**Documentación del LFS**

**- Capítulo 1:**

Descargar el Ubuntu, crear la máquina virtual y reservar el disco.

Lectura del libro.

Entre la creación de la maquina e instalación del Ubuntu se tarda 30 min aproximadamente.

1. Problemas encontrados.

La instalación tardo mucho y muchas veces tuve que hacer de vuelta.

**- Capítulo 2:**

Creación de la nueva partición para el lfs y el swap.

Desde root instalamos las herramientas de host y paquetes necesarios para construir LFS.

Verificamos si el sistema host tiene todas las versiones apropiadas y la capacidad de compilar programas.

Se realiza la configuración de la variable $LFS export LFS=/mnt/lfs donde construirá el sistema LFS.

Se creó el lfs y su directorio.

Tarda aproximadamente 15 min este capítulo y se crea la primera instantánea para poder recuperar más tarde por si hubiese necesidad.

1. Problemas encontrados.

La partición se creó mal y tuve que borrar y hacer de vuelta.

**- Capítulo 3:**

Se descargaron los paquetes, los faltantes se descargaron manualmente con wget.

Detalla qué paquetes y parches deben descargarse para construir un sistema LFS y cómo

almacenarlos en el nuevo sistema de archivos.

Se crea el directorio $LFS/sources la cual se puede utilizar como lugar para almacenar los

tarballs, paquetes, parches y también como directorio de trabajo. Así al utilizar este directorio, todos sus elementos estarían disponibles durante la construcción.

Para descargar los paquetes utilizamos wget wget-list-systemd

El capítulo toma aproximadamente 20 minutos y se crea la segunda instantánea

1. Problemas encontrados.

Algunos paquetes faltantes, pero se solucionó con el comando wget con los links de los paquetes.

**- Capítulo 4:**

Se creó una carpeta donde se almacenarán algunos programas por un cierto tiempo.

Trata sobre la construcción de un sistema temporal. Se crean un conjunto de directorios donde se instalan herramientas temporales (etc, var, tools y otros).

Como root cometer un solo error puede dañar un sistema entonces agregaremos un usuario sin privilegios llamado lfs y crearemos un entorno de compilación apropiado para ese usuario.

También habla de cuánto tiempo se tarda aproximadamente en compilar e instalar cada

paquete y sobre el conjunto de pruebas que se realizan.

El capítulo toma aproximadamente 25 minutos y se crea la tercera instantánea.

**- Capítulo 5:**

Explica la instalación de la cadena de herramientas inicial (binutils, gcc y glibc) utilizando

técnicas de compilación cruzada para aislar las nuevas herramientas del sistema host y se

instalan en el directorio $LFS/tools.

Instalación y configuración de algunos paquetes.

Como usuario lfs dentro del directorio /mnt/lfs/sources/ utilizamos tar -xvf paquete para

extraer el paquete a construir.

Con cd Cambiamos al directorio creado cuando se extrajo el paquete, se siguen los pasos del libro para instalar y compilar. Luego volvemos al directorio /mnt/lfs/sources/ y eliminamos el directorio extraído con rm -Rf a menos que se indique lo contrario.

En general este capítulo llevo 1h en terminarse. Se crea la cuarta instantánea para recuperar si hubiese necesidad.

1. Problemas encontrados.

En esta parte mi notebook se congelo y tuve que pasar a hacer en la pc.

**- Capítulo 6:**

Este capítulo muestra cómo realizar una compilación cruzada de utilidades básicas utilizando la cadena de herramientas cruzada recién construida y debe realizarse como usuario lfs.

Instalación y compilación de herramientas temporales.

Luego como en el Capítulo 5 con el usuario lfs dentro del directorio /mnt/lfs/sources/

utilizamos tar -xvf paquete para extraer el paquete a construir.

Con cd cambiamos al directorio creado cuando se extrajo el paquete, se siguen los pasos del libro para instalar y compilar. Luego volvemos al directorio /mnt/lfs/sources/ y eliminamos el directorio extraído con rm -Rf.

Este capítulo toma aproximadamente 1h y 30 minutos. Y se crea la siguiente instantánea.

**- Resumen de los comandos ejecutados, sin los parámetros.**

wget

chown

groupadd

useradd

passwd

exit

su

ls

cat

cfdisk

fdisk

mkfs

mkswap

cd

export

mkdir

mount

chmod

source

Consideraciones a tener en cuenta:

- Ingresar como usuario root, tomar capturas si es posible de cada sección de los capítulos.

**- Capítulo 7:**

Ingresa a un entorno "chroot", donde usamos las nuevas herramientas para construir el resto de las herramientas necesarias para crear el sistema LFS.

Montar el /dev y el kernel

Para que el entorno funcione correctamente, se deben los llamados sistemas de archivos de kernel virtual, que se montarán antes de ingresar al entorno chroot. Si se apaga la maquina o mejor dicho si es posible siempre se debe verificar que estén montados los sistemas de archivos de este capítulo con el comando findmnt | grep $LFS, ya que es importante para los siguientes capítulos.

chroot "$LFS" /usr/bin/env -i \

HOME=/root \

TERM="$TERM" \

PS1='(lfs chroot) \u:\w\$ ' \

PATH=/usr/bin:/usr/sbin MAKEFLAGS="-j4"\

/bin/bash –login

Se le agrego el makeflags para que al compilar e instalar utilice más recursos y sea más rápido.

Se crean directorios, archivos esenciales y enlaces simbólicos.

Se extraen paquetes con tar -xvf paquete y luego instalamos.

También se realiza una limpieza para liberar un poco de espacio y un respaldo por cualquier

problema que ocurra

Este capítulo se logra terminar en aproximadamente 50 min. Se crea una nueva instantánea.

**- Capítulo 8:**

Se construye el sistema LFS completo. Se instalan paquetes con tar -xvf paquete, se compilan y después se realizan pruebas de cada paquete, es importante verificar que no obtengamos errores

ya que es un capítulo importante para el sistema. Algunos errores el libro avisa la cual podrías ignorar, pero hay otros que se debe buscar la manera de solucionar. Lo malo de este capítulo es que te lleva como varios días ya que son más de 70 paquetes por instalar y hay algunos que tardan bastante, uno de ellos es GCC.

Se limpian algunos archivos adicionales sobrantes de la ejecución de pruebas y otros que ya no se necesitan.

Desde el 8.1 hasta el 8.5 tarda 1 hora entre compilar configurar e instalar. Instantánea creada.

Desde el 8.6 hasta el 8.26 tarda 1 hora y media. Otra instantánea creada.

El 8.27 es el más largo y toma aproximadamente 7 horas. Otra instantánea.

Desde el 8.28 al 46 son 2 horas y media. Otra instantánea.

Del 47 al 8.80 son 3 horas. Otra instantánea.

Y los últimos (8.83) tardan 2 horas en completar. Otra instantánea.

**- Resumen comandos ejecutados, sin los parámetros.**

tar

mv

rm

cat

find

pushd

popd

echo

cd

mkdir

make

make install

configure

ln

ls

sed

chown

mount

chroot

install

chgrp

exec

cp

mountpoint

umount

exit

**- Capítulo 9:**

Este capítulo trata sobre los archivos de configuración general necesarios para configurar la

red, analizan los problemas que afectan la configuración adecuada de los dispositivos,

configuración del reloj del sistema y la disposición del teclado, breve introducción a los scripts

y archivos de configuración utilizados cuando el usuario inicia sesión en el sistema (Creación

del archivo /etc/inputrc y /etc/shells), configuración del comportamiento de systemd.

Esta configuración toma alrededor de 1 hora pero me tomo más de 3 horas en completar.

**- Capítulo 10:**

Crear el archivo fstab.

Instalar el grub.

Agregar el dualBoot.

La construcción de un núcleo para el nuevo sistema LFS donde extraemos el paquete,

compilamos y configuramos el kernel a través de una interfaz con el comando make

menuconfig donde habilitas o deshabilitas las funciones para que el sistema funcione. El

comando que me ayudo bastante es make defconfig ya que te establece una configuración base.

También realizamos la instalación del cargador de arranque GRUB para que el sistema LFS

pueda seleccionarse para arrancar al inicio. Es importante leer bien las Convenciones de

nomenclatura de GRUB para que puedas crear el archivo de configuración de grub.

El capítulo 10 me tomo 30 minutos.

1. Problemas encontrados.

No se encontraba el disco y otros errores, pero se

soluciona al generar automáticamente el archivo grub del lfs en el grub Ubuntu.

**- Capitulo 11:**

LFS instalado. Se crea un /etc/lfs-release archivo para saber qué versión de LFS está instalada

en el sistema.

Salimos del entorno chroot y luego desmontamos los sistemas de archivos virtuales,

desmontar particiones y el sistema de archivos LFS. Luego se reinicia y todo LISTO.

Como ultimo podemos descargar wget y gnutls en el lfs porque sería de mucha importancia.

Este capítulo me tomo 20 minutos.

**Anexo anotaciones.**

127.0.0.2 puerto 2222 putty

192.168.100.214

10.0.2.15

127.0.0.1 2222

/dev/sdb1 /mnt/lfs ext4 defaults 1 1

export LFS=/mnt/lfs

export MAKEFLAGS="-j4"

dentro de profile

echo 'python3 /home/lfs/grupo2-shell/main.py' >> /etc/profile

chroot "$LFS" /usr/bin/env -i \

HOME=/root \

TERM="$TERM" \

PS1='(lfs chroot) \u:\w\$ ' \

PATH=/usr/bin:/usr/sbin MAKEFLAGS="-j4"\

/bin/bash --login

cd /sources/

capitulo 8 parte8.1 al 8.5

FAIL: io/tst-lchmod

FAIL: nptl/tst-cond24

FAIL: nptl/tst-mutex10

FAIL: stdlib/test-bz22786

FAIL: stdlib/test-cxa\_atexit-race2

¨

!

"·$$&=)/(&¨Ç\*^¨Ñªªºººº\\ººº

grep 'ALL DONE'

LC\_ALL=es\_PY locale charmap

LC\_ALL=es\_MX.ISO-8859-1 locale language

LC\_ALL=es\_MX.ISO-8859-1 locale charmap

LC\_ALL=es\_MX.ISO-8859-1 locale int\_curr\_symbol

LC\_ALL=es\_MX.ISO-8859-1 locale int\_prefix

/mnt/lfs/boot/vmlinuz-6.4.12-lfs-12.0-systemd

/usr/lib/needrestart/vmlinuz-get-version

/boot/vmlinuz.old

/boot/vmlinuz-5.15.0-91-generic

/boot/vmlinuz-5.15.0-92-generic

/boot/vmlinuz

lsof | grep '/mnt/lfs'

kill -9 1017

vi /boot/grub/grub.cfg

cat > /boot/grub/grub.cfg << "EOF"

# Begin /boot/grub/grub.cfg

set default=0

set timeout=5

menuentry "GNU/Linux, Linux 6.4.12-lfs-12.0-systemd" {

insmod ext2

set root=(hd1,1)

linux /boot/vmlinuz-6.4.12-lfs-12.0-systemd root=/dev/sdb1 ro

}

menuentry "Ubuntu" {

insmod part\_gpt

insmod ext2

set root='hd0,gpt2'

linux /vmlinuz-5.15.0-91-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro

initrd /initrd.img-5.15.0-91-generic

}

EOF

cat > /etc/lsb-release << "EOF"

DISTRIB\_ID="Linux From Scratch"

DISTRIB\_RELEASE="12.0-systemd"

DISTRIB\_CODENAME="GUSLFS"

DISTRIB\_DESCRIPTION="Linux From Scratch"

EOF

cat > /etc/os-release << "EOF"

NAME="Linux From Scratch"

VERSION="12.0-systemd"

ID=lfs

PRETTY\_NAME="Linux From Scratch 12.0-systemd"

VERSION\_CODENAME="GUSLFSV1"

EOF

root@gustavoserver:/home/gustavo007pc# ls /boot

System.map-5.15.0-91-generic initrd.img-5.15.0-92-generic

System.map-5.15.0-92-generic initrd.img.old

config-5.15.0-91-generic lost+found

config-5.15.0-92-generic vmlinuz

grub vmlinuz-5.15.0-91-generic

initrd.img vmlinuz-5.15.0-92-generic

initrd.img-5.15.0-91-generic vmlinuz.old

root@gustavoserver:/home/gustavo007pc#

root@gustavoserver:/home/gustavo007pc# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS

loop0 7:0 0 74.1M 1 loop /snap/core22/1033

loop1 7:1 0 152.1M 1 loop /snap/lxd/26200

loop2 7:2 0 40.4M 1 loop /snap/snapd/20671

loop3 7:3 0 148.5M 1 loop /snap/lxd/26715

sda 8:0 0 10G 0 disk

├─sda1 8:1 0 1M 0 part

├─sda2 8:2 0 1.8G 0 part /boot

└─sda3 8:3 0 8.2G 0 part

└─ubuntu--vg-ubuntu--lv 253:0 0 8.2G 0 lvm /

sdb 8:16 0 35G 0 disk

├─sdb1 8:17 0 30G 0 part /mnt/lfs

└─sdb2 8:18 0 4G 0 part

sr0 11:0 1 1024M 0 rom

/home/lfs

System.map-5.15.0-91-generic initrd.img-5.15.0-92-generic

System.map-5.15.0-92-generic initrd.img.old

config-5.15.0-91-generic lost+found

config-5.15.0-92-generic vmlinuz

grub vmlinuz-5.15.0-91-generic

initrd.img vmlinuz-5.15.0-92-generic

initrd.img-5.15.0-91-generic vmlinuz.old

root@gustavoserver:/home/gustavo007pc#

root@gustavoserver:/home/gustavo007pc# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS

loop0 7:0 0 74.1M 1 loop /snap/core22/1033

loop1 7:1 0 152.1M 1 loop /snap/lxd/26200

loop2 7:2 0 40.4M 1 loop /snap/snapd/20671

loop3 7:3 0 148.5M 1 loop /snap/lxd/26715

sda 8:0 0 10G 0 disk

├─sda1 8:1 0 1M 0 part

├─sda2 8:2 0 1.8G 0 part /boot

└─sda3 8:3 0 8.2G 0 part

└─ubuntu--vg-ubuntu--lv 253:0 0 8.2G 0 lvm /

sdb 8:16 0 35G 0 disk

├─sdb1 8:17 0 30G 0 part /mnt/lfs

└─sdb2 8:18 0 4G 0 part

sr0 11:0 1 1024M 0 rom

Filesystem Size Used Avail Use% Mounted on

tmpfs 681M 1.1M 680M 1% /run

/dev/mapper/ubuntu--vg-ubuntu--lv 8.1G 3.8G 3.9G 50% /

tmpfs 3.4G 0 3.4G 0% /dev/shm

tmpfs 5.0M 0 5.0M 0% /run/lock

/dev/sdb1 30G 5.3G 23G 19% /mnt/lfs

/dev/sda2 1.7G 251M 1.4G 16% /boot

tmpfs 681M 4.0K 681M 1% /run/user/1000

Disk model: VBOX HARDDISK

Units: sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disklabel type: gpt

Disk identifier: 21152E3F-8BA8-4575-A248-68F00BE40BB8

Device Start End Sectors Size Type

/dev/sda1 2048 4095 2048 1M BIOS boot

/dev/sda2 4096 3674111 3670016 1.8G Linux filesystem

/dev/sda3 3674112 20969471 17295360 8.2G Linux filesystem