



ugr

Universidad
de Granada

SEMINARIO

Módulos Cargables del Kernel (LKM)

Periféricos y Dispositivos de Interfaz Humana

Autor:
Clara Sola Ruiz

Índice

1. Introducción	3
2. Realización del seminario	3

1. Introducción

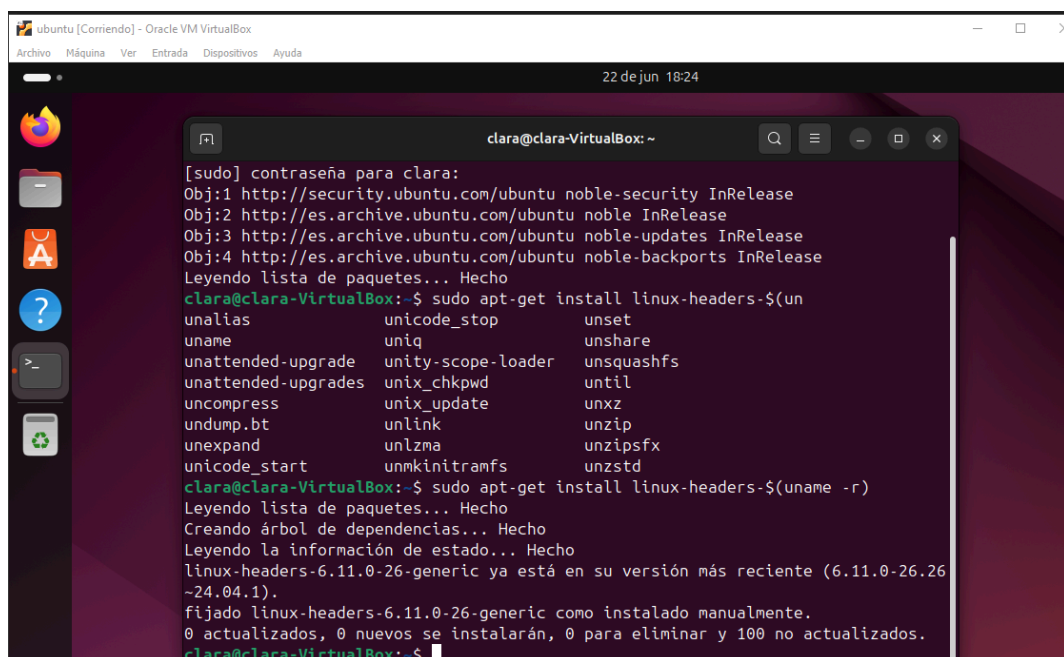
Este seminario tiene como objetivo comprender y experimentar con el funcionamiento de los módulos cargables de kernel (LKM), los cuales permiten extender la funcionalidad del núcleo de forma dinámica, sin necesidad de recompilar ni reiniciar el sistema operativo.

En este seminario desarrollamos un módulo LKM sencillo en lenguaje C, se ha compilado utilizando un Makefile y se ha probado su funcionalidad insertándolo y eliminándolo del kernel.

A continuación, se documenta mediante capturas de pantalla el proceso que se ha seguido.

2. Realización del seminario

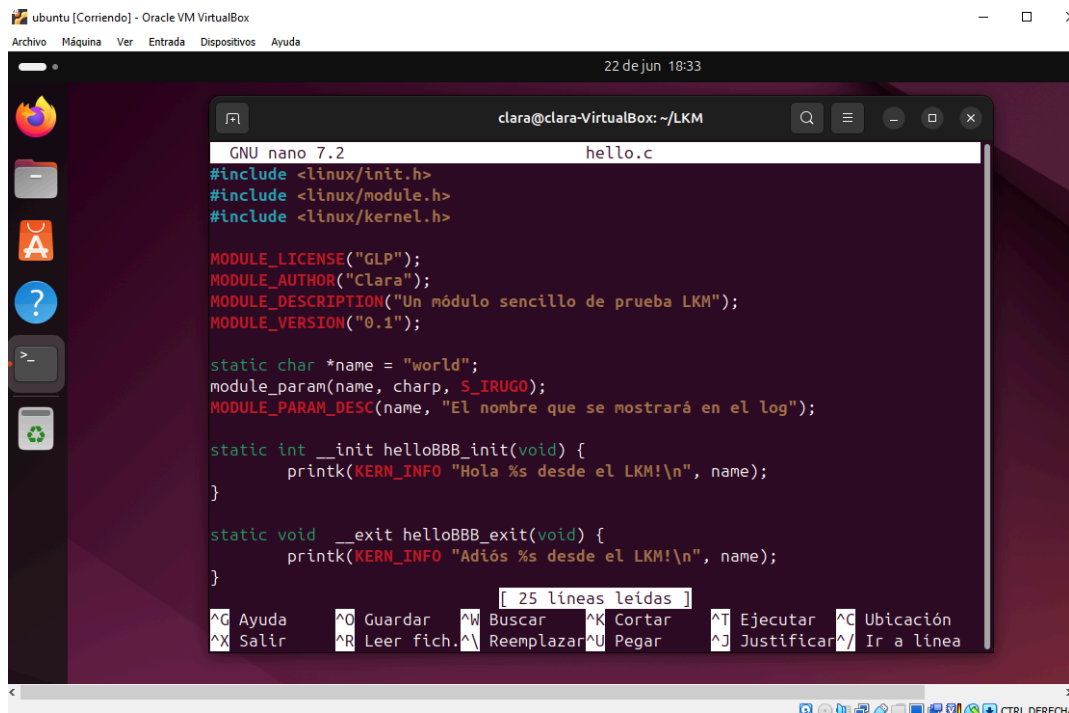
Usamos una máquina virtual para este seminario. En primer lugar instalamos las cabeceras del kernel:



```
ubuntu [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
22 de jun 18:24

clara@clara-VirtualBox: ~
[sudo] contraseña para clara:
Obj:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Obj:2 http://es.archive.ubuntu.com/ubuntu noble InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu noble-updates InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu noble-backports InRelease
Leyendo lista de paquetes... Hecho
clara@clara-VirtualBox:~$ sudo apt-get install linux-headers-$(un
unalias          unicode_stop      unset
uname            uniq            unshare
unattended-upgrade unity-scope-loader unsquashfs
unattended-upgrades unix_chkpwd      until
uncompress       unix_update     unxz
undump.bt        unlink          unzip
unexpand         unlzma          unzipsfx
unicode_start    unmkinitramfs  unzstd
clara@clara-VirtualBox:~$ sudo apt-get install linux-headers-$(uname -r)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
linux-headers-6.11.0-26-generic ya está en su versión más reciente (6.11.0-26.26
~24.04.1).
fijado linux-headers-6.11.0-26-generic como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 100 no actualizados.
clara@clara-VirtualBox:~$
```

Posteriormente hemos creado una carpeta denominada *LKM*. A continuación creamos el archivo *hello.c*:



```
GNU nano 7.2 hello.c
#include <linux/init.h>
#include <linux/module.h>
#include <linux/kernel.h>

MODULE_LICENSE("GPL");
MODULE_AUTHOR("Clara");
MODULE_DESCRIPTION("Un módulo sencillo de prueba LKM");
MODULE_VERSION("0.1");

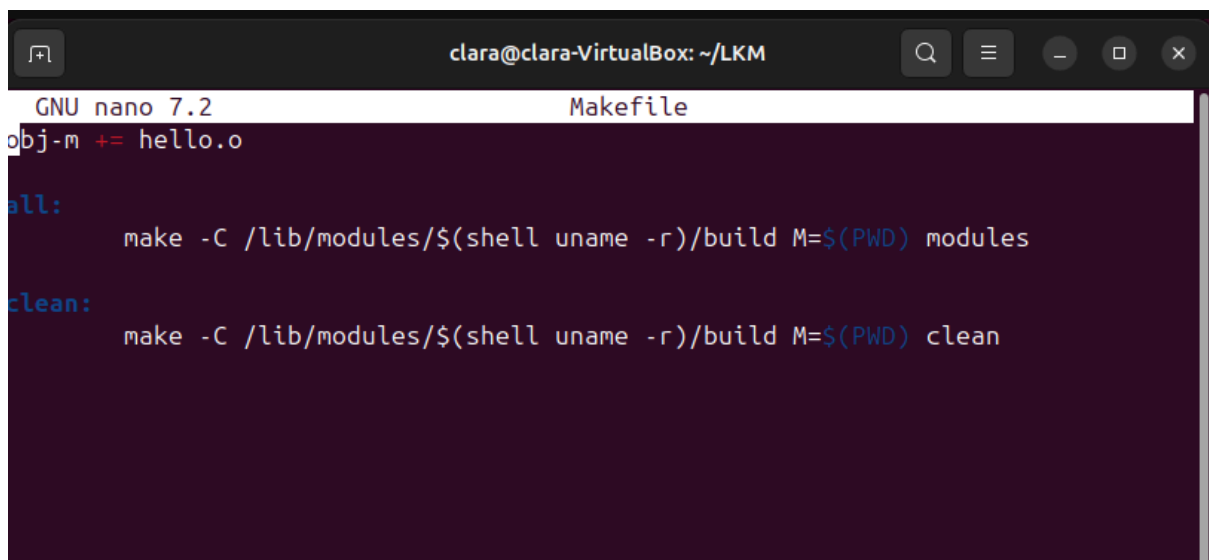
static char *name = "world";
module_param(name, charp, S_IRUGO);
MODULE_PARAM_DESC(name, "El nombre que se mostrará en el log");

static int __init helloBBB_init(void) {
    printk(KERN_INFO "Hola %s desde el LKM!\n", name);
}

static void __exit helloBBB_exit(void) {
    printk(KERN_INFO "Adiós %s desde el LKM!\n", name);
}

[ 25 líneas leídas ]
^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^C Ubicación
^X Salir ^R Leer fich. ^I Reemplazar ^U Pegar ^J Justificar ^_/ Ir a línea
```

Creado el archivo *hello.c* podemos crear el *makefile*:



```
GNU nano 7.2 Makefile
obj-m += hello.o

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

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

Una vez creado el *makefile* lo ejecutamos con *make*:

```

clara@clara-VirtualBox:~/LKM$ make
make -C /lib/modules/6.11.0-26-generic/build M=/home/clara/LKM modules
make[1]: se entra en el directorio '/usr/src/linux-headers-6.11.0-26-generic'
warning: the compiler differs from the one used to build the kernel
The kernel was built by: x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
You are using: gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
CC [M] /home/clara/LKM/hello.o
MODPOST /home/clara/LKM/Module.symvers
CC [M] /home/clara/LKM/hello.mod.o
LD [M] /home/clara/LKM/hello.ko
BTF [M] /home/clara/LKM/hello.ko
Skipping BTF generation for /home/clara/LKM/hello.ko due to unavailability of vmlinux
make[1]: se sale del directorio '/usr/src/linux-headers-6.11.0-26-generic'
clara@clara-VirtualBox:~/LKM$ ls
hello.c  hello.mod  hello.mod.o  Makefile      Module.symvers
hello.ko  hello.mod.c  hello.o      modules.order
clara@clara-VirtualBox:~/LKM$

```

Cargamos el módulo, verificamos que esté cargado y consultamos su información:

```

clara@clara-VirtualBox:~/LKM$ sudo insmod hello.ko
clara@clara-VirtualBox:~/LKM$ lsmod | grep hello
hello                12288  0
clara@clara-VirtualBox:~/LKM$ modinfo hello.ko
filename:            /home/clara/LKM/hello.ko
version:             0.1
description:         Un módulo sencillo de prueba LKM
author:              Clara
license:             GLP
srcversion:          D7F5898C01C849719538A2A
depends:
retpoline:           Y
name:                hello
vermagic:            6.11.0-26-generic SMP preempt mod_unload modversions
parm:                name:El nombre que se mostrará en el log (charp)

```

Por último revisamos el log y descargamos el módulo:

```
clara@clara-VirtualBox:~/LKM$ sudo rmmod hello.ko
clara@clara-VirtualBox:~/LKM$ dmesg | tail
dmesg: fallo al leer el «buffer» del núcleo: Operación no permitida
clara@clara-VirtualBox:~/LKM$ sudo su -
root@clara-VirtualBox:~# tail -f kern.log
tail: no se puede abrir 'kern.log' para lectura: No existe el archivo o el directorio
tail: no queda ningún fichero
root@clara-VirtualBox:~# dmesg | tail
 228.293724] [drm:vmw_msg_ioctl [vmwgfx]] *ERROR* Failed to open channel.
 228.294002] [drm:vmw_msg_ioctl [vmwgfx]] *ERROR* Failed to open channel.
1104.158656] hrtimer: interrupt took 7372357 ns
1303.528949] hello: loading out-of-tree module taints kernel.
1303.528954] hello: module license 'GLP' taints kernel.
1303.528956] Disabling lock debugging due to kernel taint
1303.528957] hello: module verification failed: signature and/or required key
missing - tainting kernel
1303.528958] hello: module license taints kernel.
1303.529944] Hola world desde el LKM!
1416.401749] Adiós world desde el LKM!
root@clara-VirtualBox:~#
```