Clone Linux System

Overview

This page assumes that system is installed in EFI mode having: a 200-500MB vfat/fat32 “EFI system” partition.

Get Source Image

Create a live USB and boot system from USB. Once booted into the live-cd, mount the source filesystem. First check the correct device with lsblk

```
# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda      8:0    0 931,5G  0 disk
├─sda1   8:1    0   512M  0 part /boot/efi
├─sda2   8:2    0  18,6G  0 part /
└─sdb3   8:3    0   9,8G  0 part [SWAP]
sdb      8:0    0 931,5G  0 disk
├─sdb1   8:1    0   512M  0 part /boot/efi
├─sdb2   8:2    0  18,6G  0 part /
├─sdb3   8:3    0   9,8G  0 part [SWAP]
└─sdb4   8:4    0 902,6G  0 part /media/storage
```

We need to copy the **“/boot/efi”** and **“/“** partitions to the new system.

Write Target System

Create a live USB and boot system from USB. Once booted into the live-cd, mount your destination filesystem.

Mount the partition your broken Linux installation is on. If you are not sure which it is, launch GParted (included in the Live CD) and find out. It is usually a EXT4 Partition. Replace the XY with the drive letter, and partition number, for example: sudo mount /dev/sda1 /mnt.

```
# mount /dev/sdXY /mnt
```

Now bind the directories that grub needs access to to detect other operating systems, like so.

```
# mount --bind /dev /mnt/dev
# mount --bind /dev/pts /mnt/dev/pts
# mount --bind /proc /mnt/proc
```

```
# mount --bind /sys /mnt/sys
```

Internet access For internet access inside chroot:

```
# mv /mnt/etc/resolv.conf /mnt/etc/resolv.conf.org
# cp /etc/resolv.conf /mnt/etc/resolv.conf
```

Now we jump into that using chroot.

```
# chroot /mnt
```

Now install, check, and update grub. This time you only need to add the drive letter (usually a) to replace X, for example: grub-install /dev/sda, grub-install –recheck /dev/sda.

```
# grub-install /dev/sdX
# grub-install --recheck /dev/sdX
```

Alternatively, in case of persistent problems, you can purge and reinstall grub2, make new config files:

```
apt-get remove --purge grub-pc grub-common
apt-get install grub-pc
grub-mkconfig
update-grub
grub-install /dev/sda
```

Now grub is back, all that is left is to exit the chrooted system and unmount everything:

```
# exit
# umount /mnt/sys
# umount /mnt/proc
# umount /mnt/dev/pt
# umount /mnt/dev
# umount /mnt
```

Shut down and turn your computer back on, and you will be met with the default Grub2 screen.