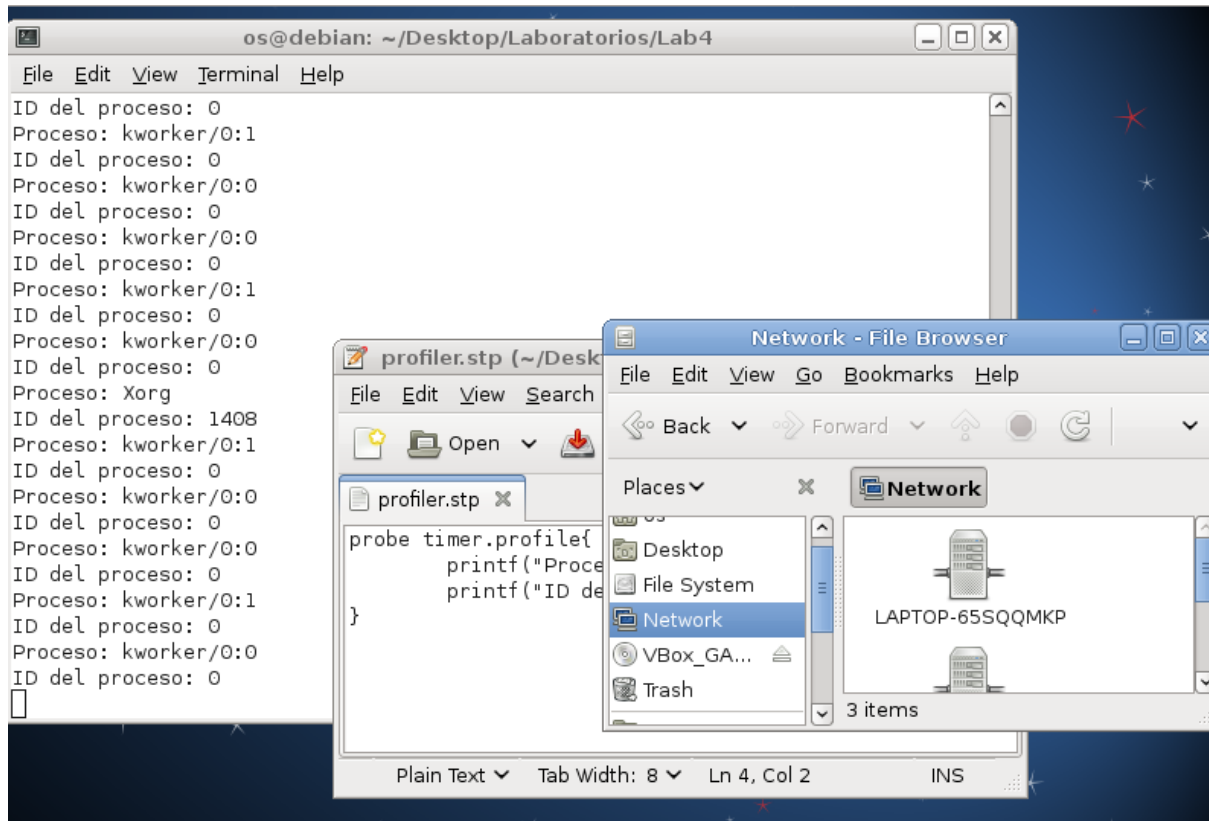
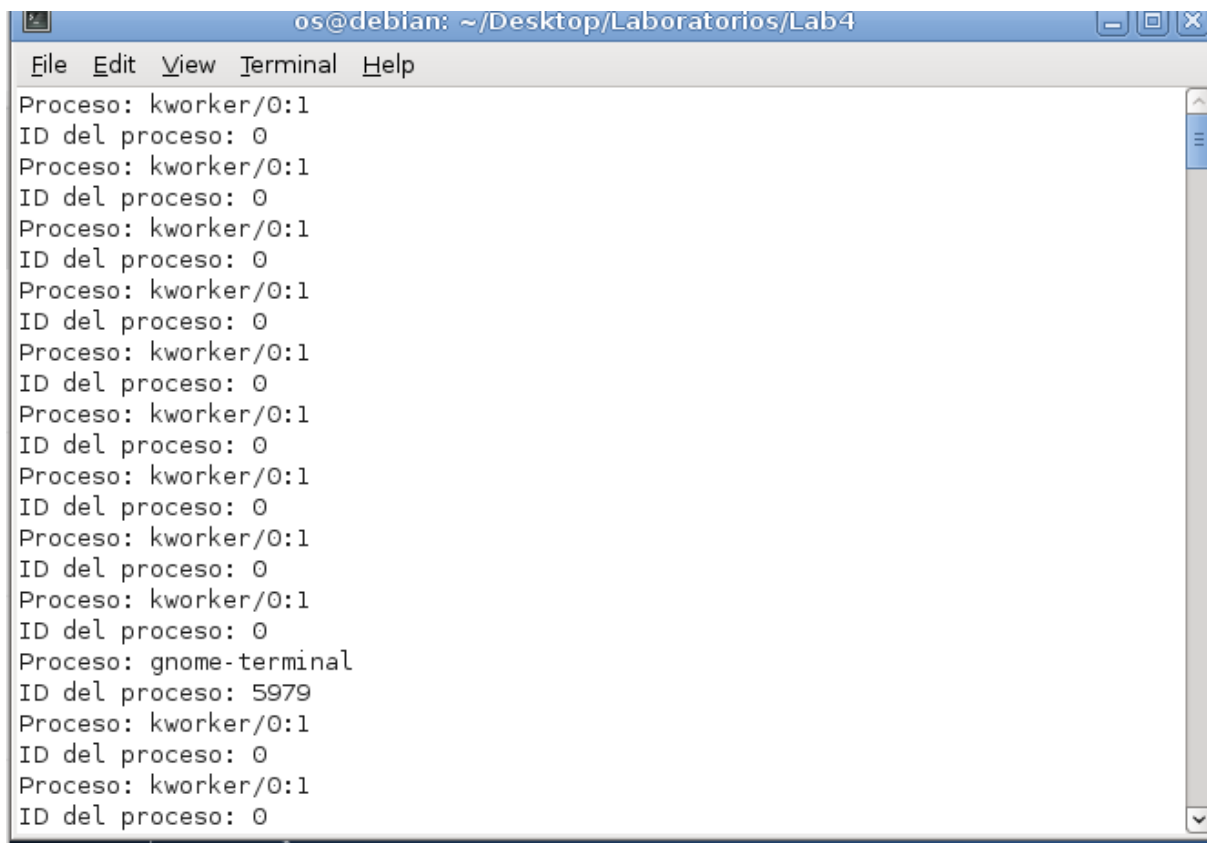


## Laboratorio 4

### Ejercicio 1





**Figura 1:** Output del profiler.stp en el OS debian.

### ¿Qué puede ver en el output cuando realiza estas acciones?

- Podemos ver que el output muestra el proceso ID con el kernel sobre las aplicaciones que el usuario está iniciando/acabando a través del SystemTap. Con esta información, podemos detectar que evento/aplicación del SO se está utilizando y se ejecuta un controlador para dicho programa.

## ¿Para qué sirve SystemTap?

- SystemTap proporciona una interfaz de línea de comandos simple y un lenguaje de scripting para escribir instrumentación para un kernel en ejecución más aplicaciones de espacio de usuario. Además, ayuda a los desarrolladores a no tener que reiniciar o recompilar el sistema operativo para obtener datos respecto a los eventos que se necesitan debuggear o conocer el estado.

## ¿Qué es un probe?

- Básicamente los probes, en el Systemtap, son los identificadores para un evento y su controlador correspondiente.

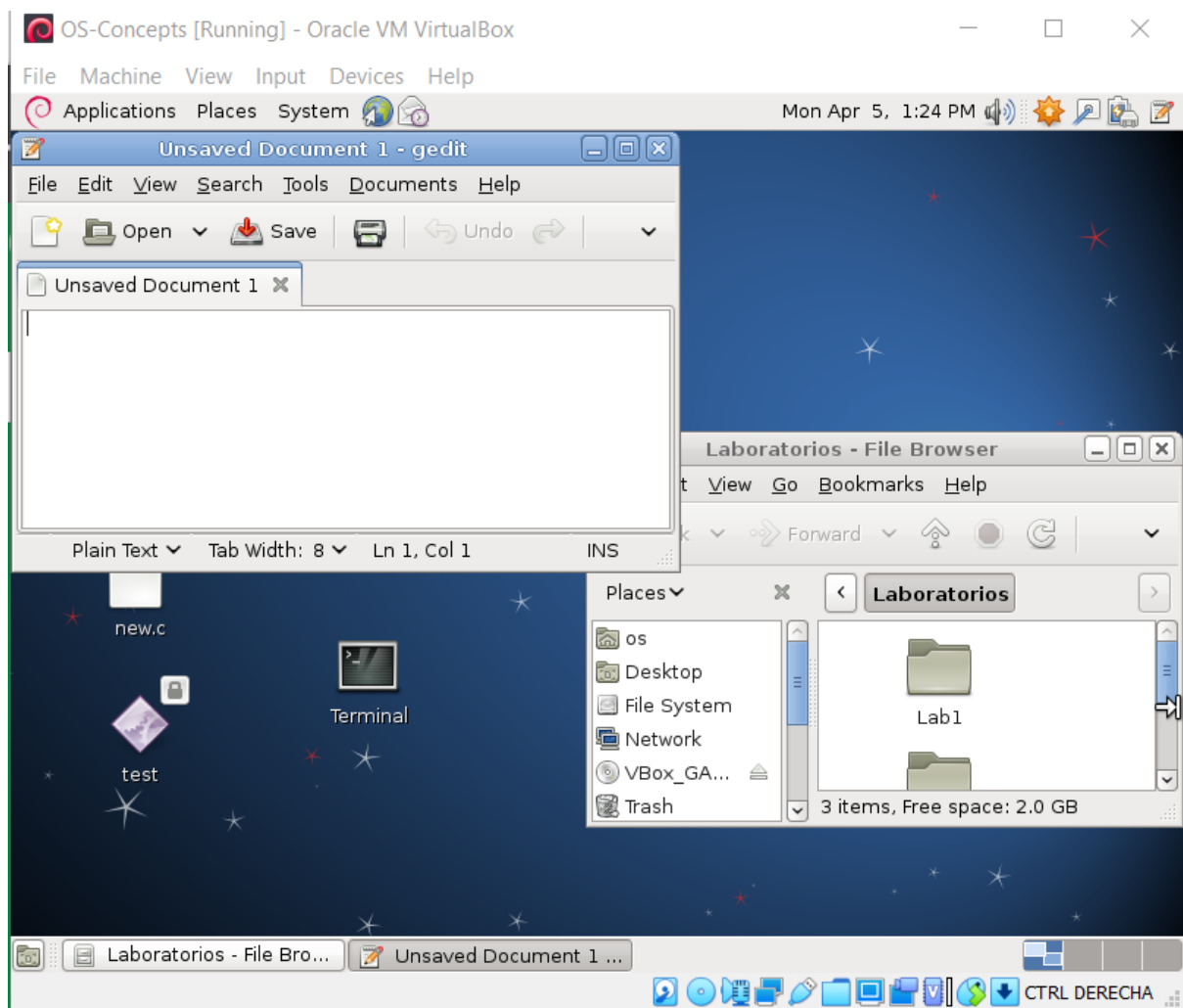
## ¿Cómo funciona SystemTap?

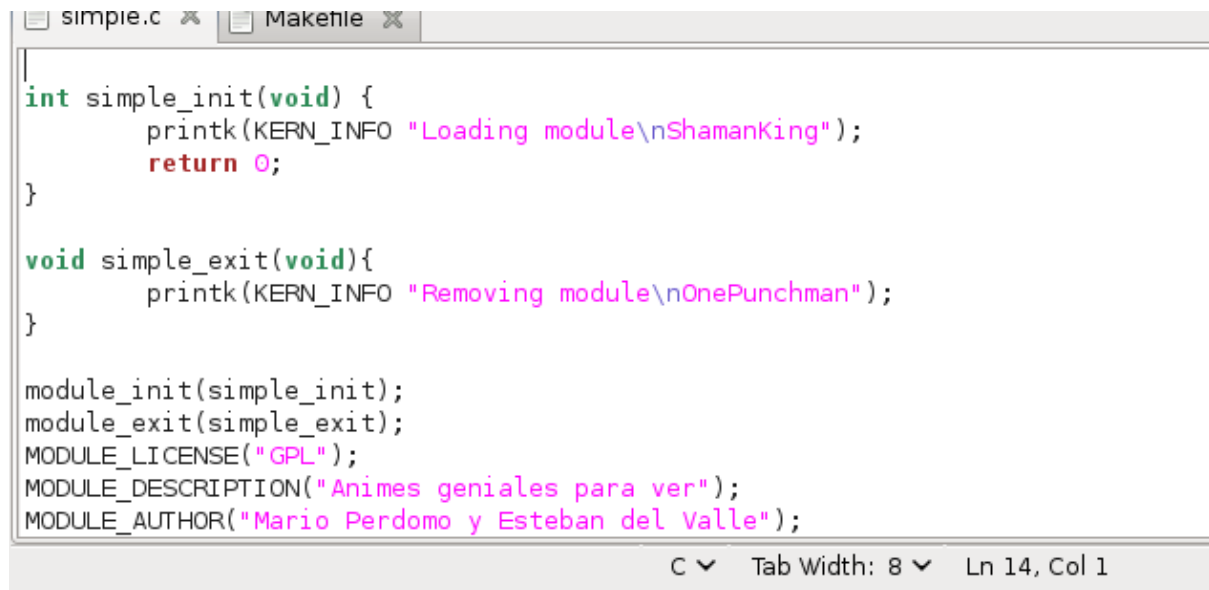
○ Este paquete permite a los usuarios escribir y utilizar scripts simples para examinar en profundidad las actividades de un sistema Linux en ejecución. Lo proporciona a través de una interfaz de línea de comandos simple y un lenguaje de secuencias de comandos para escribir instrumentación para un kernel en ejecución más aplicaciones de espacio de usuario

### ¿Qué es hacer profiling y qué tipo de profiling se hace en este ejercicio?

○ *Profiling* es un análisis dinámico de programas que mide espacio (memoria), la complejidad temporal de un programa, el uso de instrucciones particulares o la frecuencia y duración de la llamada a la función. En este caso, el tipo de profiling es el de “Event-based Profiler”, ya que la arquitectura de Systemtap trabaja al extraer datos del contexto del evento, almacenarlos en variables internas e imprimir resultados.

### Ejercicio 2





```
int simple_init(void) {
    printk(KERN_INFO "Loading module\nShamankKing");
    return 0;
}

void simple_exit(void){
    printk(KERN_INFO "Removing module\nOnePunchman");
}

module_init(simple_init);
module_exit(simple_exit);
MODULE_LICENSE("GPL");
MODULE_DESCRIPTION("Animes geniales para ver");
MODULE_AUTHOR("Mario Perdomo y Esteban del Valle");
```

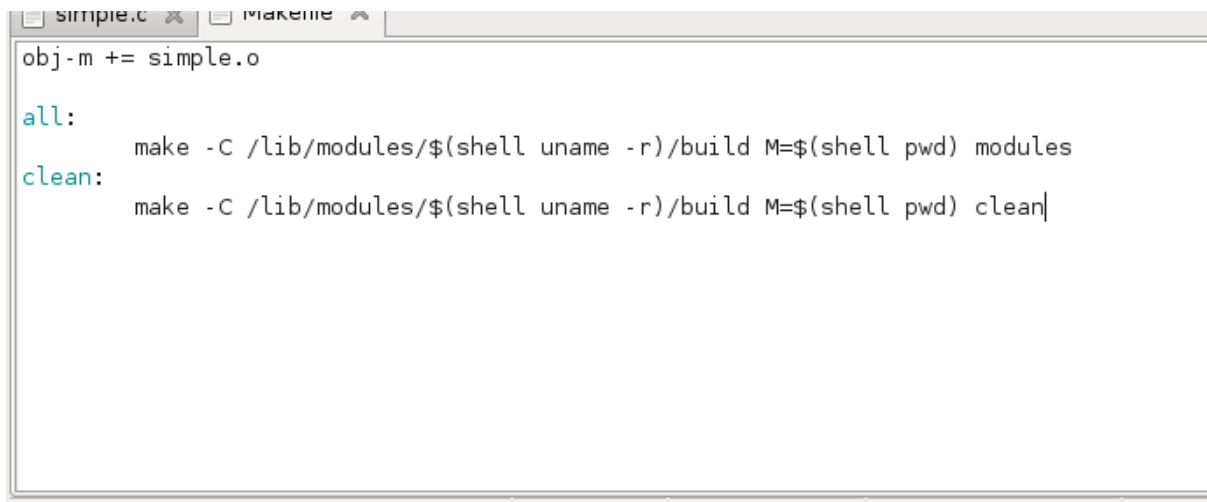
**Figura 2:** Output del simple.c en el OSC-2016.

**¿Cuál es la diferencia en C entre un método que no recibe parámetros y uno que recibe void?**

- Cuando una función no tiene argumentos, no recibe ningún dato de la función que la llama. De manera similar, cuando no devuelve un valor, la función que llama no recibe ningún dato de la función llamada. Esta diferencia es la posibilidad de pasar parámetros a una función con una lista vacía; el compilador no se quejara. Ese no es el caso de una función con un parámetro void.

**¿Qué diferencia hay entre printk y printf?**

- Es la especificación del nivel de un registro. printk () es una función de nivel de kernel, que tiene la capacidad de imprimir en diferentes niveles de registro, y el kernel usa el nivel de registro para decidir si imprime el mensaje en la consola.

A screenshot of a text editor window with two tabs: 'simple.c' and 'Makefile'. The 'Makefile' tab is active, showing the following content:

```
obj-m += simple.o

all:
    make -C /lib/modules/$(shell uname -r)/build M=$(shell pwd) modules

clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(shell pwd) clean
```

### ¿Qué es y para qué sirve KERN\_INFO?

- Es el nivel de registro como identificador sobre las acciones realizadas por el kernel.

### ¿Qué es una goal definition o definición de meta en un Makefile, y qué se está haciendo con la definición de meta obj-m?

- Una *goal definition* en los Makefiles es compilar cualquier archivo que deba compilarse, en función de los archivos que hayan cambiado. Cuando se ejecuta el programa, se utiliza la versión más reciente del archivo.
- obj-m guarda los comandos de compilación con todos sus parámetros del programa simple, con el fin de encontrar librerías, ficheros de cabecera (.h). Gracias a esto, no se tendría que escribir grandes líneas de código para compilar.

### ¿Qué función tienen las líneas all: y clean:?

- All: y Clean: son comandos de punteros, o mejor dicho, son *targets* dentro del programa Makefile. Clean: se usa a menudo como un objetivo que elimina la salida de otros objetivos, mientras que All: busca a todos los programas o archivos que necesita para ejecutarse dicha regla escrita en el Makefile con el kernel.

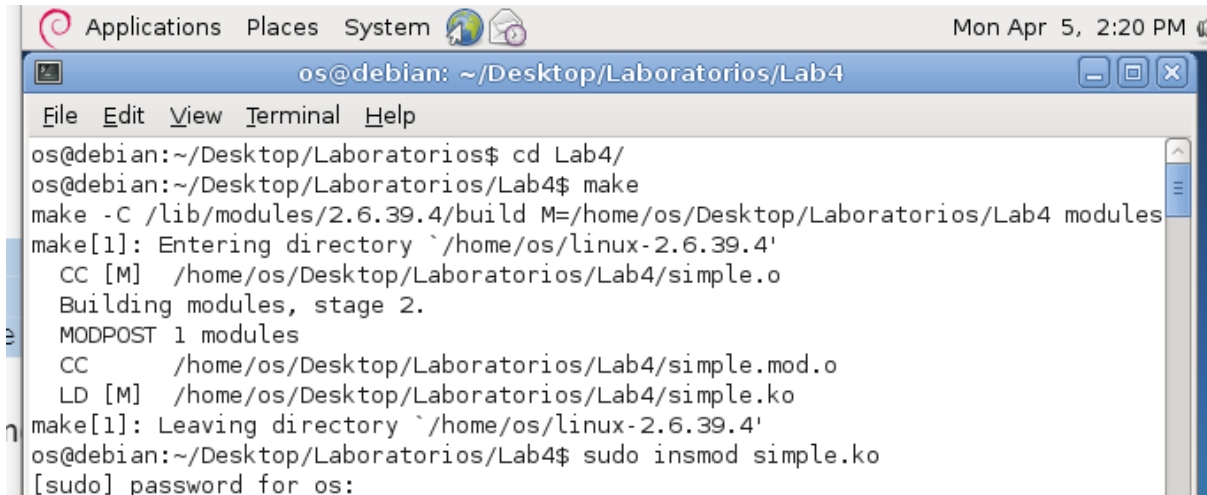
### ¿Qué hace la opción -C en este Makefile?

- Cambia al directorio dir antes de leer los archivos MAKE. Esto se usa típicamente con invocaciones recursivas de make.

### ¿Qué hace la opción M en este Makefile?

- Permite que make vuelva a su directorio actual después de procesar Makefile, ya que M es una variable asignada a la ejecución de make.

**Tome una captura de pantalla de los resultados de ambos comandos e inclúyala en sus entregables.**



```

os@debian: ~/Desktop/Laboratorios/Lab4
File Edit View Terminal Help
os@debian:~/Desktop/Laboratorios$ cd Lab4/
os@debian:~/Desktop/Laboratorios/Lab4$ make
make -C /lib/modules/2.6.39.4/build M=/home/os/Desktop/Laboratorios/Lab4 modules
make[1]: Entering directory `/home/os/linux-2.6.39.4'
  CC [M]  /home/os/Desktop/Laboratorios/Lab4/simple.o
Building modules, stage 2.
  MODPOST 1 modules
  CC      /home/os/Desktop/Laboratorios/Lab4/simple.mod.o
  LD [M]  /home/os/Desktop/Laboratorios/Lab4/simple.ko
make[1]: Leaving directory `/home/os/linux-2.6.39.4'
os@debian:~/Desktop/Laboratorios/Lab4$ sudo insmod simple.ko
[sudo] password for os:

```

```

3/sbin/VBoxService
16.220224] 19:23:03.147132 main      Process ID: 1589
16.220225] 19:23:03.147133 main      Package type: LINUX_32BITS_GENERIC
16.220837] 19:23:03.147744 main      6.1.18 r142142 started. Verbose level =

16.221554] 19:23:03.148450 main      vbglR3GuestCtrlDetectPeekGetCancelSuppc
: Supported (#1)
16.228816] 19:23:03.155687 automount Error: vbsvcAutomounterMountIt: Failec
mount 'debian_lab' on '/media/sf_debian_lab': No such device (-1,19)
16.231399] 19:23:03.158239 automount Error: vbsvcAutomounterMountIt: Failec
mount 'sharefolder' on '/media/sf_sharefolder': No such device (-1,19)
17.234631] 19:23:04.161488 automount Error: vbsvcAutomounterMountIt: Failec
mount 'debian_lab' on '/media/sf_debian_lab': No such device (-1,19)
17.236278] 19:23:04.163118 automount Error: vbsvcAutomounterMountIt: Failec
mount 'sharefolder' on '/media/sf_sharefolder': No such device (-1,19)
23.181860] ISO 9660 Extensions: Microsoft Joliet Level 3
23.183233] ISO 9660 Extensions: RRIP_1991A
24.688799] eth1: no IPv6 routers present
3428.005536] Loading module
3428.005538] Shamanking

```

### ¿Para qué sirve dmesg?

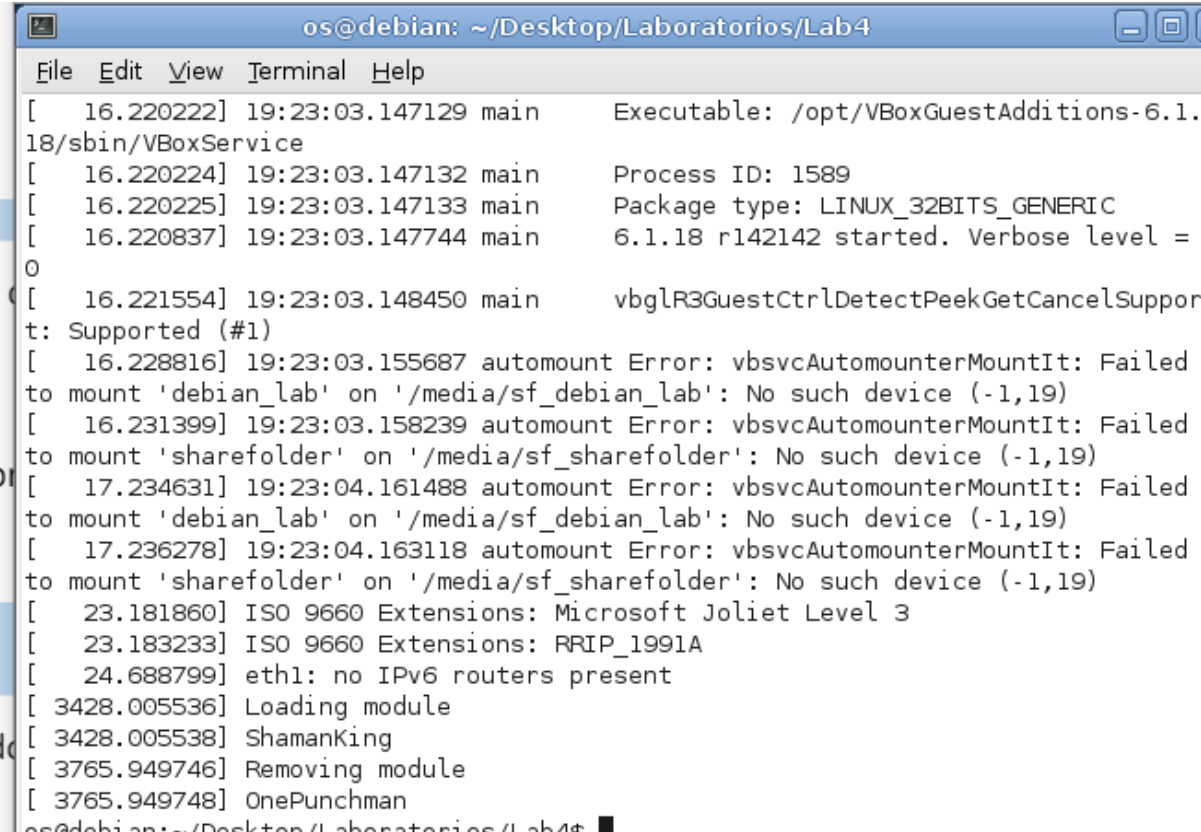
- dmesg (mensaje de diagnóstico) es un comando en la mayoría de los sistemas operativos que imprime el búfer de mensajes del kernel. La salida incluye mensajes producidos por los controladores del dispositivo. En este caso, mostró el buffer del kernel.

### ¿Qué hace la función `simple_init` en su programa `simple.c`?

- Esta función creada en `simple.c` llama el módulo cuando se cargando.

**Tome una nueva captura de pantalla de los resultados de ambos comandos e inclúyala en sus entregables.**

- 



The screenshot shows a terminal window titled "os@debian: ~/Desktop/Laboratorios/Lab4". The terminal displays a series of log messages from the `VBoxService` process. The messages include timestamps, process IDs, and details about the system's state, such as the package type (`LINUX_32BITS_GENERIC`), the verbose level (`0`), and the detection of supported features (`vbglR3GuestCtrlDetectPeekGetCancelSupport: Supported (#1)`). There are also several error messages indicating failed mount attempts for `'debian_lab'` and `'sharefolder'` on `"/media/sf_debian_lab"` and `"/media/sf_sharefolder"` respectively, with the error code `(-1,19)`. The terminal also shows the loading and removing of modules `ShamanKing` and `OnePunchman`, and the detection of IPv6 routers on the `eth1` interface.

```
os@debian: ~/Desktop/Laboratorios/Lab4
[ 16.220222] 19:23:03.147129 main      Executable: /opt/VBoxGuestAdditions-6.1.18/sbin/VBoxService
[ 16.220224] 19:23:03.147132 main      Process ID: 1589
[ 16.220225] 19:23:03.147133 main      Package type: LINUX_32BITS_GENERIC
[ 16.220837] 19:23:03.147744 main      6.1.18 r142142 started. Verbose level = 0
[ 16.221554] 19:23:03.148450 main      vbglR3GuestCtrlDetectPeekGetCancelSupport: Supported (#1)
[ 16.228816] 19:23:03.155687 automount Error: vbsvcAutomounterMountIt: Failed to mount 'debian_lab' on '/media/sf_debian_lab': No such device (-1,19)
[ 16.231399] 19:23:03.158239 automount Error: vbsvcAutomounterMountIt: Failed to mount 'sharefolder' on '/media/sf_sharefolder': No such device (-1,19)
[ 17.234631] 19:23:04.161488 automount Error: vbsvcAutomounterMountIt: Failed to mount 'debian_lab' on '/media/sf_debian_lab': No such device (-1,19)
[ 17.236278] 19:23:04.163118 automount Error: vbsvcAutomounterMountIt: Failed to mount 'sharefolder' on '/media/sf_sharefolder': No such device (-1,19)
[ 23.181860] ISO 9660 Extensions: Microsoft Joliet Level 3
[ 23.183233] ISO 9660 Extensions: RRIP_1991A
[ 24.688799] eth1: no IPv6 routers present
[ 3428.005536] Loading module
[ 3428.005538] ShamanKing
[ 3765.949746] Removing module
[ 3765.949748] OnePunchman
os@debian:~/Desktop/Laboratorios/Lab4$
```

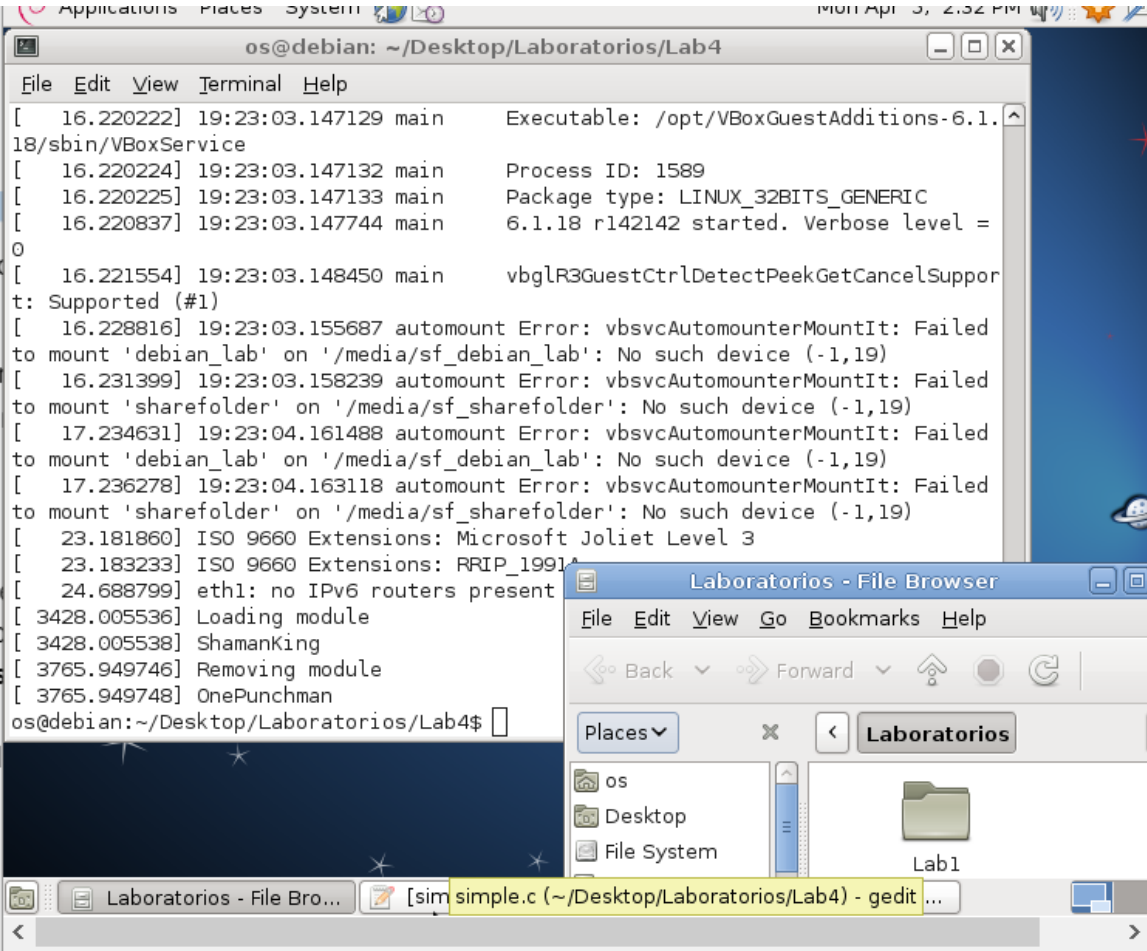
### ¿Qué hace la función `simple_exit` en su programa `simple.c`?

- Remueve el módulo del buffer del kernel, a lo que tengo entendido, y deja un mensaje, lo cual son los que personalizamos.

**Usted ha logrado crear, cargar y descargar un módulo de Linux. ¿Qué poder otorga el ejecutar código de esta forma?**

- La ventaja de no necesitar compilar todo el kernel para realizar cambios en un módulo. Por lo tanto, esto ahorra tiempo y nos evita tener problemas al cargar nuestro núcleo base. Otra ventaja es que nos ayuda a ahorrar memoria ya que solo los cargamos cuando necesitamos usarlos.

### Ejercicio 3



The screenshot shows a Linux desktop environment. In the foreground, there is a terminal window titled 'os@debian: ~/Desktop/Laboratorios/Lab4'. The terminal displays a series of log messages from the VboxGuestAdditions-6.1.18/sbin/VBoxService process. The messages include timestamps, process IDs, and details about the automounter's failure to mount 'debian\_lab' and 'sharefolder' on the specified media paths. The terminal also shows the loading and removing of modules like 'ShamanKing' and 'OnePunchman'. In the background, there is a file browser window titled 'Laboratorios - File Browser' showing the contents of the 'Laboratorios' directory, which includes a folder named 'Lab1'. The desktop background is a blue gradient with a small icon in the bottom right corner.

```
os@debian: ~/Desktop/Laboratorios/Lab4
[ 16.220222] 19:23:03.147129 main Executable: /opt/VBoxGuestAdditions-6.1.18/sbin/VBoxService
[ 16.220224] 19:23:03.147132 main Process ID: 1589
[ 16.220225] 19:23:03.147133 main Package type: LINUX_32BITS_GENERIC
[ 16.220837] 19:23:03.147744 main 6.1.18 r142142 started. Verbose level = 0
[ 16.221554] 19:23:03.148450 main vbglR3GuestCtrlDetectPeekGetCancelSupport: Supported (#1)
[ 16.228816] 19:23:03.155687 automount Error: vbsvcAutomounterMountIt: Failed to mount 'debian_lab' on '/media/sf_debian_lab': No such device (-1,19)
[ 16.231399] 19:23:03.158239 automount Error: vbsvcAutomounterMountIt: Failed to mount 'sharefolder' on '/media/sf_sharefolder': No such device (-1,19)
[ 17.234631] 19:23:04.161488 automount Error: vbsvcAutomounterMountIt: Failed to mount 'debian_lab' on '/media/sf_debian_lab': No such device (-1,19)
[ 17.236278] 19:23:04.163118 automount Error: vbsvcAutomounterMountIt: Failed to mount 'sharefolder' on '/media/sf_sharefolder': No such device (-1,19)
[ 23.181860] ISO 9660 Extensions: Microsoft Joliet Level 3
[ 23.183233] ISO 9660 Extensions: RRIP_1991A
[ 24.688799] eth1: no IPv6 routers present
[ 3428.005536] Loading module
[ 3428.005538] ShamanKing
[ 3765.949746] Removing module
[ 3765.949748] OnePunchman
os@debian:~/Desktop/Laboratorios/Lab4$
```

C.



```
os@debian: /dev/disk/by-id
File Edit View Terminal Help
bin  etc      lib      mnt      root      srv      usr
boot home     lost+found opt     /sbin     sys      var
dev  initrd.img media    proc     selinux  tmp      vmlinuz
os@debian:/$ cd dev/disk/by-id
os@debian:/dev/disk/by-id$ ls -Al
total 0
lrwxrwxrwx 1 root root 9 Apr 5 16:55 ata-VBOX_CD-ROM_VB2-01700376 -> ../../hdc
lrwxrwxrwx 1 root root 9 Apr 5 16:50 ata-VBOX_HARDDISK_VB214e22a0-17744117 ->
../../sda
lrwxrwxrwx 1 root root 10 Apr 5 16:50 ata-VBOX_HARDDISK_VB214e22a0-17744117-part1 ->
../../sda1
lrwxrwxrwx 1 root root 10 Apr 5 16:50 ata-VBOX_HARDDISK_VB214e22a0-17744117-part2 ->
../../sda2
lrwxrwxrwx 1 root root 10 Apr 5 16:50 ata-VBOX_HARDDISK_VB214e22a0-17744117-part5 ->
../../sda5
lrwxrwxrwx 1 root root 9 Apr 5 16:50 scsi-SATA_VBOX_HARDDISK_VB214e22a0-17744117 ->
../../sda
lrwxrwxrwx 1 root root 10 Apr 5 16:50 scsi-SATA_VBOX_HARDDISK_VB214e22a0-17744117-part1 ->
../../sda1
lrwxrwxrwx 1 root root 10 Apr 5 16:50 scsi-SATA_VBOX_HARDDISK_VB214e22a0-17744117-part2 ->
../../sda2
lrwxrwxrwx 1 root root 10 Apr 5 16:50 scsi-SATA_VBOX_HARDDISK_VB214e22a0-17744117-part5 ->
../../sda5
os@debian:/dev/disk/by-id$
```

Anote el contenido de la columna<file system>.

O

```
fstab [Read-Only] (/etc) - gedit
File Edit View Search Tools Documents Help
*Unsaved Document 1 x fstab x
# /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>
proc /proc proc defaults 0 0
# / was on /dev/sda1 during installation
UUID=c6c1c4e3-4e42-4b7a-89e2-69140f21f585 / ext3 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=e2976f5e-f45d-4d31-8a8f-ac19d4f04620 none swap sw 0 0
/dev/scd0 /media/cdrom0 udf,iso9660 user,noauto 0 0

Plain Text Tab Width: 8 Ln 6, Col 2 INS
os@debian: /etc etc - File Browser fstab [Read-Only] (/etc... Right Ctrl
```

### **¿Qué es y para qué sirve el archivo fstab?**

○ fstab (File System table) compila información sobre sistemas de archivos y es usado al bootear el OS. Al invocar mount, se utiliza. En file system se encuentra en que participación, en el mount point está el punto de montaje, en type está el tipo de sistema de archivos, en options van las opciones para el montaje de filesystem, en dump está la indicación a dump para saber si hacer un backup del filesystem y pass es una indicación para el fsck.

### **¿Qué almacena el directorio /etc?**

○ Almacena archivos de configuración del sistema operativo. Contiene solo ficheros y no binarios

### **¿En Windows, quién (hasta cierto punto) funge como /etc?**

○ System32 podría ser considerado el equivalente a etc

### **¿Qué se almacena en /dev y en /dev/disk?**

○ En dev se almacenan archivos especiales o de dispositivos. Aquí se guardan diferentes particiones del sistema y tiene todos los dispositivos de almacenamiento que estén conectados y se entiendan con un volumen lógico como usbs o CD-ROMs. En dev/disk está toda la información del disco duro conocido por el sistema operativo.

### **¿Por qué se usa root < la dirección completa del link hacia sda>en lugar de sólo /dev/sda, y cuál es el papel que el programa udev cumple en todo esto?**

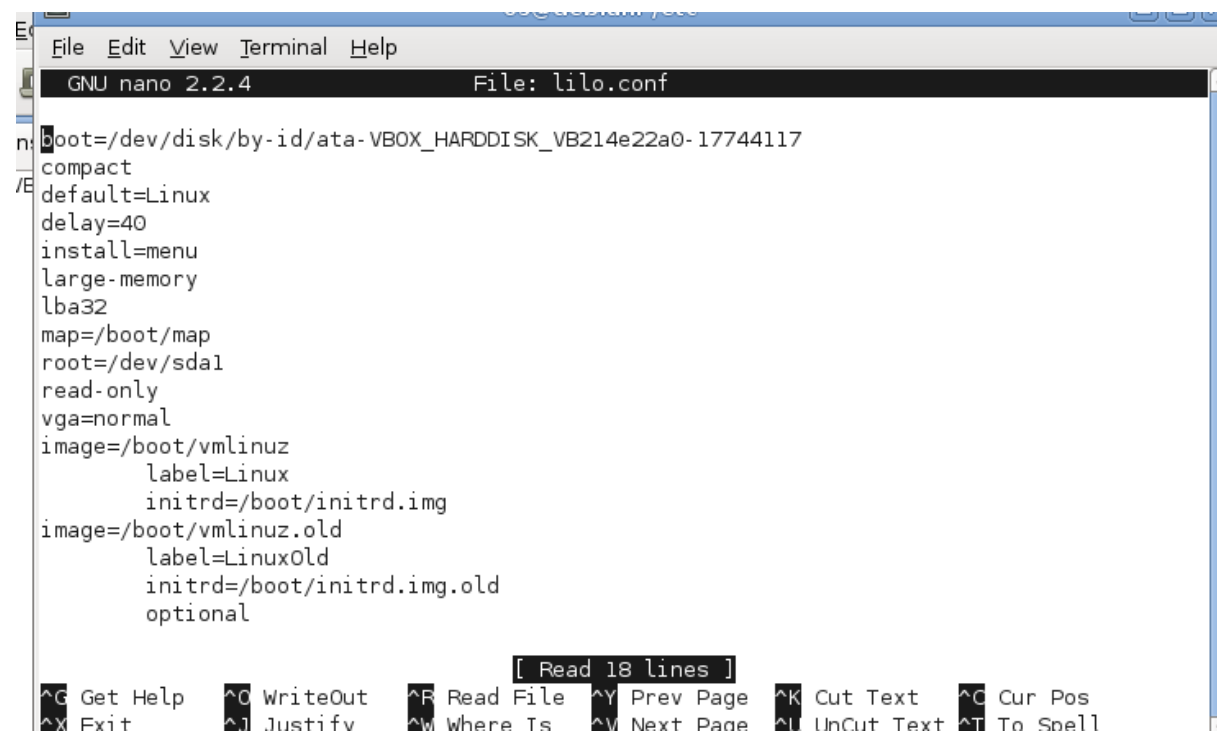
○ Si se usa solamente el sda, puede variar al iniciar el sistema de diferentes maneras. Esto podría liderar a fallos. Al usar la dirección específica se vuelve más seguro. El papel que cumple el programa udev en esto es proporcionar el software eventos del dispositivos. Este administra permisos de nodos del dispositivo y crea enlaces en el directorio dev o cambiar nombre de interfaces red.

### **¿Qué es un block device y qué significado tiene sdxN, donde x es una letra y N es un número, en direcciones como /dev/sdb? Investigue y explique los conceptos de Master Boot Record(MBR) y Volume Boot Record (VBR), y su relación con UEFI.**

○ block device es un archivo que referencia a un dispositivo y da acceso al mismo. El significado que tiene sdxN en direcciones como dev/sdb es para hacer referencia al directorio del disco que se quiera especificar. la x especifica el disco al cual se quiere acceder y N es la partición del disco. En el BIOS se lee el primer sector del disco que es el MBR. Este tiene un tamaño limitado y pasa al bootloader. El bootloader encuentra la primera partición de arranque que sería el VBR. El bootloader gestiona las opciones del menu y carga el sistema operativo. El mecanismo UEFI tiene un proceso de arranque diferente. Este no tiene que recuperar código. Lo encuentra en ficheros con formato PE.

### ¿Qué es hacer chain loading?

○ Chain loading es cuando se le indica al GRUB leer +n sectores del inicio de la partición. De esta manera se permite una carga en cadena del sistema y las particiones en un orden apropiado.



```
File Edit View Terminal Help
GNU nano 2.2.4 File: lilo.conf
boot=/dev/disk/by-id/ata-VBOX_HARDDISK_VB214e22a0-17744117
compact
default=Linux
delay=40
install=menu
large-memory
lba32
map=/boot/map
root=/dev/sda1
read-only
vga=normal
image=/boot/vmlinuz
    label=Linux
    initrd=/boot/initrd.img
image=/boot/vmlinuz.old
    label=LinuxOld
    initrd=/boot/initrd.img.old
optional

[ Read 18 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^N Next Page ^U UnCut Text ^T To Spell
```

### ¿Qué se está indicando con la configuración root="<el file system anotado>"?

○ La ubicación del archivo con instrucciones para bootear el sistema, el gestor de arranque estático y el programa núcleo y archivos de configuración para iniciar linux.

```

root@debian:/# ls
bin  dev  home  lib      media  opt  root  selinux  sys  usr  vmlinuz
boot etc  initrd.img lost+found mnt    proc  sbin  srv      tmp  var
root@debian:/# cd vmlinuz
bash: cd: vmlinuz: Not a directory
root@debian:/# rm vmlinuz
root@debian:/# rm initrd.img
root@debian:/# ls
bin  dev  home  lost+found  mnt  proc  sbin  srv  tmp  var
boot etc  lib  media      opt  root  selinux  sys  usr
root@debian:/# ls
bin  dev  home  lost+found  mnt  proc  sbin  srv  tmp  var
boot etc  lib  media      opt  root  selinux  sys  usr
root@debian:/# █

```

## ¿Qué es vmlinuz?

- Es la imagen comprimida del núcleo o kernel. Se descomprime, carga a memoria y se ejecuta al iniciar el sistema.

```

[1]+  Stopped                  man ln
root@debian:/boot# ln -s vmlinuz-2.6.39.4 vmlinuz
root@debian:/boot# ln -s initrd.img-2.6.39.4 initrd.img
root@debian:/boot# sudo rm -r /boot/grub
root@debian:/boot# ls
coffee.bmp      initrd.img      System.map-2.6.32-5-686
config-2.6.32-5-686  initrd.img-2.6.32-5-686  System.map-2.6.39.4
config-2.6.39.4    initrd.img-2.6.39.4    vmlinuz
debian.bmp        sarge.bmp      vmlinuz-2.6.32-5-686
debianlilo.bmp    sid.bmp        vmlinuz-2.6.39.4
root@debian:/boot# █

```

**Cree una nueva snapshot de su máquina virtual y luego use esta y la snapshot anterior para tomar fotos del proceso de booteo, evidenciando el empleo de GRUB y LILO en cada caso. Incluya sus fotos o capturas con sus entregables.**

○

```
root@debian: /
File Edit View Terminal Help
--ignore-depends=<package>,...
--force-... Ignore dependencies involving <package>.
--no-force-...|--refuse-... Override problems (see --force-help).
--abort-after <n> Stop when problems encountered.
--abort-after <n> Abort after encountering <n> errors.

Comparison operators for --compare-versions are:
lt le eq ne ge gt (treat empty version as earlier than any version);
lt-nl le-nl ge-nl gt-nl (treat empty version as later than any version);
< << <= = >= >> > (only for compatibility with control file syntax).

Use `dselect' or `aptitude' for user-friendly package management.
root@debian:/# sudo dpkg-reconfigure linux-image-3.16.0-4-686-pae
Package `linux-image-3.16.0-4-686-pae' is not installed and no info is available.
Use dpkg --info (= dpkg-deb --info) to examine archive files,
and dpkg --contents (= dpkg-deb --contents) to list their contents.
/usr/sbin/dpkg-reconfigure: linux-image-3.16.0-4-686-pae is not installed
root@debian:/# sudo dpkg-reconfigure linux-image-2.6.39.4
Package `linux-image-2.6.39.4' is not installed and no info is available.
Use dpkg --info (= dpkg-deb --info) to examine archive files,
and dpkg --contents (= dpkg-deb --contents) to list their contents.
/usr/sbin/dpkg-reconfigure: linux-image-2.6.39.4 is not installed
root@debian:/#
```

### Mencione tres diferencias funcionales entre GRUB y LILO.

- GRUB tiene capacidad de leer particiones ext2
- GRUB tiene una interfaz de comando interactiva.
- LILO guarda información sobre la localización del kernel o de si existe otro OS que se deba cargar en el MBR.