

Universidad Mariano Galvez

Ingeniería en Ciencias y Sistemas de la información

Sistemas Operativos II

Ing. Mélvín Cali

### PROYECTO III

Kevin Josué Del Aguila Pérez 7691-19-15167

Pedro Luis Juarez Osorio 7691-19-17925

Juan José Estrada Ramirez 7691-17-13751

Instalamos el compilador que nos servirá para la compilación de nuestro kernel

```
kevin@kevin:~$ sudo apt install gcc libncurses5-dev dpkg-dev
[sudo] password for kevin:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
gcc ya está en su versión más reciente (4:10.2.1-1).
libncurses5-dev ya está en su versión más reciente (6.2+20201114-2).
dpkg-dev ya está en su versión más reciente (1.20.10).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 10 no actualizados.
kevin@kevin:~$ sudo apt upgrade gcc libncurses5-dev dpkg-dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
gcc ya está en su versión más reciente (4:10.2.1-1).
libncurses5-dev ya está en su versión más reciente (6.2+20201114-2).
dpkg-dev ya está en su versión más reciente (1.20.10).
Calculando la actualización... Hecho
Se actualizarán los siguientes paquetes:
  firefox-esr firefox-esr-l10n-es-ar firefox-esr-l10n-es-cl
  firefox-esr-l10n-es-es firefox-esr-l10n-es-mx gir1.2-javascriptcoregtk-4.0
  gir1.2-webkit2-4.0 libjavascriptcoregtk-4.0-18 libwebkit2gtk-4.0-37 rsyslog
10 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 0 B/80.8 MB de archivos.
Se utilizarán 139 kB de espacio de disco adicional después de esta operación.
```

Descargamos el kernel, en este caso usaremos el kernel 5.10

```
kevin@kevin:~$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.10.1.tar.xz
```

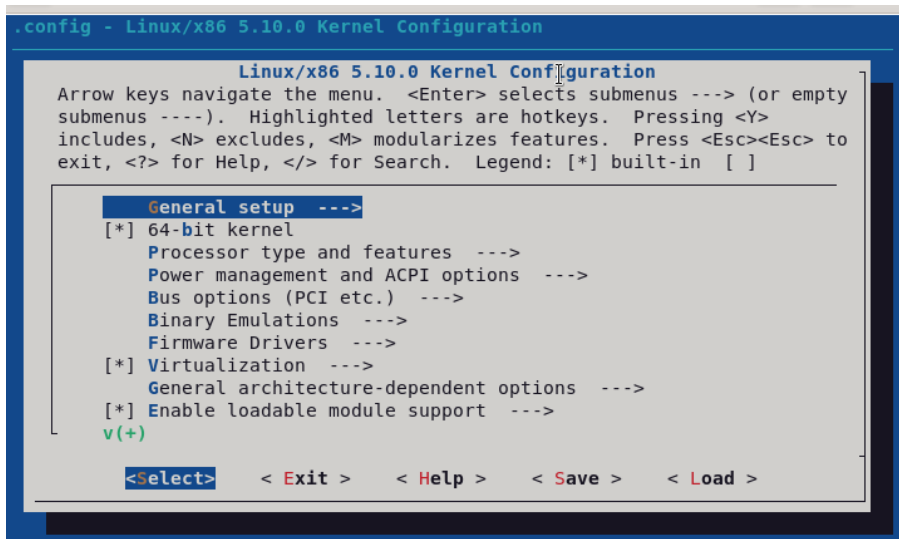
Una vez descargado el kernel, lo desempaquetaremos

```
kevin@kevin:~$ tar Jxvf linux-5.10.tar.xz
```

Entramos a la carpeta y comenzaremos a personalizar nuestro kernel con el comando `sudo make menuconfig`

```
kevin@kevin:~$ cd linux-5.10/
kevin@kevin:~/linux-5.10$ sudo make menuconfig
```

Se abrirá la ventana en donde podremos cambiar las configuraciones y también personalizar nuestro kernel. Guardamos los cambios en la opción `save`



Compilaremos el kernel con el comando sudo make

```
kevin@kevin:~/linux-5.10$ sudo make
```

Una vez compilado el kernel tendremos el archivo para que podamos instalarlo con el comando make modules\_install

```
CALL    scripts/checksyscalls.sh
CALL    scripts/atomic/check-atomics.sh
DESCEND objtool
CHK     include/generated/compile.h
Kernel: arch/x86/boot/bzImage is ready (#2)
```

```
kevin@kevin:~/linux-5.10$ sudo make modules_install
INSTALL drivers/firmware/google/coreboot_table.ko
INSTALL drivers/firmware/google/gsmi.ko
INSTALL drivers/firmware/google/memconsole-coreboot.ko
INSTALL drivers/firmware/google/memconsole-x86-legacy.ko
INSTALL drivers/firmware/google/memconsole.ko
INSTALL drivers/firmware/google/vpd-sysfs.ko
INSTALL drivers/thermal/intel/x86_pkg_temp_thermal.ko
INSTALL fs/efivarfs/efivarfs.ko
INSTALL net/ipv4/netfilter/iptables_nat.ko
INSTALL net/ipv4/netfilter/nf_log_arp.ko
INSTALL net/ipv4/netfilter/nf_log_ipv4.ko
INSTALL net/ipv6/netfilter/nf_log_ipv6.ko
INSTALL net/netfilter/nf_log_common.ko
INSTALL net/netfilter/xt_LOG.ko
INSTALL net/netfilter/xt_MASQUERADE.ko
INSTALL net/netfilter/xt_addrtype.ko
INSTALL net/netfilter/xt_mark.ko
INSTALL net/netfilter/xt_nat.ko
DEPMOD  5.10.0
```

```
kevin@kevin:~/linux-5.10$ sudo make -j12 && sudo make modules_install -j12
```

```
DESCEND objtool
CALL    scripts/atomic/check-atomics.sh
CALL    scripts/checksyscalls.sh
CHK     include/generated/compile.h
Kernel: arch/x86/boot/bzImage is ready (#2)
INSTALL drivers/firmware/google/coreboot_table.ko
INSTALL drivers/firmware/google/gsmi.ko
INSTALL drivers/firmware/google/memconsole-coreboot.ko
INSTALL drivers/firmware/google/memconsole-x86-legacy.ko
INSTALL drivers/firmware/google/memconsole.ko
INSTALL drivers/firmware/google/vpd-sysfs.ko
INSTALL drivers/thermal/intel/x86_pkg_temp_thermal.ko
INSTALL fs/efivarfs/efivarfs.ko
INSTALL net/ipv4/netfilter/iptables_nat.ko
INSTALL net/ipv4/netfilter/nf_log_arp.ko
INSTALL net/ipv4/netfilter/nf_log_ipv4.ko
INSTALL net/ipv6/netfilter/nf_log_ipv6.ko
INSTALL net/netfilter/nf_log_common.ko
INSTALL net/netfilter/xt_LOG.ko
INSTALL net/netfilter/xt_MASQUERADE.ko
INSTALL net/netfilter/xt_addrtype.ko
INSTALL net/netfilter/xt_mark.ko
```

Nombramos nuestro kernel al grup de Linux

```
kevin@kevin:~/linux-5.10$ sudo cp arch/x86/boot/bzImage /boot/vmlinuz-KevinPedroJuanjo
```

Podemos ver que fue añadido satisfactoriamente

```
kevin@kevin:~/linux-5.10$ ls /boot/
config-5.10.0-10-amd64      System.map-5.10.0-14-amd64
config-5.10.0-14-amd64     vmlinuz-5.10.0-10-amd64
grub                       vmlinuz-5.10.0-14-amd64
initrd.img-5.10.0-10-amd64 vmlinuz-KevinPedroJuanjo
initrd.img-5.10.0-14-amd64 vmlinuz-KevinPedroJuanjo7691-19-15167
System.map-5.10.0-10-amd64
```

Al iniciar la maquina virtual podremos ver como el kernel fue compilado

```
*Debian GNU/Linux, with Linux KevinPedroJuanjo7691-19-15167
Debian GNU/Linux, with Linux KevinPedroJuanjo7691-19-15167 (recovery mo→
Debian GNU/Linux, with Linux KevinPedroJuanjo
Debian GNU/Linux, with Linux KevinPedroJuanjo (recovery mode)
Debian GNU/Linux, with Linux 5.10.0-14-amd64
Debian GNU/Linux, with Linux 5.10.0-14-amd64 (recovery mode)
Debian GNU/Linux, with Linux 5.10.0-10-amd64
Debian GNU/Linux, with Linux 5.10.0-10-amd64 (recovery mode)
```

LINK DE LOS VIDEOS EXPLICANDO FUNCIONAMIENTO:

<https://drive.google.com/drive/folders/17hRs3-Lvu1y0iO8Asqhi0M41SFeOFML9?usp=sharing>