

Josué Sagastume - 18173
Sistemas Operativos
LABORATORIO 4

Ejercicio 1

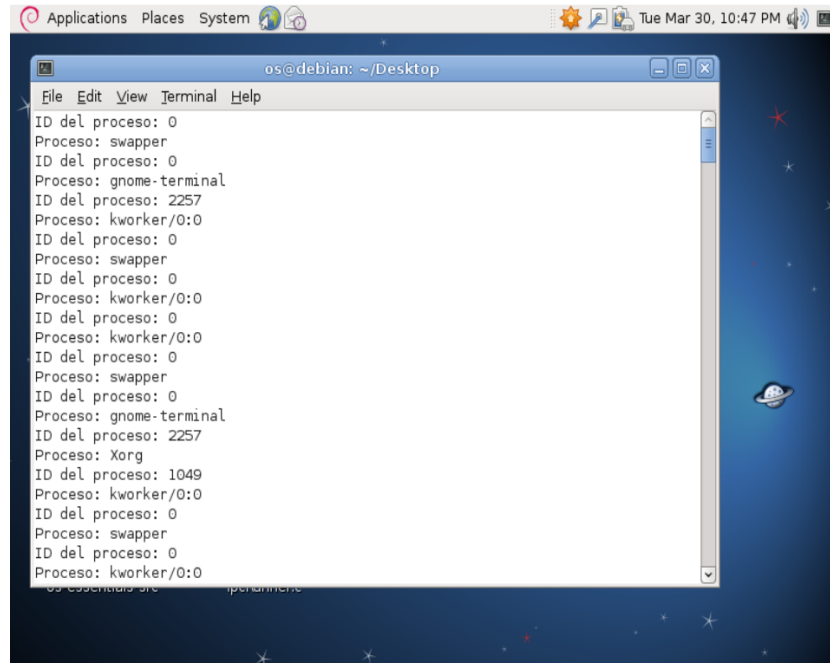


Imagen 1. Ejecución del comando *sudo stap profiler.stp*

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

Al interactuar con el sistema durante la ejecución, se puede ver que se crea el proceso `gnome-terminal` con su respectivo ID de proceso y también el proceso `Xorg` con su respectivo ID.

- **¿Para qué sirve SystemTap?**

Proporciona una infraestructura de software libre para simplificar la recopilación de información sobre el sistema Linux en ejecución. SystemTap puede extraer filtrar y resumir datos para permitir el diagnóstico de problemas complejos de rendimiento o funcionales (sourceware, s.f.).

- **¿Cómo funciona SystemTap?**

En SystemTap se manejan los “events” y los “handlers”, existen dos tipos de eventos, síncronos y asíncronos. Un eventos síncrono ocurre cuando cualquier proceso ejecuta una operación en el kernel, como entradas a syscalls, entradas en funciones kernel, etc. Y los

eventos asíncronos son aquellos que no están ligados a ninguna instrucción particular en el código. Los “handlers” es la parte del código llamada cuando ocurre un evento. Este código nos permite tratar la información del evento, almacenarla y generalmente utilizar cualquier función disponible en cualquier lenguaje moderno (Vela, 2010).

Ejercicio 2

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

Una función sin parámetros indica que la función toma una cantidad no especificada de parámetros, mientras que una función con un parámetro void sin nombre, indica que dicha función no recibe parámetros.

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

La diferencia es que *printk* es una función a nivel de kernel, que tiene la habilidad de imprimir en diferentes niveles, mientras que *printf* siempre imprimirá en un descriptor de archivo.

- **¿Qué es y para qué sirve *KERN_INFO*?**

Así como esta cadena existen más, estas especifican el nivel de registro. Consisten en el comienzo ASCII del carácter de encabezado seguido de un dígito que describe el nivel de registro. Para este caso, *KERN_INFO* indicaría un mensaje informativo.

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

Estas metas son objetivos que *Makefile* debería esforzarse por actualizar. De manera predeterminada, la meta es el primer objetivo en el archivo *make*, por lo tanto, estos archivos generalmente se escriben de manera que el primer objetivo sea compilar todo el programa. Por lo que en este caso obj-m sería la meta.

- **¿Qué función tienen las líneas *all:* y *clean:*?**

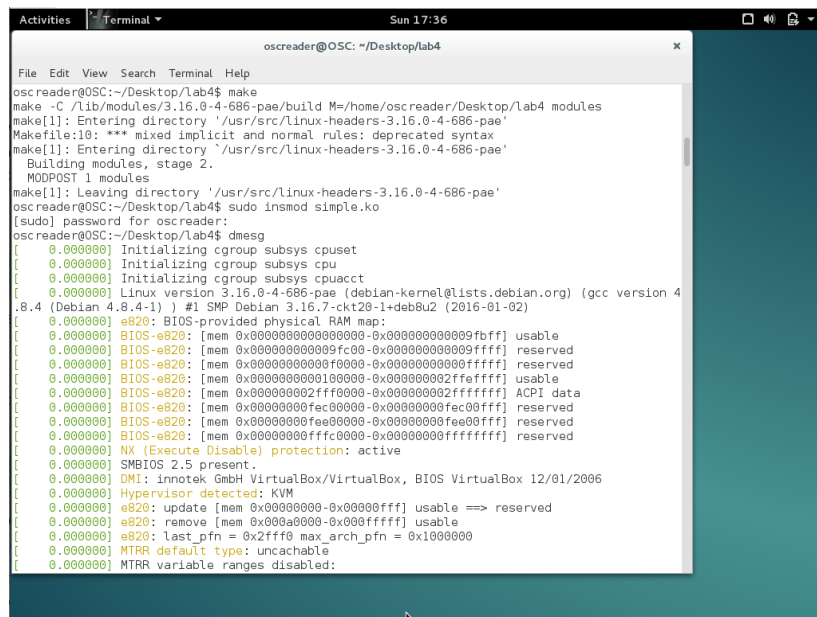
La función *all:* dice que se hagan todos los destinos de nivel superior que conoce el archivo *make*. Y la función *clean:* dice que se eliminen todos los archivos que normalmente se crean ejecutando *make*.

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

Este describe convenciones para escribir los *Makefiles* para programas GNU. En este caso **-C** cambia el directorio de *make*, entonces *make* va a leer el *Makefile* en este directorio.

- ¿Qué hace la opción **M** en este *Makefile*?

M es una variable asignada a la ejecución de *make*, si *make* ejecuta un *makefile*, este archivo puede leer la variable M y usar su contenido.

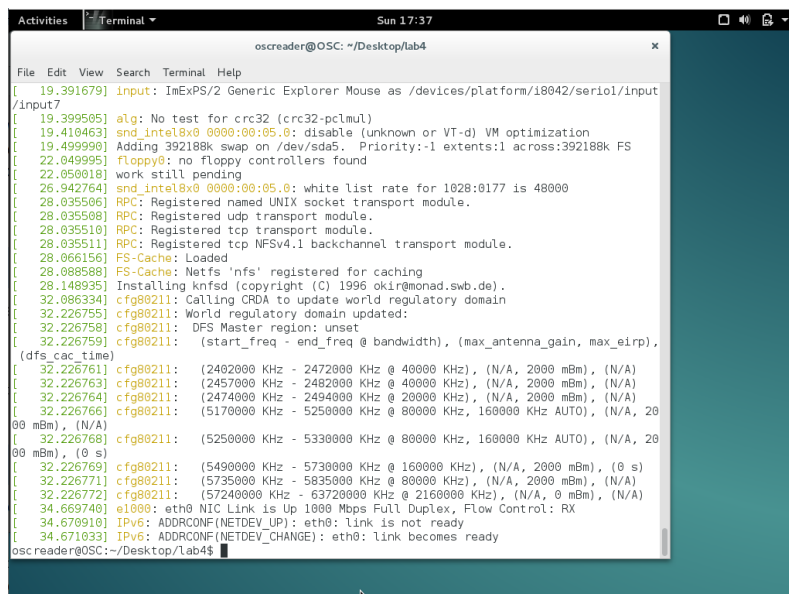


```

oscreader@OSC: ~/Desktop/lab4
File Edit View Search Terminal Help
oscreader@OSC:~/Desktop/lab4$ make
make -C /lib/modules/3.16.0-4-686-pae/build M=/home/oscreader/Desktop/lab4 modules
make[1]: Entering directory '/usr/src/linux-headers-3.16.0-4-686-pae'
Makefile:10: *** mixed implicit and normal rules: deprecated syntax
make[1]: Entering directory '/usr/src/linux-headers-3.16.0-4-686-pae'
Building modules, stage 2.
MODPOST 1 modules
make[1]: Leaving directory '/usr/src/linux-headers-3.16.0-4-686-pae'
oscreader@OSC:~/Desktop/lab4$ sudo insmod simple.ko
[sudo] password for oscreader:
oscreader@OSC:~/Desktop/lab4$ dmesg
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Initializing cgroup subsys cpuctl
[ 0.000000] Linux version 3.16.0-4-686-pae (debian-kernel@lists.debian.org) (gcc version 4
.8.4 (Debian 4.8.4-1) ) #1 SMP Debian 3.16.7-ckt20-1+deb8u2 (2016-01-02)
[ 0.000000] e820: BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000000fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000001000000-0x0000000002fffff] usable
[ 0.000000] BIOS-e820: [mem 0x000000002ffff000-0x000000002fffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: innovek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] e820: update [mem 0x00000000-0x0000ffff] usable ==> reserved
[ 0.000000] e820: remove [mem 0x000a0000-0x000fffff] usable
[ 0.000000] e820: last_pfn = 0x2ffff0 max_arch_pfn = 0x1000000
[ 0.000000] MTRR default type: uncachable
[ 0.000000] MTRR variable ranges disabled:

```

Imagen 2. Ejecución del comando *make* y *sudo insmod simple.ko*.



```

oscreader@OSC: ~/Desktop/lab4
File Edit View Search Terminal Help
[ 19.391679] input: ImEXPS/2 Generic Explorer Mouse as /devices/platform/i8042/serio/input
[ 19.391679] /input7
[ 19.399505] alg: No test for crc32 (crc32-pclmul)
[ 19.410463] snd_intel8x0 0000:00:05.0: disable (unknown or VT-d) VM optimization
[ 19.499990] Adding 392188k swap on /dev/sda5. Priority:-1 extents:1 across:392188k FS
[ 22.049995] floppy0: no floppy controllers found
[ 22.050018] work still pending
[ 26.942764] snd_intel8x0 0000:00:05.0: white list rate for 1028:0177 is 48000
[ 28.035508] RPC: Registered named UNIX socket transport module.
[ 28.035510] RPC: Registered tcp transport module.
[ 28.035511] RPC: Registered tcp NFSv4.1 backchannel transport module.
[ 28.066156] FS-Cache: Loaded
[ 28.088588] FS-Cache: Netfs 'nfs' registered for caching
[ 28.148935] Installing knfsd (copyright (C) 1996 okir@monad.swb.de).
[ 32.086334] cfg80211: Calling CRDA to update world regulatory domain
[ 32.226755] cfg80211: World regulatory domain updated:
[ 32.226758] cfg80211: DFS Master region: unset
[ 32.226759] (start_freq - end_freq @ bandwidth), (max_antenna_gain, max_eirp),
[ 32.226761] (dfs_cac_time)
[ 32.226763] cfg80211: (2402000 KHz - 2472000 KHz @ 40000 KHz), (N/A, 2000 mBm), (N/A)
[ 32.226764] cfg80211: (2457000 KHz - 2482000 KHz @ 40000 KHz), (N/A, 2000 mBm), (N/A)
[ 32.226764] cfg80211: (2474000 KHz - 2494000 KHz @ 20000 KHz), (N/A, 2000 mBm), (N/A)
[ 32.226766] cfg80211: (5170000 KHz - 5250000 KHz @ 80000 KHz, 160000 KHz AUTO), (N/A, 20
[ 32.226768] cfg80211: (5250000 KHz - 5330000 KHz @ 80000 KHz, 160000 KHz AUTO), (N/A, 20
[ 32.226769] cfg80211: (5490000 KHz - 5730000 KHz @ 160000 KHz), (N/A, 2000 mBm), (0 s)
[ 32.226771] cfg80211: (5735000 KHz - 5835000 KHz @ 80000 KHz), (N/A, 2000 mBm), (N/A)
[ 32.226772] cfg80211: (57240000 KHz - 63720000 KHz @ 2160000 KHz), (N/A, 0 mBm), (N/A)
[ 34.669740] e1000: eth0 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[ 34.670910] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
[ 34.671031] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
oscreader@OSC:~/Desktop/lab4$

```

Imagen 3. Ejecución del comando *dmesg*.

- **¿Para qué sirve dmesg?**

Es una instrucción que lista el buffer de mensajes del núcleo. Este buffer contiene una gran variedad de mensajes importantes generados durante el arranque del sistema, la detección del hardware, asignación de controladores y durante la depuración de aplicaciones.

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

Debido a que esta función se da como parámetro a la función *module_init*, entonces esta función se llamará en el momento de la inserción del módulo, o en el momento del arranque.

The screenshot shows a terminal window titled "simple.c" with the path "~/Desktop/lab-4". The user is logged in as "oscreader@OSC" in the directory "~/Desktop/lab4".

```
#ifncl
#ifncl
int s
}
void
}

moduleX
module[
module[
module[
MODULE
HOLA :D
Adios :D
HOLA :D
Adios :D

oscreader@OSC:~/Desktop/lab4$
```

The terminal output displays the following information:

- File Edit View Search Terminal Help**
- (Bm), (N/A)**
- [32.226766] cfg80211: (5170000 KHz - 5250000 KHz @ 80000 KHz, 160000 KHz AU TO), (N/A, 2000 mBm), (N/A)**
- [32.226768] cfg80211: (5250000 KHz - 5330000 KHz @ 80000 KHz, 160000 KHz AU TO), (N/A, 2000 mBm), (0 s)**
- [32.226769] cfg80211: (5490000 KHz - 5730000 KHz @ 160000 KHz), (N/A, 2000 mBm), (0 s)**
- [32.226771] cfg80211: (5735000 KHz - 5835000 KHz @ 80000 KHz), (N/A, 2000 mBm), (N/A)**
- [32.226772] cfg80211: (57240000 KHz - 63720000 KHz @ 2160000 KHz), (N/A, 0 mBm), (N/A)**
- [34.669740] e1000: eth0 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: R**
- [34.670910] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready**
- [34.671033] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready**
- [1276.337750] Loading Module**
- HOLA :D**
- [1453.610713] Removing Module**
- Adios :D**
- [1607.704793] Loading Module**
- HOLA :D**
- [1622.609799] Removing Module**
- Adios :D**

The status bar at the bottom indicates "Tab Width: 8", "Ln 7, Col 25", and "INS".

Imagen 4. Ejecución del comando *sudo rmmmod simple*.

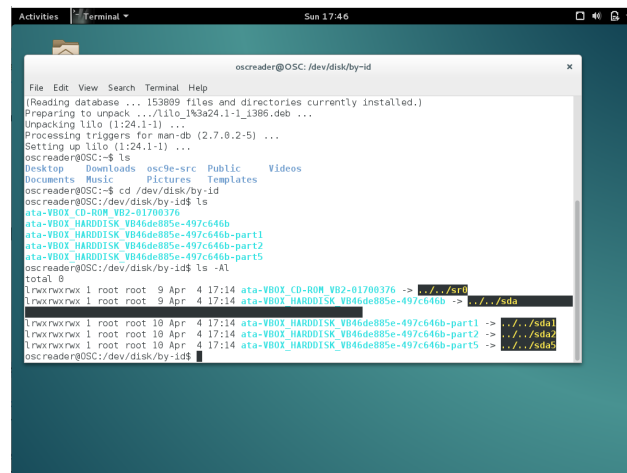
- ¿Qué hace la función `simple_exit` en su programa `simple.c`?

Como esta es dada como parámetro a la función *module_exit*, entonces esta se llamará en el momento de la eliminación del módulo. Solo se llamará si el recuento de uso del módulo ha llegado a cero.

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

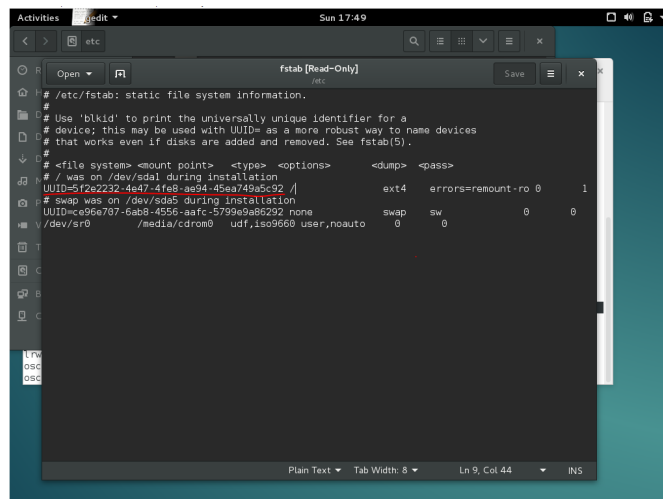
Con esto se podría mejorar la funcionalidad del sistema sin necesidad de compilar el kernel o reiniciar el sistema.

Ejercicio 3



```
oscreader@OSC: /dev/disk/by-id
[Reading database ... 153809 files and directories currently installed.]
Preparing to unpack .../lilo_1%3a24.1-1_1386.deb ...
Unpacking lilo (1:24.1-1) ...
Processing triggers for man-db (2.7.8.2-5) ...
Setting up lilo (1:24.1-1) ...
oscreader@OSC:~$ ls
Desktop  Downloads  osc9e-src  Public  Videos
Documents  Music  Pictures  Templates
oscreader@OSC:~$ cd /dev/disk/by-id
oscreader@OSC:/dev/disk/by-id$ ls
ata-VBOX_CD-ROM_VB2-01700376
ata-VBOX_HARDDISK_VB46de885e-497c646b
ata-VBOX_HARDDISK_VB46de885e-497c646b-part1
ata-VBOX_HARDDISK_VB46de885e-497c646b-part2
ata-VBOX_HARDDISK_VB46de885e-497c646b-part5
oscreader@OSC:/dev/disk/by-id$ ls -Al
total 0
lrwxrwxrwx 1 root root 9 Apr 4 17:14 ata-VBOX_CD-ROM_VB2-01700376 -> ../../sda
lrwxrwxrwx 1 root root 9 Apr 4 17:14 ata-VBOX_HARDDISK_VB46de885e-497c646b -> ../../sda
lrwxrwxrwx 1 root root 10 Apr 4 17:14 ata-VBOX_HARDDISK_VB46de885e-497c646b-part1 -> ../../sda1
lrwxrwxrwx 1 root root 10 Apr 4 17:14 ata-VBOX_HARDDISK_VB46de885e-497c646b-part2 -> ../../sda2
lrwxrwxrwx 1 root root 10 Apr 4 17:14 ata-VBOX_HARDDISK_VB46de885e-497c646b-part5 -> ../../sda5
oscreader@OSC:/dev/disk/by-id$
```

Imagen 5. Ejecución del comando `ls -Al`.



```
# /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/sda1 during installation
UUID=512e2232-4e47-4f68-a634-d5e74b2de92 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=c96e707-6ab8-4556-aafc-5799e9a86292 none swap sw 0 0
v /dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0
```

Imagen 6. Archivo `fstab` del directorio `/etc`.

- ¿Qué es y para qué sirve el archivo `fstab`?

Este forma parte de la configuración del sistema. Este lista los discos y particiones disponibles, y este indica cómo montar cada dispositivo y qué configuración utilizar.

- ¿Qué almacena el directorio `/etc`?

Este directorio es el encargado de almacenar los archivos de configuración, tanto a nivel de componentes del sistema como de los programas y aplicaciones instaladas. Este debería contener únicamente archivos de configuración.

- ¿Qué se almacena en /dev?

Este directorio incluye todos los dispositivos de almacenamiento conectados al sistema. Es decir, contiene cualquier disco duro, partición, USB, CD-ROM, en forma de archivo, que este conectado al sistema.

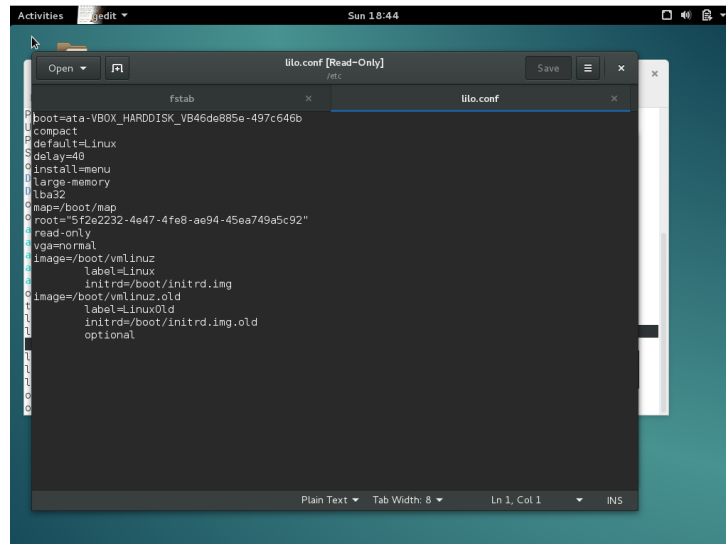


Imagen 7. Archivo lilo.conf.

Referencias

Ampalio, F. (2020). *El archivo fstab al descubierto*. Extraído de:
<https://blog.carreralinux.com.ar/2020/04/el-archivo-fstab-al-descubierto/>

Gnu.org. (s.f.). *Arguments to Specify the Goals*. Extraído de:
https://www.gnu.org/software/make/manual/html_node/Goals.html#:~:text=The%20goals%20are%20the%20targets%20that%20make%20should%20strive%20ultimately%20to%20update.&text=By%20default%2C%20the%20goal%20is,program%20or%20programs%20they%20describe.

Kernel.org. (s.f.). *Common Routines*. Extraído de:
<https://www.kernel.org/doc/html/docs/kernel-hacking/routines-moduleexit.html>

Sourceware, (s.f.). *SystemTap*. Extraído de: <https://sourceware.org/systemtap/index.html>

Vela, J. (2010). *Introducción a SystemTap*. Extraído de:
<https://www.securityartwork.es/2010/11/16/introduccion-a-systemtap/>