

# Prácticas

## 1. GRUB, GRUB2 y el proceso de boot

Objetivo: iniciar Linux en un modo mínimo en caso de pérdida de contraseña. Esta artimaña no parece funcionar en Ubuntu.

1. Encienda su PC y deténgase en el menú de GRUB. Si está en modo gráfico, pulse [Esc]. En la entrada por defecto, pulse la tecla e para editar la información de boot de este menú.
2. Con las teclas de dirección, vaya a la línea que empieza por "kernel" y pulse la tecla e para editarlo. Con GRUB2, ubíquese en el final de la línea que comienza con "Linux".
3. Al final de esta línea, añada init=/bin/sh y pulse [Entrar]. Finalmente, pulse b para bootear. Con GRUB2, no pulse [Entrar] sino [Ctrl] x.
4. Después de unos segundos aparece un prompt. Pase al teclado español (qwerty por defecto).

```
# loadkeys es
```

5. Vuelva a montar el sistema de archivos raíz en sólo lectura:

```
# mount -o remount,rw /2
```

6. Ahora, teclee "passwd" y valide: puede cambiar la contraseña de root.

```
# passwd
```

## 2. init y runlevel

Objetivo: manejar la configuración de init y de los niveles de ejecución en sistemas.

1. Si arranca en modo gráfico, modifique /etc/inittab para iniciar en el nivel 3. Cambie el valor de 5 a 3 en la línea initdefault y vuelva a bootear.

```
id:3:initdefault:
```

2. Liste los servicios que se inician en el nivel 3:

```
# ls -l /etc/init.d/rc3.d/S*
```

3. Si el servicio sshd no arranca, actívalo en los niveles 3 y 5:

```
# chkconfig -list sshd  
# chkconfig sshd --level 35
```

4. chkconfig no inicia el servicio, pero lo activa para el próximo boot o cambio de nivel. Inicie sshd:

```
# service sshd start
```

5. Pase a la consola virtual 6 con [Alt][F6] (o [Ctrl][Alt][F6] bajo X). Conéctese y luego desconéctese. ¿Por qué el terminal vuelve a su punto de partida? Porque /etc/inittab contiene el comando **respawn**, que permite al proceso iniciarse de nuevo si había finalizado.

6. Con el comando **shutdown**, apague el ordenador ahora. ¿Cuál es el nivel de ejecución activado?

```
# shutdown -r now
```

Es el nivel 0, de parada, el que está activado.

### 3. Núcleo y módulos

Objetivo: gestionar los parámetros dinámicos del núcleo y los módulos.

1. Verifique su versión del núcleo de Linux y diríjase al directorio de sus módulos:

```
# uname -r  
2.6.25.5-1.1-default  
# cd /lib/modules/2.6.25.5-1.1-default
```

2. Verifique la fecha del archivo módulos.dep. Si parece antigua, inicie un comando para regenerarla.

```
# ls -l modules.dep  
-rw-r--r-- 1 root root 385503 junio 18 20:41 modules.dep  
# depmod -a  
# ls -l modules.dep  
-rw-r--r-- 1 root root 385601 junio 29 20:12 modules.dep
```

3. Liste los módulos actualmente cargados. Si no está el módulo vfat, cárguelo con sus dependencias.

```
# lsmod  
# lsmod | grep vfat  
# modprobe vfat
```

4. Del mismo modo, descargue el módulo vfat y sus dependencias.

```
# modprobe -r vfat
```

5. El parámetro dinámico arp\_announce del núcleo permite modificar los encabezamientos ARP en función de la dirección de destino del paquete. En una máquina que dispone de varias tarjetas de red, el valor por defecto puede ocasionar problemas, ya que Linux puede contestar con la dirección de una tarjeta cualquiera. La tarjeta tiene que contestar con una dirección de la misma subred que la de destino. Verifique qué parámetros del núcleo se ven involucrados:

```
# sysctl -a | grep arp_announce  
net.ipv4.conf.all.arp_announce = 0  
net.ipv4.conf.default.arp_announce = 0  
net.ipv4.conf.lo.arp_announce = 0  
net.ipv4.conf.eth0.arp_announce = 0  
net.ipv4.conf.eth1.arp_announce = 0  
net.ipv4.conf.eth2.arp_announce = 0
```

6. Modifique de manera dinámica para el conjunto de los adaptadores el valor arp\_announce a 1. El manual de sysctl indica que se debe especificar el parámetro `-w` para modificar el valor sobre la marcha:

```
# sysctl -w net.ipv4.conf.all.arp_announce=1
```

7. Esta modificación debe ser definitiva. Modifique el archivo `/etc/sysctl.conf` y vuelva a cargarlo:

```
# vi /etc/sysctl.conf
```

Añada:

```
net.ipv4.conf.all.arp_announce = 1
```

Y guarde. Vuelva a cargar los nuevos valores:

```
# sysctl -p
```

## 4. Recompilación del núcleo

Objetivo: recompilar el núcleo proporcionado con su distribución.

1. Con el gestor de paquetes de su distribución, instale las fuentes de su núcleo. El paquete se llama por lo general `kernel-source` o `linux-source`. Utilice APT, Yum, zypper o una herramienta gráfica que gestione las dependencias.

2. Dirijase a `/usr/src/linux` y teclee `make oldconfig`. Eso tiene como efecto que configura el archivo `.config` de forma idéntica a la configuración actual.

```
# cd /usr/src/linux  
# make oldconfig
```

3. Inicie la configuración texto o gráfico, según su entorno de trabajo, con:

```
# make menuconfig
```

o:

```
# make xconfig
```

Modifique, si lo desea, una opción, por ejemplo para hacer corresponder las optimizaciones del núcleo a su procesador.

4. Inicie la compilación con el comando **make**. Veamos un truco que puede usar si dispone de una máquina potente: `make` puede simultanear, cuando es posible, la compilación tratando varios archivos a la vez. Es lo ideal con una máquina multiprocesador (SMP, multicore, HyperThreading, etc.). Utilice la opción `-j` seguida del número de compilaciones simultáneas:

```
# make -j 2
```



5. Cuando esté terminada la instalación, instale los diferentes componentes, los módulos primero, luego el núcleo:

```
# make modules_install
# make install
```

Esta última etapa debería crear el initrd y modificar la configuración de GRUB.

6. Verifique en /boot/grub/menu.lst (GRUB) o en /boot/grub/grub.cfg (GRUB2) la presencia del nuevo núcleo, y vuelva a iniciar su máquina para cargar el núcleo.

## 5. Los periféricos y el hardware

Objetivo: manejar los periféricos de cualquier tipo y obtener la información relativa a su hardware.

1. Vaya a /dev y liste todos los periféricos de tipo sd\* o hd\*, según el controlador de sus discos. Liste los números mayor y menor. El número mayor es el mismo para un tipo dado. ¿Qué observa para el número menor?

En el caso de un disco SCSI, sólo puede haber 15 particiones, que se numeran para cada disco de sda1 a sda15, sdb1 a sdb15, y así sucesivamente. El número menor del disco es un múltiplo de 16: 0, 16, 32, 48, etc. El número menor de la partición va de 1 a 15, luego de 17 a 31, y así sucesivamente.

2. Aísle la dirección física de su tarjeta gráfica en el bus PCI. Las tarjetas AGP y PCI Express se consideran como si fueran un bus PCI.

```
# lspci | grep -i vga
01:00.0 VGA compatible controller: nVidia Corporation GeForce
8600 GT (rev a1)
```

3. Obtenga más detalles de esta tarjeta. En particular, ¿qué módulo la gestiona? Se puede obtener esta información con `-v` al especificar únicamente la tarjeta `-s` de `lspci`:

```
# lspci -s 01:00.0 -v
01:00.0 VGA compatible controller: nVidia Corporation GeForce
8600 GT (rev a1) (prog-if 00 [VGA controller])
    Subsystem: ASUSTeK Computer Inc. Device 8243
    Flags: bus master, fast devsel, latency 0, IRQ 16
    Memory at fd000000 (32-bit, non-prefetchable) [size=16M]
...
Kernel driver in use: nvidia
Kernel modules: nvidiafb, nvidia
```

4. Liste la información de su procesador. ¿Podría ver si dispone de varios procesadores, o de varios cores, o del HyperThreading?

```
# cat /proc/cpuinfo
```

Para saber si tiene varias CPU, mire los valores de "processor", "cpu cores" y "core id". processor indica el número de procesador. Si hay varios, dispone de varios procesadores, SMP clásico, multicore o HyperThreading.

Si muestra un valor para "cpu cores" y es superior a 1, su procesador dispone de varios cores, como los Athlons X2 o los Intel Core2 Duo o Quad. En este caso, el "core id" identifica el core actual.

Para saber si su máquina gestiona el HyperThreading, mire en los "flags" si "ht" está presente. Si es el caso, en principio el HyperThreading está presente.

5. Conecte un dispositivo USB en su PC. Si está en entorno gráfico, puede que el gestor de archivos se abra. ¿Qué mecanismos se ponen en marcha?

En un primer tiempo, el núcleo detecta la conexión y carga el módulo USB correspondiente.

Se ha creado el periférico básico y se ha generado un evento.

El servicio udev detecta el evento y ejecuta las reglas asociadas: modificación, por ejemplo, de los permisos sobre el periférico.

Para la interfaz gráfica, hay otro servicio: hal (hardware abstraction layer) es un bus de comunicación entre los diversos elementos. También intercepta el evento y ejecuta otras reglas, esta vez en el espacio del usuario: abre el gestor de archivos.

## 1. 6. systemd

En este ejercicio práctico, verá como systemd puede remplazar una tarea crontab para ejecutar un servicio (un job por ejemplo) en un periodo de tiempo dado.

1. Cree un script llamado /usr/local/bin/timer.sh que añada una línea con la fecha y la hora actuales en el archivo /var/log/timer.log.

Se trata de un script muy simple:

```
#!/usr/bin/bash  
date >> /var/log/timer.log
```

Hágalo ejecutable:

```
# chmod +x /usr/local/bin/timer.sh
```

Compruebe que funciona correctamente ejecutándolo:

```
# /usr/local/bin/timer.sh  
# /usr/local/bin/timer.sh  
# cat /var/log/time.log  
jue abr 29 23:45:25 CEST 2021  
jue abr 29 23:45:27 CEST 2021
```

2. Cree un servicio systemd llamado mytimer que se encargará de ejecutar el script. El archivo se creará en /etc/systemd/system/mytimer.service y será de tipo oneshot: systemd ejecutará el comando una sola vez, al arranque, sin controlar el servicio. El objetivo será multi-user.target:

```
[Unit]
Description=Put the current timedate in a var file

[Service]
Type=oneshot
ExecStart=/usr/local/bin/timer.sh
User=root
Group=root

[Install]
WantedBy=multi-user.target
```

3. Compruebe la ejecución del servicio:

```
# systemctl start mytimer
# systemctl start mytimer
# systemctl start mytimer
# cat /var/log/time.log
jue abr 29 23:49:28 CEST 2021
jue abr 29 23:49:35 CEST 2021
jue abr 29 23:49:56 CEST 2021
```

4. Ahora pedirá a systemd que ejecute este servicio en intervalos regulares. Un objetivo especial llamado timer le permitirá realizar esta acción. Cree un archivo llamado mytimer.timer, que ejecutará el servicio anteriormente creado cada minuto, en el mismo objetivo.

La línea OnCalendar define la periodicidad. Encontrará la documentación sobre los timers aquí: <https://www.freedesktop.org/software/systemd/man/systemd.timer.html>

```
[Unit]
Description=Run mytimer every minute

[Timer]
OnCalendar=*:0/1
Unit=mytimer.service

[Install]
WantedBy=multi-user.target
```

5. Arranque este timer:

```
# systemctl start mytimer.timer
```

6. Liste los timers activos y observe la línea correspondiente al que hemos creado:

```
# systemctl list-timers
NEXT          LEFT    LAST          PASSED
UNIT          ACTIVATES
Thu 2020-02-06 22:01:00 CET 15s left  Thu 2020-02-06 22:00:05 CET 38s ago
mytimer.timer  mytimer.service
...
```

7. Eche un vistazo al archivo `/var/log/timer.log`:

```
jue abr 29 23:54:10 CEST 2021  
jue abr 29 23:55:05 CEST 2021  
jue abr 29 23:56:32 CEST 2021  
jue abr 29 23:57:27 CEST 2021
```

8. Como puede comprobar, aproximadamente, cada minuto el servicio se ejecuta. El "aproximadamente" es importante. La documentación le explicará que el OnCalendar no ejecutará precisamente su comando en el momento exacto, ya que su ajuste depende de un parámetro de precisión llamado AccuracySec, que es de un minuto por defecto. Systemd tiene, por lo tanto, un minuto por defecto a partir de la fecha prevista para ejecutar su comando.