

# GRUB 2

## Grub Configuration

Your own GRUB settings are stored in the `/etc/default/grub` file. Edit this file to change GRUB2's settings. Scripts are also located in the `/etc/grub.d/` directory. For example, on Ubuntu, there are scripts here that configure the default theme. There's also an `os-prober` script that checks the system's internal hard drives for other installed operating systems — Windows, other Linux distributions, Mac OS X, and so on — and automatically adds them to GRUB2's menu.

When you run the `update-grub` command, GRUB automatically combines the settings from the `/etc/default/grub` file, the scripts from the `/etc/grub.d/` directory, and everything else, creating a `/boot/grub/grub.cfg` file that's read at boot.

In other words, to customize your GRUB2 settings, you'll have to edit the `/etc/default/grub` file and then run the `update-grub` command. Subsequently install/update the Grub loader in the disk.

```
# /etc/default/grub
# update-grub
# grub-install /dev/sdX
# grub-install --recheck /dev/sdX
```

## Repair, Restore, or Reinstall Grub 2 with Live USB

Grub 2 typically gets overridden when you install Windows or another Operating System. To make Linux control the boot process, you need Reinstall (Repair/Restore) Grub using a Live CD.

Create a live USB and boot system from USB

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
```

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
```

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.

From:  
<https://wiki.oscardegroot.nl/> - **HomeWiki**

Permanent link:  
<https://wiki.oscardegroot.nl/doku.php?id=linux:debian:grub&rev=1566497580>

Last update: **2022/01/15 11:38**

