Práctica 3
Análisis forense de sistemas Linux

En la práctica anterior, utilizamos la máquina virtual. Ahora, nos sumergimos más profundamente en las herramientas al aprender a compilar el kernel. Mientras que en la práctica anterior nos proporcionaron el kernel precompilado, esta vez lo compilamos nosotros mismos. Para lograrlo, emplearé Debian 10 con el kernel 4.19.0-16-amd64.

```
alumno@debian:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Debian
Description:    Debian GNU/Linux 10 (buster)
Release:        10
Codename:       buster
alumno@debian:~$ uname -r
4.19.0-16-amd64
```

fmem

Para fmem, descargaremos la herramienta desde su repositorio en GitHub. Allí, encontraremos un archivo Makefile que automatiza todo el proceso de compilación del kernel. Por lo tanto, solo necesitaremos ejecutar el comando make para llevar a cabo la compilación

```
alumno@debian:/media/alumno/42EC-66CA/fmem$ make
rm -f *.o *.ko *.mod.c Module.symvers Module.markers modules.order \.*.o.cmd \.*.ko.cmd \.*.o.d
rm -rf \.tmp_versions
make -C /lib/modules/`uname -r`/build KBUILD_EXTMOD=`pwd` modules
make[1]: se entra en el directorio '/usr/src/linux-headers-4.19.0-16-amd64'
  CC [M]  /media/alumno/42EC-66CA/fmem/lkm.o
  LD [M]  /media/alumno/42EC-66CA/fmem/fmem.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC      /media/alumno/42EC-66CA/fmem/fmem.mod.o
  LD [M]  /media/alumno/42EC-66CA/fmem/fmem.ko
make[1]: se sale del directorio '/usr/src/linux-headers-4.19.0-16-amd64'
alumno@debian:/media/alumno/42EC-66CA/fmem$
```

Al ejecutar este comando, el kernel se compila, generando el archivo **fmem.ko**,

```
alumno@debian:/media/alumno/42EC-66CA/fmem$ ls
AUTHORS    COPYING  fmem.ko     fmem.mod.o  lkm.c  Makefile       Module.symvers  run.sh
ChangeLog  debug.h  fmem.mod.c  fmem.o      lkm.o  modules.order  README          TODO
alumno@debian:/media/alumno/42EC-66CA/fmem$
```

Una vez generado el archivo **fmem.ko**, podemos seguir los mismos pasos que en la práctica anterior, ejecutando el script que tomará el archivo fmem.ko para montar el dispositivo **/dev/fmem**.

```
alumno@debian:/media/alumno/42EC-66CA/fmem$ sudo ./run.sh
./run.sh: 6: [: 0xffffffffb467f650: unexpected operator
Module: insmod fmem.ko a1=0xffffffffb467f650 : OK
Device: /dev/fmem
----Memory areas: -----
----------------------
!!! Don't forget add "count=" to dd !!!
```

```
alumno@debian:/media/alumno/42EC-66CA/fmem$ free -m
              total        used        free      shared  buff/cache   available
Mem:           1995         677         606          18         711        1149
Swap:           974           0         974
alumno@debian:/media/alumno/42EC-66CA/fmem$ sudo dd if=/dev/fmem of=//media/alumno/42EC-66CA/volcado1.raw bs=1MB count=1995
1995+0 registros leídos
1995+0 registros escritos
1995000000 bytes (2,0 GB, 1,9 GiB) copied, 332,575 s, 6,0 MB/s
alumno@debian:/media/alumno/42EC-66CA/fmem$
```

Lime

Para LIME, también encontraremos la herramienta en su repositorio de GitHub. Navegaremos al directorio src, donde encontraremos un archivo Makefile que automatiza el proceso de compilación del kernel, al igual que en el caso anterior.

```
alumno@debian:/media/alumno/42EC-66CA/LiME/src$ make
make -C /lib/modules/4.19.0-16-amd64/build M="/media/alumno/42EC-66CA/LiME/src" modules
make[1]: se entra en el directorio '/usr/src/linux-headers-4.19.0-16-amd64'
  CC [M]  /media/alumno/42EC-66CA/LiME/src/tcp.o
/media/alumno/42EC-66CA/LiME/src/tcp.c: In function 'setup_tcp':
/media/alumno/42EC-66CA/LiME/src/tcp.c:75:5: warning: ISO C90 forbids mixed declarations and code [-Wdeclaration-after-statement]
     int opt = 1;
     ^~~
  CC [M]  /media/alumno/42EC-66CA/LiME/src/disk.o
  CC [M]  /media/alumno/42EC-66CA/LiME/src/main.o
  CC [M]  /media/alumno/42EC-66CA/LiME/src/hash.o
  CC [M]  /media/alumno/42EC-66CA/LiME/src/deflate.o
  LD [M]  /media/alumno/42EC-66CA/LiME/src/lime.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC      /media/alumno/42EC-66CA/LiME/src/lime.mod.o
  LD [M]  /media/alumno/42EC-66CA/LiME/src/lime.ko
make[1]: se sale del directorio '/usr/src/linux-headers-4.19.0-16-amd64'
strip --strip-unneeded lime.ko
mv lime.ko lime-4.19.0-16-amd64.ko
```

Una vez finalizado, se generará el archivo con el kernel compilado.

```
alumno@debian:/media/alumno/42EC-66CA/LiME/src$ ls
deflate.c  disk.c  hash.c  lime-4.19.0-16-amd64.ko  lime.mod.c  lime.o  main.o    Makefile.sample  Module.symvers  tcp.o
deflate.o  disk.o  hash.o  lime.h                   lime.mod.o  main.c  Makefile  modules.order    tcp.c
alumno@debian:/media/alumno/42EC-66CA/LiME/src$
```

Una vez tengamos el archivo, podremos continuar con el proceso de adquisición como lo hicimos en la práctica anterior, utilizando el comando insmod.

```
alumno@debian:/media/alumno/42EC-66CA/LiME/src$ sudo insmod lime-4.19.0-16-amd64.ko "path=/media/alumno/42EC-66CA/volcado2.raw
format=raw"
[sudo] password for alumno:
```

```
alumno@debian:/media/alumno/42EC-66CA$ ls -la
total 4045572
drwxrwxrwx  1 alumno alumno     131072 mar 30 23:53  .
drwxr-x---+ 3 root   root         4096 mar 30 23:30  ..
drwxrwxrwx  1 alumno alumno     131072 mar 30 23:33  fmem
drwxrwxrwx  1 alumno alumno     131072 mar 30 23:29  LiME
drwxrwxrwx  1 alumno alumno     131072 mar 22 16:45 'System Volume Information'
-rwxrwxrwx  1 alumno alumno 1995000000 mar 30 23:41  volcado1.raw
-rwxrwxrwx  1 alumno alumno 2147019776 mar 31 01:34  volcado2.raw
alumno@debian:/media/alumno/42EC-66CA$
```