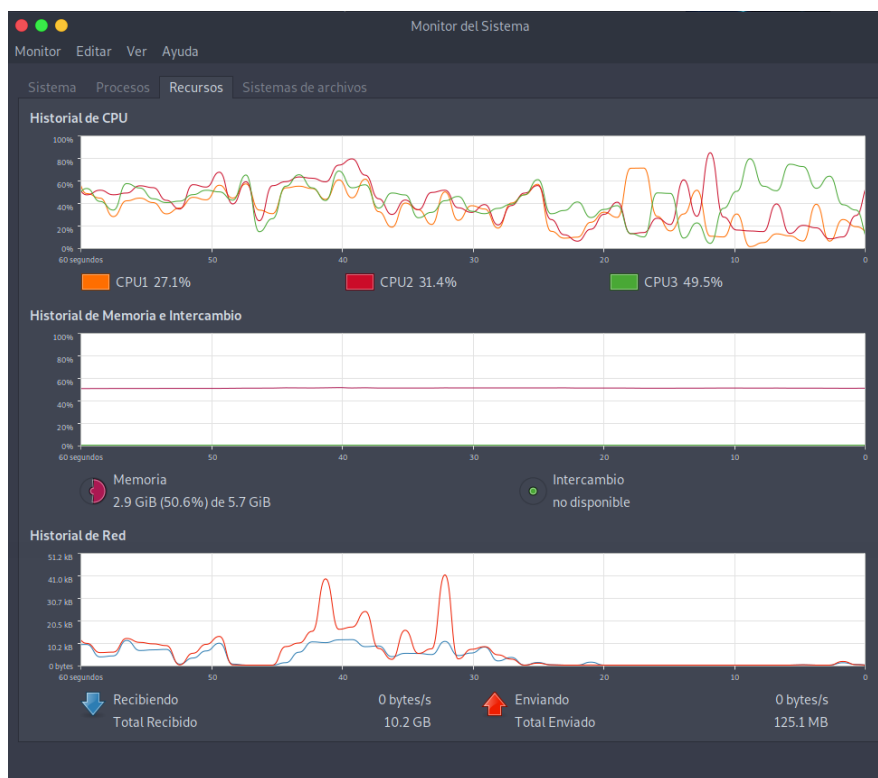
6. **ps** es utilizado para mostrar por pantalla un listado de los procesos que se están ejecutando en el sistema, podemos utilizar múltiples parámetros pero para esta ocasión utilizaremos **ps -aux** para listar procesos de todos los usuarios con información añadida.

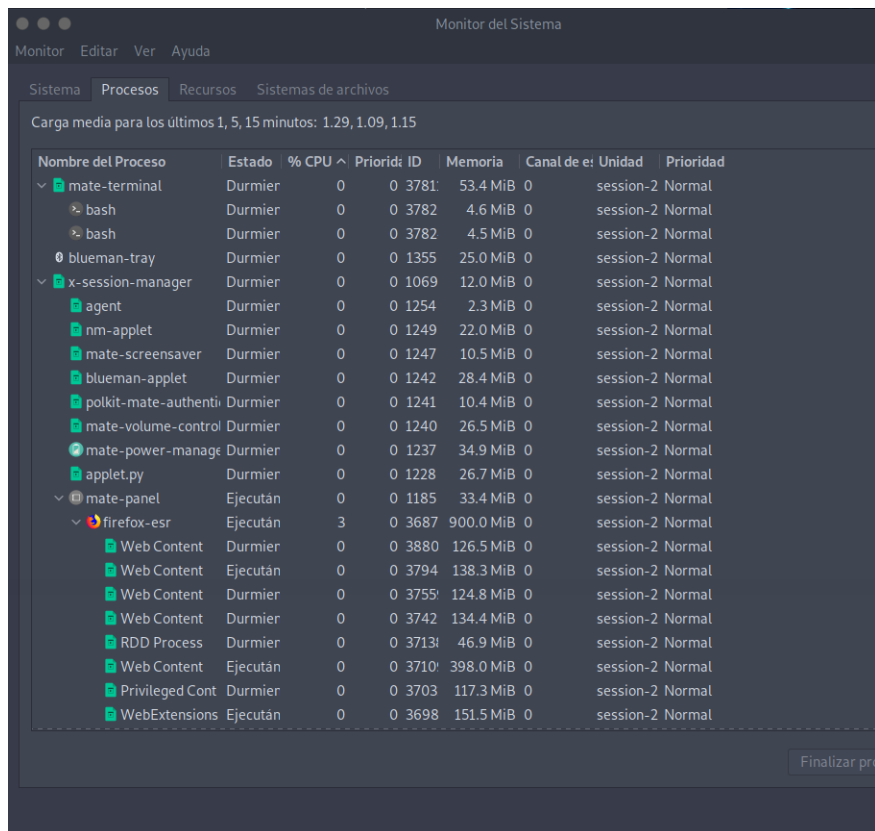
```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root    14400  0.0  0.0 314784 4352 ?    Ssl 17:36  0:00 /usr/sbin/ModemManager
root    14454  0.0  0.0 393656 5600 ?    Ssl 17:36  0:00 /usr/libexec/udisks2/udisksd
root    26050  0.6  0.4 1020784 27924 ?    Ssl 17:44  1:26 /usr/lib/snapd/snapd
found404 28918  0.0  0.0 277384 5156 ?    Sl  18:44  0:00 /usr/libexec/gvfsd-http --spawner :1.
root    33877  0.0  0.0  0 0 ?    I<  19:31  0:00 [iprt-VBoxQueue]
root    33878  0.0  0.0  0 0 ?    S  19:31  0:00 [iprt-VBoxTscThr]
found404 34354  0.0  0.1 385348 11184 ?    Sl  19:35  0:00 /usr/libexec/gvfsd-network --spawner
found404 34368  0.0  0.1 313296 10520 ?    Sl  19:35  0:00 /usr/libexec/gvfsd-dnssd --spawner :1
root    36816  0.0  0.0  0 0 ?    I  20:23  0:02 [kworker/0:0-events]
found404 36874 62.8 12.8 4015232 773488 ?    Sl  20:23 42:34 firefox-esr
found404 36983  1.8  3.9 34029056 237836 ?    Sl  20:24  1:14 /usr/lib/firefox-esr/firefox-esr -con
found404 37031  0.2  2.0 2433924 121156 ?    Sl  20:24  0:08 /usr/lib/firefox-esr/firefox-esr -con
found404 37061 14.4  6.8 3108432 414496 ?    Sl  20:24  9:45 /usr/lib/firefox-esr/firefox-esr -con
found404 37109 11.0  6.7 2889408 408200 ?    Sl  20:24  7:27 /usr/lib/firefox-esr/firefox-esr -con
found404 37138  2.1  0.7 239804 48072 ?    Sl  20:24  1:26 /usr/lib/firefox-esr/firefox-esr -con
found404 37206  7.4  5.2 3010004 317820 ?    Sl  20:26  4:53 /usr/lib/firefox-esr/firefox-esr -con
found404 37429  1.4  2.2 2472012 137664 ?    Sl  20:29  0:53 /usr/lib/firefox-esr/firefox-esr -con
found404 37559  0.3  2.1 2499252 127368 ?    Sl  20:34  0:13 /usr/lib/firefox-esr/firefox-esr -con
found404 37811  0.5  0.8 481484 54200 ?    Sl  20:46  0:16 mate-terminal
found404 37824  0.0  0.0  8036 4616 pts/0  Ss+ 20:46  0:00 bash
found404 37826  0.0  0.0  7896 4708 pts/1  Ss  20:46  0:00 bash
found404 37942  1.5  2.3 2487936 139776 ?    Sl  20:50  0:37 /usr/lib/firefox-esr/firefox-esr -con
root    38422  0.0  0.0  0 0 ?    I  21:02  0:00 [kworker/u6:5-btrfs-endio-write]
found404 38608  0.9  1.5 1127588 96144 ?    Sl  21:10  0:12 /usr/bin/python3 /usr/bin/kazam
root    38688  0.0  0.0  0 0 ?    I  21:12  0:00 [kworker/u6:2-events_unbound]
found404 38803  0.5  2.1 2468396 128360 ?    Sl  21:17  0:04 /usr/lib/firefox-esr/firefox-esr -con
root    38844  0.0  0.0  0 0 ?    I  21:19  0:00 [kworker/1:1-events]
root    38859  0.0  0.0  0 0 ?    I  21:20  0:00 [kworker/2:0-mm_percpu_wq]
root    38923  0.0  0.0  0 0 ?    I  21:23  0:00 [kworker/u6:1-btrfs-endio-write]
root    38946  0.1  0.0  0 0 ?    I  21:24  0:00 [kworker/1:0-events]
root    38948  0.0  0.0  0 0 ?    I  21:24  0:00 [kworker/0:1-events]
root    38977  0.0  0.0  0 0 ?    I  21:25  0:00 [kworker/2:1-ata_sff]
found404 39027  0.0  1.1 2385044 68968 ?    Sl  21:28  0:00 /usr/lib/firefox-esr/firefox-esr -con
root    39078  0.0  0.0  0 0 ?    I  21:30  0:00 [kworker/2:2-ata_sff]
root    39079  0.0  0.0  0 0 ?    I  21:30  0:00 [kworker/u6:0-btrfs-endio-write]
root    39080  0.0  0.0  0 0 ?    I  21:30  0:00 [kworker/u6:3-events_unbound]
found404 39092  0.0  0.0  9888 3580 pts/1  R+  21:31  0:00 ps -aux

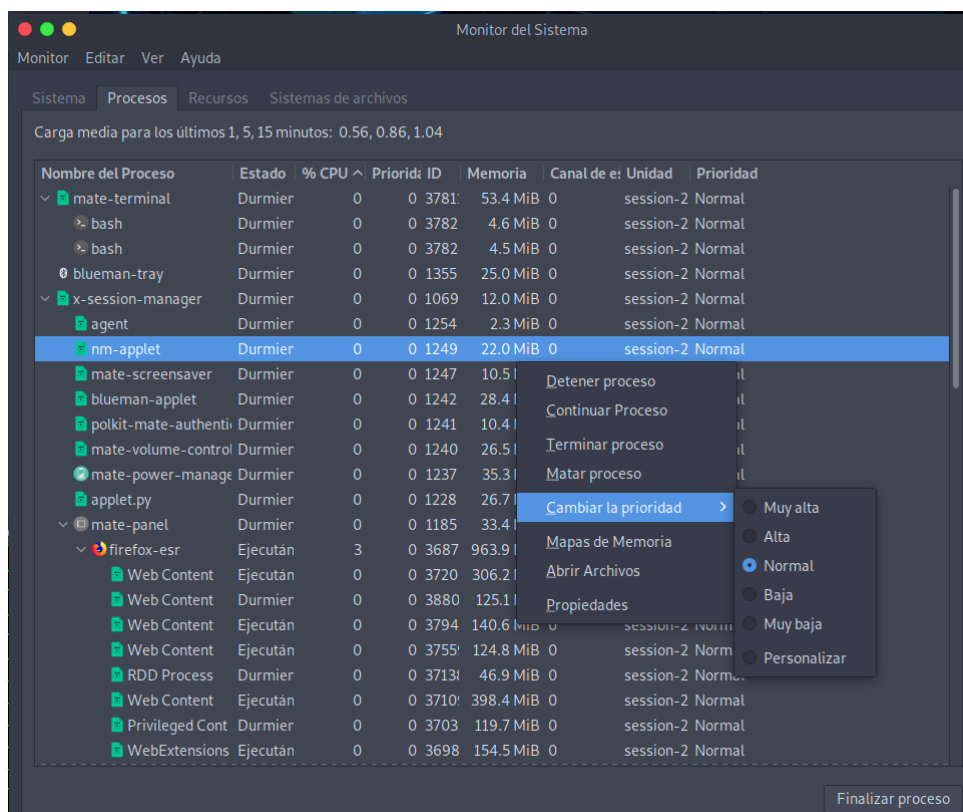
```

7. De igual manera podemos ver las tareas y uso de recursos en Parrot con el monitor del sistema como se muestra en la imagen.





8. De la misma manera es posible cambiar la prioridad del proceso, terminar y demás operaciones.



Manejo de interrupciones

9. Es posible volver a lanzar un procesos con **bg** y **fg** la diferencia es que **bg** lanza el proceso pausado en segundo plano mientras que **fg** lanza el proceso pausado en primer plano monopolizando el terminal.

```

Parrot Terminal
Archivo  Editor  Ver  Buscar  Terminal  Ayuda

/usr/lib/python3/dist-packages/kazam/app.py:145: Warning: value "((GtkIconSize)
32)" of type 'GtkIconSize' is invalid or out of range for property 'icon-size' o
f type 'GtkIconSize'
  self.builder.add_from_file(os.path.join(prefs.datadir, "ui", "kazam.ui"))

(kazam:39651): Gtk-WARNING **: 22:01:16.649: Can't set a parent on widget which
has a parent

(kazam:39651): Gtk-WARNING **: 22:01:16.670: Can't set a parent on widget which
has a parent

^Z
[1]+  Detenido                  kazam
[~]-[found404@404-Fount]-[~]
$bg kazam
[1]+ kazam &
[~]-[found404@404-Fount]-[~]
$fg kazam
kazam
TypeError: position() takes 2 positional arguments but 4 were given
TypeError: position() takes 2 positional arguments but 4 were given

```

Percances:

En realidad no existe mayor dificultad, fue un poco laborioso investigar ciertos comandos que no conocía pero sencillamente son muy intuitivos.

Conclusión:

En esta actividad aprendimos a manejar y entender el funcionamiento y manejo de procesos e interrupciones pero además tuvimos la oportunidad de probar la seguridad del kernel de linux, específicamente de la distribución de Ubuntu utilizando lilo, en base a lo experimentado podemos concluir que si es mucho más robusto que Windows ya que se necesita de un proceso un poco más complejo para modificar el kernel pero además inevitablemente se requieren de los permisos del superusuario y no es para menos que en conceptos de seguridad linux destaque y este por encima de linux e incluso macOS, por mencionar algún dato importante en general linux a sido calificado como el SO mas seguro, ademas es mucho más ligero y consume menos recursos, cuenta con herramientas que facilitan el manejo y gestión de procesos de una manera muy rápida y sencilla sin contar con que estas son gratuitas, cuenta con una amplia comunidad que proporciona herramientas muy útiles y sin ningún costo, como se mostró en la actividad linux cuenta con una amplia cantidad de herramientas que gestionan los recursos y procesos pudiendo incluso controlar aspectos como el grado de importancia que recibe cada proceso, terminarlo, detenerlo y de más, en general Linux facilita muchas cosas con el uso de la terminal, algo que a llegado a considerarse una desventaja debido a que se tiene la idea de que es muy compleja, nada más lejos de la realidad,

Bibliografía

- (S/f). Mit.edu. Recuperado el 2 de febrero de 2022, de
<http://web.mit.edu/rhel-doc/3/rhel-rg-es-3/s1-grub-lilo.html>
- Gestores de arranque. (s/f). Ediciones-Eni.Com. Recuperado el 2 de febrero de 2022, de
<https://www.ediciones-eni.com/open/mediabook.aspx?idR=f89a27115bb2af39a0f4040851e6b2f3>
- OpenWebinars.net. 2022. *20 comandos para administrar y gestionar procesos Linux*. [online] Available at:
<https://openwebinars.net/blog/20-comandos-para-administrar-y-gestionar--facilmente-los-procesos-linux>
- Linuxito. 2022. *Ejecutar un comando con timeout*. [online] Available at:
<https://www.linuxito.com/gnu-linux/nivel-basico/1327-ejecutar-un-comando-con-timeout>
- Linux, C., 2022. *Linux: comandos shutdown para apagar y reiniciar el sistema*. [online] IONOS Digitalguide. Available at:
<https://www.ionos.mx/digitalguide/servidores/configuracion/comandos-shutdown-de-linux>
- # rm-rf.es | Blog de un sysadmin Unix, GNU/Linux, Windows y lo que haga falta... 2022. *El comando PS: Listar y manejar procesos* | *rm-rf.es*. [online] Available at:
<https://rm-rf.es/el-comando-ps-listar-y-manejar-procesos>
- Solvetic. 2022. *Cómo abrir administrador de tareas similar para Linux*. [online] Available at:
<https://www.solvetic.com/tutoriales/article/4115-como-abrir-administrador-tareas-similar-linux>
- Bonaval Multimedia. 2022. *Manejar procesos en linux en terminal (comandos bg y fg: background y foreground)*. [online] Available at:
<https://www.bonaval.com/kb/sistemas-operativos/linux-sistemas-operativos/manejar-procesos-en-linux-en-terminal-comandos-bg-y-fg-background-y-foreground>