

基于 deepin 的操作系统构建

第 15 课 构建 Linux 操作系统 - 制作镜像

徐小东

统信软件

2022-11-14

课程目录

- ① 制作镜像
- ② 补充阅读材料
- ③ 参考资料

Section 1

制作镜像

制作镜像

- ▶ 目的
为了便于安装已经构建好的系统
- ▶ 应用
通过 `dd` 命令直接将其刷写到磁盘，如：优盘、硬盘
- ▶ 格式
以 `.img` 镜像格式为例

准备虚拟磁盘

► TIP: 下列操作在宿主系统以 root 帐号执行

计算虚拟磁盘的大小

```
SIZE="$(du -sm "$LFS" | awk '{print $1}')"
EXTRA_SIZE=500
REAL_SIZE="$((SIZE+EXTRA_SIZE))"
IMGNAME="lfs-11.1-systemd.img"
```

创建虚拟磁盘文件

```
truncate -s "${REAL_SIZE}M" "$IMGNAME"
```

对虚拟磁盘分区及格式化

确定可用的 loop 设备，并附加虚拟磁盘文件

```
LDEV="$(losetup -f)"
```

```
losetup "$LDEV" "$IMGNAME"
```

对虚拟磁盘分区

此处分为一个区：

1: /

可按需要调整

```
parted -s "$LDEV" mktable msdos
```

```
parted -s "$LDEV" mkpart primary ext4 2048s 100%
```

```
parted -s "$LDEV" toggle 1 boot
```

```
# 更新分区信息  
partprobe $LDEV
```

```
# 格式化分区  
mkfs.ext4 -L ROOT_LFS "${LDEV}p1"
```

挂载虚拟磁盘分区，并拷贝 rootfs

```
# 挂载虚拟磁盘分区
TARGET=/home/deepin/target
mkdir $TARGET
mount "${LDEV}p1" $TARGET

# 拷贝 rootfs
cp -a "$LFS"/* $TARGET
```

使用 GRUB 设定引导过程

► 将 GRUB 文件安装到 `/boot/grub` 并设定引导磁道：

```
# 查询虚拟磁盘设备
```

```
echo $LDEV  
/dev/loop0
```

```
# 进入 chroot 环境
```

```
./chroot.sh
```

```
# 安装
```

```
grub-install -d /usr/lib/grub/i386-pc <$LDEV 的值>
```

► 创建 GRUB 配置文件:

```
cat > /boot/grub/grub.cfg << "EOF"
# Begin /boot/grub/grub.cfg
set default=0
set timeout=5

insmod ext2
set root=(hd0,1)

menuentry "GNU/Linux, Linux 5.16.9-lfs-11.1" {
    linux /boot/vmlinuz-5.16.9-lfs-11.1 root=/dev/sda1 ro
}
EOF
```

清理

- ▶ 至此，已制作完毕磁盘镜像，其文件名为
lfs-11.1-systemd.img

```
exit  
umount target  
losetup -d "$LDEV"  
rm -r target
```

引导镜像

安装 QEMU

```
sudo apt install qemu-system-x86
```

运行

```
qemu-system-x86_64 -m 1024 -drive \  
    file=lfs-11.1-systemd.img,format=raw
```

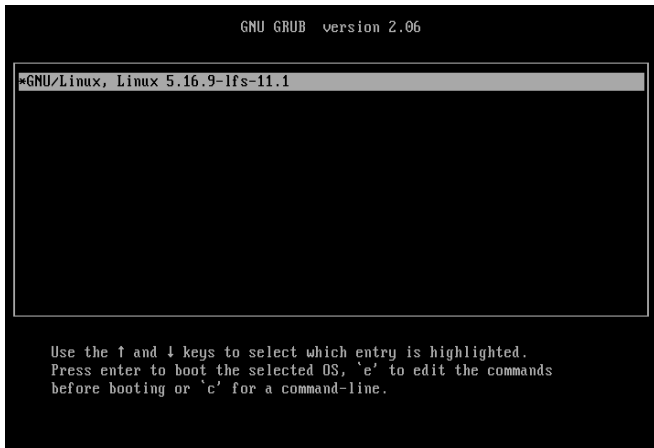


图 1: GRUB 菜单

```
[ OK ] Started Daily Cleanup of Temporary Directories.
[ OK ] Reached target System Time Set.
[ OK ] Reached target Timer Units.
[ OK ] Listening on D-Bus System Message Bus Socket.
[ OK ] Reached target Socket Units.
[ OK ] Reached target Basic System.
[ OK ] Started D-Bus System Message Bus.
[ OK ] Started Getty on tty1.
[ OK ] Reached target Login Prompts.
        Starting Name Service Cache Daemon...
        Starting User Login Management...
[ OK ] Started Name Service Cache Daemon.
[ OK ] Started Network Name Resolution.
[ OK ] Reached target Network.
[ OK ] Reached target Host and Network Name Lookups.
[ OK ] Started User Login Management.
[ OK ] Reached target Multi-User System.
[ OK ] Reached target Graphical Interface.
        Starting Record Runlevel Change in UTMF...
[ OK ] Finished Record Runlevel Change in UTMF.

Linux From Scratch 11.1-systemd
Kernel 5.16.9 on an x86_64 (-)

lfs login:
```

图 2: Systemd 引导

```
total 0
-bash-5.1# cd /
-bash-5.1# ls -l
total 60
lrwxrwxrwx 1 root root 7 Jul 25 15:37 bin -> usr/bin
drwxr-xr-x 3 root root 4096 Nov 2 2022 boot
drwxr-xr-x 14 root root 2940 Nov 2 03:16 dev
drwxr-xr-x 22 root root 4096 Nov 2 02:24 etc
drwxr-xr-x 2 root root 4096 Aug 4 11:04 home
lrwxrwxrwx 1 root root 7 Jul 25 15:37 lib -> usr/lib
drwxr-xr-x 2 root root 4096 Aug 2 16:05 lib64
drwx----- 2 root root 16384 Nov 1 14:52 lost+found
drwxr-xr-x 4 root root 4096 Aug 4 11:04 media
drwxr-xr-x 2 root root 4096 Aug 4 11:04 mnt
drwxr-xr-x 2 root root 4096 Aug 4 11:04 opt
dr-xr-xr-x 109 root root 0 Nov 2 2022 proc
drwxr-xr-x 2 root root 4096 Nov 2 03:15 root
drwxr-xr-x 12 root root 300 Nov 2 03:16 run
lrwxrwxrwx 1 root root 8 Jul 25 15:37/sbin -> usr/sbin
drwxr-xr-x 2 root root 4096 Aug 4 11:04 srv
dr-xr-xr-x 12 root root 0 Nov 2 2022 sys
drwxrwxrwt 9 root root 180 Nov 2 03:16 tmp
drwxr-xr-x 10 root root 4096 Oct 21 14:34 usr
drwxr-xr-x 10 root root 4096 Nov 2 02:24 var
-bash-5.1#
```

图 3: 使用系统

将镜像刷写到磁盘

- ▶ 将镜像刷写到磁盘（如：优盘）上，以便用于物理机

```
dd if=lfs-11.1-systemd.img of=/dev/sda \  
    bs=4M status=progress
```

重新调整 / 分区

- ▶ 当 img 镜像被刷写到磁盘上时，可对 / 分区进行调整，以便使用完整的物理磁盘空间

设置目标块设备

```
TARGET_DEV="/dev/sda"
```

```
TOTAL="$(parted -m "$TARGET_DEV" unit s print |  
grep "$TARGET_DEV" | cut -d ':' -f 2 | tr -d 's')"
```

```
ROOT_PART_END="$((TOTAL-1))"
```

```
ROOT_PART_NUM=2
```

```
parted -s "$TARGET_DEV" unit chs resizepart \  
    "$ROOT_PART_NUM" "$ROOT_PART_END"  
e2fsck -f -y "${TARGET_DEV}${ROOT_PART_NUM}"  
resize2fs "${TARGET_DEV}${ROOT_PART_NUM}"
```

Section 2

补充阅读材料

补充阅读材料

- ▶ 《精通 Linux》
 - 第 3 章 设备管理
 - 第 4 章 磁盘和文件系统

Section 3

参考资料

参考资料

- ▶ Parted User's Manual
- ▶ e2fsck
- ▶ resize2fs
- ▶ GNU GRUB Manual
- ▶ chroot.sh