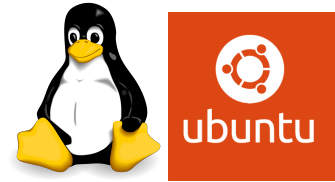


# Linux Bootloader



## Inhalt

### **1. Bootvorgang**

- 1.1 Arten von Firmware
  - i. BIOS (Legacy)
  - ii. UEFI
- 1.2 Bootvorgang
  - i. BIOS-Modus
  - ii. UEFI-Modus

### **2. Linux Bootloaders**

- 2.1 Bootloaders für beide BIOS und UEFI
  - i. GRUB (GNU GRUBv1.9x)
  - ii. Syslinux
- 2.2 Bootloaders nur für UEFI
  - i. Linux Kernel EFISTUB
- 2.3 Bootloaders nur für BIOS
  - i. GRUB Legacy (v0.9x)
  - ii. LILO (LInux LOader)
  - iii. NeoGRUB

### **3. Konfiguration > GRUB**

- i. Edit GRUB configuration file
- ii. Set default OS
- ii. Hided/unhide GRUB menu
- iii. Set GRUB menu timeout
- iv. Set background image
- v. Save changes

### **4. Quelle**

# 1 - Bootvorgang

## 1.1 Arten von Firmware

### Legacy BIOS

Die Abkürzung BIOS steht für *Basic Input Output System*, zu Deutsch etwa grundlegendes Ein- und Ausgabesystem. Jeder PC hat ein BIOS

### UEFI

UEFI ist die Kurzform für *Unified Extensible Firmware Interface*, also die vereinheitlichte und erweiterbare Firmware-Schnittstelle. UEFI ist der Nachfolger des Mitte der 1970er-Jahre eingeführten BIOS.

#### **Warum UEFI?**

Als das BIOS entwickelt wurde, waren die technischen Fortschritte nicht abzusehen, etwa die Einführung der 64-Bit-Technologie. Zwar ließ sich das BIOS immer wieder an die Neuerungen anpassen, die veraltete Grundlage setzte den BIOS-Entwicklern und Hardware-Herstellern aber sehr enge Grenzen. Teilweise musste für 64-Bit-Prozessoren erheblich improvisiert werden. Das ist mit UEFI nicht mehr nötig.

#### **Was kann UEFI?**

Die Einstellungen im BIOS lassen sich zum Beispiel ausschließlich mit der Tastatur im Textmodus vornehmen, in seltenen Fällen gibt es auch eine Mausunterstützung. UEFI kann hingegen sogar eine grafische Bedienoberfläche mit vollständiger Mausbedienung bieten. Kann, muss aber nicht. Deshalb wird auch nicht bei allen Implementationen von UEFI auf die grafischen Möglichkeiten zurückgegriffen. Treiber lassen sich zudem als Modul vom UEFI nachladen. So kann etwa die Netzwerkschnittstelle genutzt werden, noch bevor Windows gestartet wird. In Verbindung mit einem UEFI-Webbrowser lässt sich somit ohne Betriebssystem im Internet surfen.

#### **Habe ich UEFI?**

Ob Ihr Mainboard UEFI hat finden Sie nur durch einen Blick in das Mainboard-Handbuch heraus. Der BIOS-Nachfolger UEFI lässt sich zudem nicht nachrüsten. Wer eventuell einen Wechsel auf UEFI in Betracht zieht, der muss sich ein Mainboard mit UEFI-Chip kaufen.

#### UEFI kann mehr

Das unterscheidet UEFI vom alten BIOS		
	BIOS	UEFI
Partitionsschema	Master Boot Record (MBR)	Master Boot Record (MBR) GUID Partition Table (GPT)
Bootet Festplatten bis zu	2,2 Terabyte	8192 Exabyte
Mausunterstützung	<input type="radio"/>	<input checked="" type="radio"/>
Netzwerk- und Internetzugriff	<input type="radio"/>	<input checked="" type="radio"/>
Maximale Bildauflösung	800 x 600 Pixel, 16 Farben	Auflösung nur begrenzt durch die Grafikkarte
Kann Programme starten, etwa einen Browser	<input type="radio"/>	<input checked="" type="radio"/>
● = ja    ○ = nein		

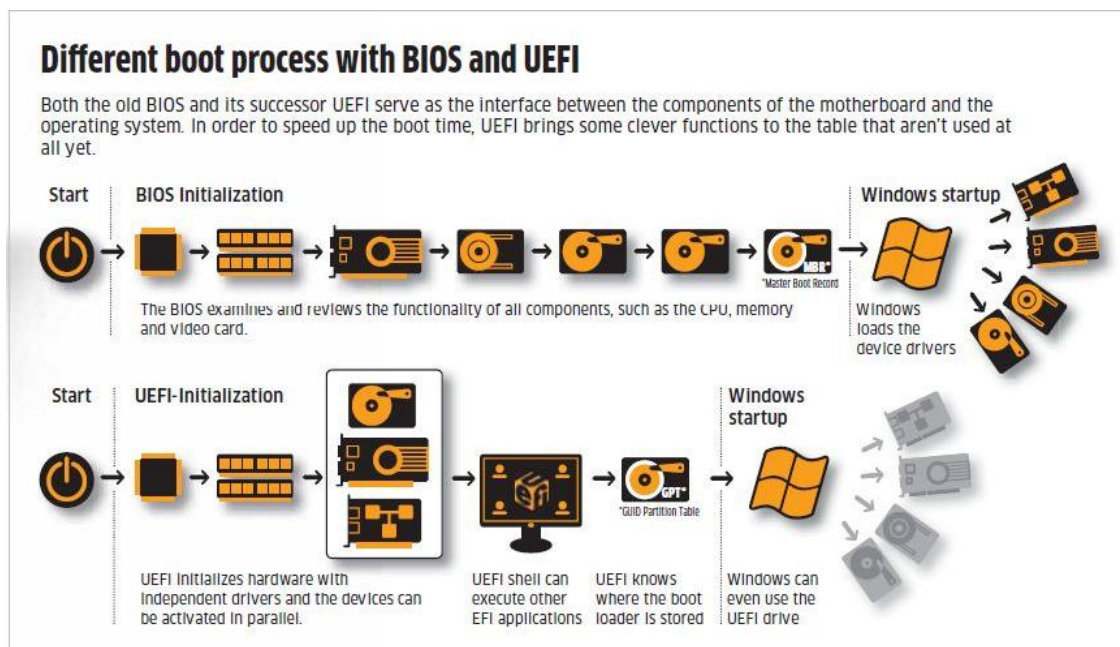
## 1.2 Bootvorgang

### BIOS-Modus

1. System switched on - Power-on self-test or POST process
2. After POST, BIOS initializes the necessary system hardware for booting (disk, keyboard controllers etc.)
3. BIOS launches the first 440 bytes (Master Boot Record) of the first disk in the BIOS disk order
4. The MBR boot code then takes control from BIOS and launches its next stage code (if any) (mostly boot loader code)
5. The launched (2nd stage) code (actual boot loader) then reads its support and config files
6. Based on the data in its config files, the boot loader loads the kernel and initial RAM-based file system (initramfs) into system memory (RAM) and launches the kernel

### UEFI-Modus

1. System switched on - Power On Self Test, or POST process.
2. UEFI firmware is loaded. Firmware initializes the hardware required for booting.
3. Firmware then reads its Boot Manager data to determine which UEFI application to be launched and from where (i.e. from which disk and partition).
4. Firmware then launches the UEFI application as defined in the boot entry in the firmware's boot manager.



# 2 - Linux Bootloaders

## 2.1 Bootloaders für beide BIOS und UEFI

### i. GRUB2 (ab GRUB 1.9x)

Grand Unified Bootloader (kurz GRUB, englisch für Großer vereinheitlichter Bootloader) ist ein freies Bootloader-Programm, das oft zum Starten von unixoiden Betriebssystemen wie z. B. Linux eingesetzt wird.

### ii. Syslinux

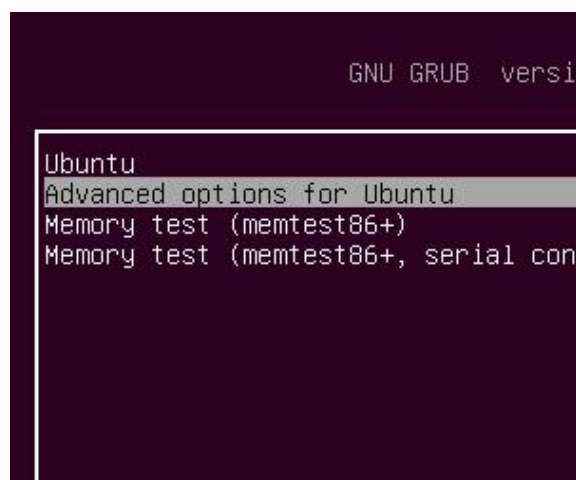
Das SYSLINUX-Projekt erstellt eine Serie schlanker Bootloader für das Booten mit dem Linux-Kernel von Hans Peter Anvin. Die bekannteste Software davon stellt ISOLINUX dar.

## 2.2 Bootloaders nur für UEFI

1. Linux Kernel EFISTUB
2. systemd-boot(Gummiboot)
3. rEFInd
4. Clover

## 2.3 Bootloaders nur für BIOS

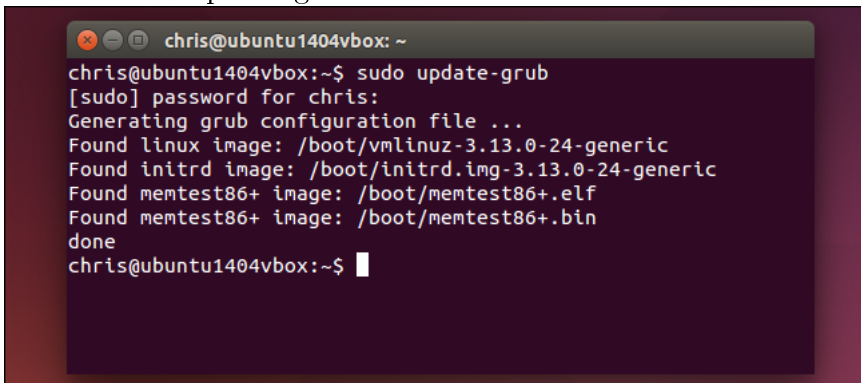
1. GRUB Legacy (v0.9x)
2. LILO (LInux LOader)
3. NeoGRUB



GRUB

## 3 - Konfiguration > GRUB

To customize your GRUB2 settings, you'll have to edit the `/etc/default/grub` file and then run the `sudo update-grub` command.

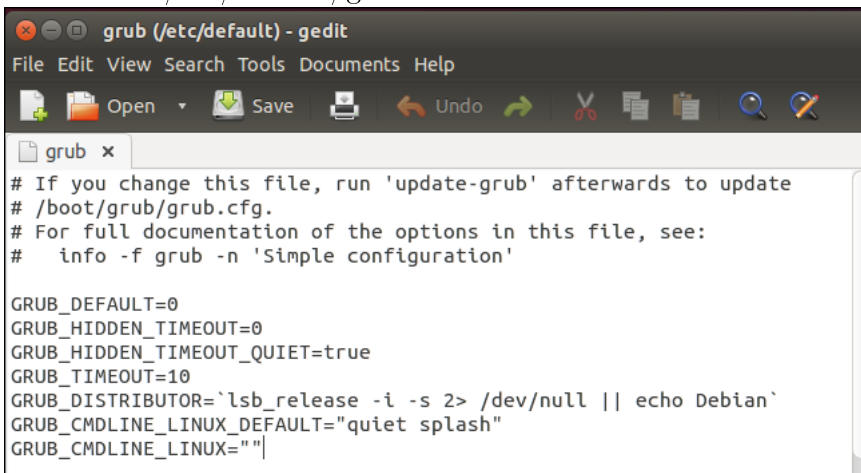


```
chris@ubuntu1404vbox: ~  
chris@ubuntu1404vbox:~$ sudo update-grub  
[sudo] password for chris:  
Generating grub configuration file ...  
Found linux image: /boot/vmlinuz-3.13.0-24-generic  
Found initrd image: /boot/initrd.img-3.13.0-24-generic  
Found memtest86+ image: /boot/memtest86+.elf  
Found memtest86+ image: /boot/memtest86+.bin  
done  
chris@ubuntu1404vbox:~$
```

### Edit the GRUB Configuration File

-gksu gedit `/etc/default/grub`

-sudo nano `/etc/default/grub`



```
grub (/etc/default) - gedit  
File Edit View Search Tools Documents Help  
Open Save Undo  
grub x  
# If you change this file, run 'update-grub' afterwards to update  
# /boot/grub/grub.cfg.  
# For full documentation of the options in this file, see:  
# info -f grub -n 'Simple configuration'  
  
GRUB_DEFAULT=0  
GRUB_HIDDEN_TIMEOUT=0  
GRUB_HIDDEN_TIMEOUT_QUIET=true  
GRUB_TIMEOUT=10  
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`  
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"  
GRUB_CMDLINE_LINUX=""
```

### Choose the Default OS

Change the `GRUB_DEFAULT=` line. By default, `GRUB_DEFAULT=0` uses the first entry as the default — change the number to 1 to use the second entry, 2 to use the third entry, or so on. You could also use `GRUB_DEFAULT=saved` and GRUB would automatically boot the last operating system you chose each time you boot. You can also specify a label in quotes. For example, if you had an operating system named Windows 7 (loader) in your OS list, you could use `GRUB_DEFAULT="Windows 7 (loader)"`

### Save a Default Operating System

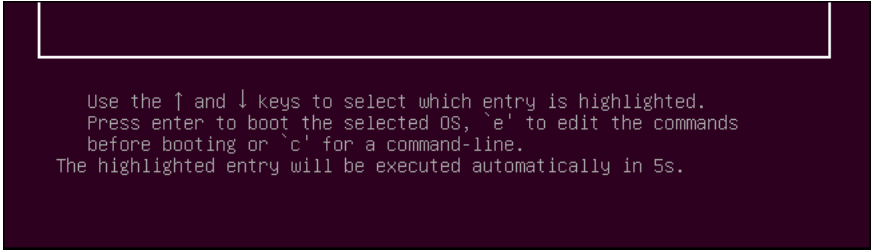
If you choose `GRUB_DEFAULT=saved`, you also need to add a `GRUB_SAVEDEFAULT=true` line — otherwise it won't work.

## Choose Whether GRUB is Hidden

With only one operating system installed, Ubuntu defaults GRUB to automatically boot to the default OS with the `GRUB_HIDDEN_TIMEOUT=0` option. This option specifies GRUB will be hidden and it will automatically boot to the default OS after 0 seconds –immediately, in other words. You can still access the menu by holding Shift as your computer boots. To set a higher timeout, use something like `GRUB_HIDDEN_TIMEOUT=5` — GRUB will display an empty screen or splash screen for five seconds, during which you can press any key to view the menu. To prevent GRUB from being automatically hidden, comment the line out — just add a `#` before it so that it reads `#GRUB_HIDDEN_TIMEOUT=0` .

## Control GRUB's Menu Timeout

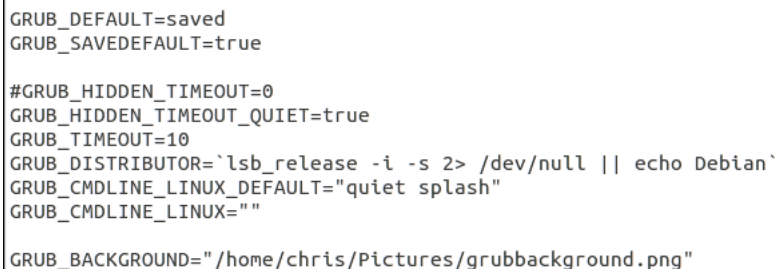
If GRUB isn't automatically hidden, you'll see the menu each time your computer boots. GRUB will automatically boot the default operating system after a period of time, usually ten seconds. During that time, you can choose another OS or leave it be to automatically boot. To change the timeout period, edit the `GRUB_TIMEOUT=10` line and enter any number of seconds you like. (Remember, this is only used if GRUB isn't hidden.) To prevent GRUB from booting automatically and always wait for you to choose an OS, change the line to `GRUB_HIDDEN=-1`



```
Use the ↑ and ↓ keys to select which entry is highlighted.  
Press enter to boot the selected OS, 'e' to edit the commands  
before booting or 'c' for a command-line.  
The highlighted entry will be executed automatically in 5s.
```

## Choose a Background Image

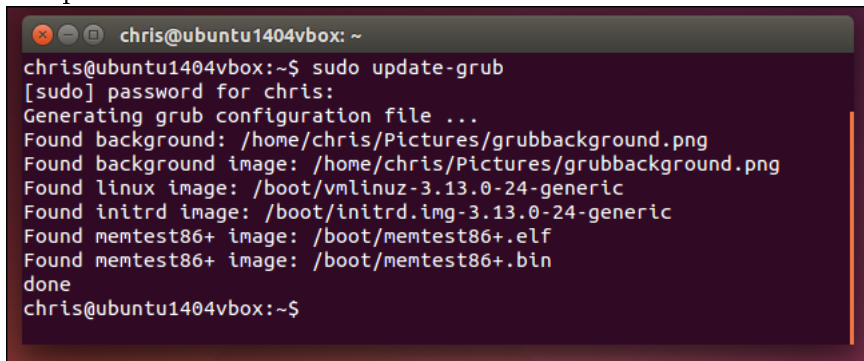
The `GRUB_BACKGROUND` line controls whether a background image is used — by default, GRUB uses a white-on-black monochrome look. You can add a line like `GRUB_BACKGROUND="/home/user/Pictures/background.png"` to specify an image file GRUB will use. The image file must meet various specifications. GRUB supports JPG/JPEG images, but these are limited to 256 colors — so you probably won't want to use a JPG image. Instead, you'll probably want to use a PNG image that can have any number of colors. You could also use a TGA image file.



```
GRUB_DEFAULT=saved  
GRUB_SAVEDefault=true  
  
#GRUB_HIDDEN_TIMEOUT=0  
GRUB_HIDDEN_TIMEOUT_QUIET=true  
GRUB_TIMEOUT=10  
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`  
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"  
GRUB_CMDLINE_LINUX=""  
  
GRUB_BACKGROUND="/home/chris/Pictures/grubbackground.png"
```

## Make Your Changes Take Effect

To have your changes take effect, simply save the text file — File > Save in Gedit or Ctrl + O and then Enter to save the file in Nano — and then run the `sudo update-grub` command. Your changes will become part of the `grub.cfg` file and will be used each time you boot your computer.

A terminal window titled 'chris@ubuntu1404vbox: ~' with a dark purple background. The window shows the execution of the 'sudo update-grub' command. The output includes the password prompt, the generation of the grub configuration file, and a list of found images: background, linux, initrd, and memtest86+. The command ends with 'done' and the prompt 'chris@ubuntu1404vbox:~\$'.

```
chris@ubuntu1404vbox:~$ sudo update-grub
[sudo] password for chris:
Generating grub configuration file ...
Found background: /home/chris/Pictures/grubbackground.png
Found background image: /home/chris/Pictures/grubbackground.png
Found linux image: /boot/vmlinuz-3.13.0-24-generic
Found initrd image: /boot/initrd.img-3.13.0-24-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
done
chris@ubuntu1404vbox:~$
```

## 4 - Quelle

<https://lakkireddymadhu.wordpress.com/author/lakkireddymadhu/page/13/>

<https://wiki.archlinux.org/>

<http://www.howtogeek.com/196655/how-to-configure-the-grub2-boot-loaders-settings/>

<http://www.com-magazin.de/praxis/uefi/bios-uefi-wissen-231367.html>

<http://www.howtogeek.com/193669/whats-the-difference-between-gpt-and-mbr-when-partitioning/>

<https://help.ubuntu.com/community/Boot-Repair>

<https://de.wikipedia.org/wiki/SYSLINUX>

[https://de.wikipedia.org/wiki/Grand\\_Unified\\_Bootloader](https://de.wikipedia.org/wiki/Grand_Unified_Bootloader)

<https://www.linux.com/learn/answers/view/1201-what-is-the-difference-between-grub-and-syslinux>

<https://www.google.de>

<https://www.google.com>