Installing Linux alongside Mac OS-X on an Apple computer

Below is a brief guide on how to set up a multi-boot Mac with Linux and OS-X, just as I did for our computers (MacBook and iMac). For us, Linux has been the primary operating system for years, and we now use OS-X only for specific purposes – and even then, less and less frequently.

Hardware and systems used

– MacBook 5.1, 8 GB RAM, 1 TB SSD; OS-X 10.6.8 Snow Leopard, OS-X 10.11.6 El Capitan, Linux Mint 21.3

– iMac 11.2, 16 GB RAM, 2 TB SSD; OS-X 10.6.8 Snow Leopard, OS-X 10.11.6 El Capitan, OS-X 10.13.6 High Sierra, LinuxMint 22.3

Please note – Disclaimer

THIS GUIDE IS PROVIDED WITHOUT ANY WARRANTY!

There is no warranty or guarantee that these instructions will work for you! You assume all risks regarding performance and stability! No responsibility is assumed for any other risks! This includes, in particular, risks such as data loss, potential damage, and other inconveniences that may result from the installation.

These instructions apply to Macs with Intel processors, using Linux Mint as an example!

Preparation

Back up your data! Create a bootable backup of your Mac system on an external hard drive, for example using CarbonCopyCloner.

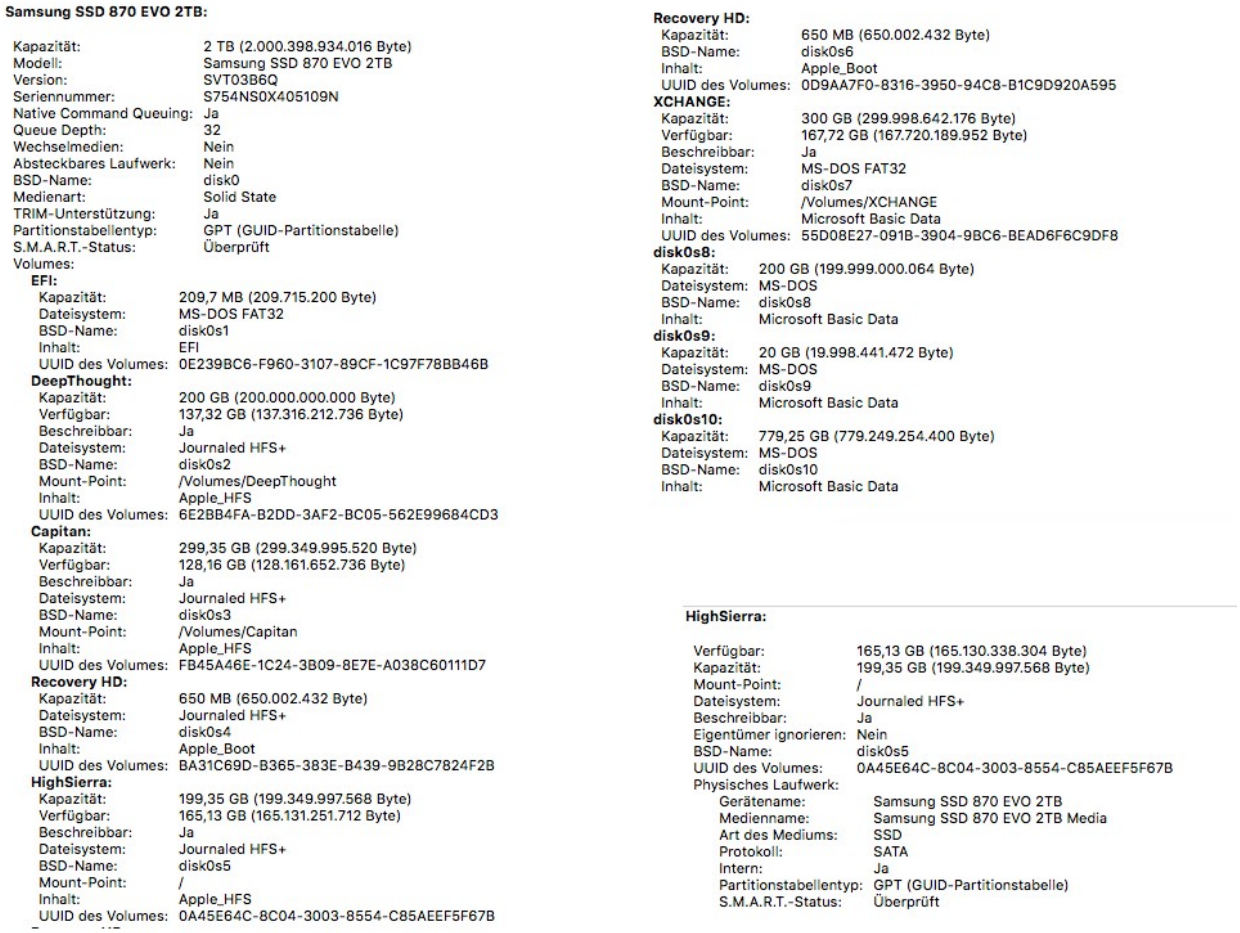
Download the desired Linux system (ISO file) and create a live system on a USB drive (instructions can be found online).
Start the computer with the clone on the external drive (hold down the Alt key while booting).

Format and partition the internal hard drive as desired using Mac Disk Utility. Warning: All data on this drive will be lost.
You need at least one partition for OS-X and one for Linux. In this guide, however, I will create separate partitions for root, home, and swap for Linux. Advantage: You can repair the system without affecting the user folders.

You should create the Mac partition as follows: Partition table type GPT (GUID Partition Table), file system Journaled HFS+.

Now create 3 more partitions at the end of the disk and format them using MS-DOS FAT32:

1) A partition for “root”; this is for the system. 100 GB to 200 GB should be enough. On my iMac, the system takes up about 40 GB.
2) A partition for “swap”; the size is usually 2 times the amount of RAM; I have 20 GB of swap space on my iMac; with 16 GB of RAM. The swap space is rarely used; it depends on what you're doing.
3) A partition for /home; this is where all your data will be stored. So make this partition as large as possible …



Samsung SSD 870 EVO 2TB:

Kapazität:	2 TB (2.000.398.934.016 Byte)
Modell:	Samsung SSD 870 EVO 2TB
Version:	SVT03B6Q
Seriennummer:	S754NS0X405109N
Native Command Queuing:	Ja
Queue Depth:	32
Wechselmedien:	Nein
Absteckbares Laufwerk:	Nein
BSD-Name:	disk0
Medienart:	Solid State
TRIM-Unterstützung:	Ja
Partitionstabellentyp:	GPT (GUID-Partitionstabelle)
S.M.A.R.T.-Status:	Überprüft
Volumes:	
EFI:	
Kapazität:	209,7 MB (209.715.200 Byte)
Dateisystem:	MS-DOS FAT32
BSD-Name:	disk0s1
Inhalt:	EFI
UUID des Volumes:	0E239BC6-F960-3107-89CF-1C97F78BB46B
DeepThought:	
Kapazität:	200 GB (200.000.000.000 Byte)
Verfügbar:	137,32 GB (137.316.212.736 Byte)
Beschreibbar:	Ja
Dateisystem:	Journaled HFS+
BSD-Name:	disk0s2
Mount-Point:	/Volumes/DeepThought
Inhalt:	Apple_HFS
UUID des Volumes:	6E2BB4FA-B2DD-3AF2-BC05-562E99684CD3
Capitan:	
Kapazität:	299,35 GB (299.349.995.520 Byte)
Verfügbar:	128,16 GB (128.161.652.736 Byte)
Beschreibbar:	Ja
Dateisystem:	Journaled HFS+
BSD-Name:	disk0s3
Mount-Point:	/Volumes/Capitan
Inhalt:	Apple_HFS
UUID des Volumes:	FB45A46E-1C24-3B09-8E7E-A038C60111D7
Recovery HD:	
Kapazität:	650 MB (650.002.432 Byte)
Dateisystem:	Journaled HFS+
BSD-Name:	disk0s4
Inhalt:	Apple_Boot
UUID des Volumes:	BA31C69D-B365-383E-B439-9B28C7824F2B
HighSierra:	
Kapazität:	199,35 GB (199.349.997.568 Byte)
Verfügbar:	165,13 GB (165.131.251.712 Byte)
Beschreibbar:	Ja
Dateisystem:	Journaled HFS+
BSD-Name:	disk0s5
Mount-Point:	/
Inhalt:	Apple_HFS
UUID des Volumes:	0A45E64C-8C04-3003-8554-C85AEEF5F67B

Recovery HD:

Kapazität:	650 MB (650.002.432 Byte)
BSD-Name:	disk0s6
Inhalt:	Apple_Boot
UUID des Volumes:	0D9AA7F0-8316-3950-94C8-B1C9D920A595

XCHANGE:

Kapazität:	300 GB (299.998.642.176 Byte)
Verfügbar:	167,72 GB (167.720.189.952 Byte)
Beschreibbar:	Ja
Dateisystem:	MS-DOS FAT32
BSD-Name:	disk0s7
Mount-Point:	/Volumes/XCHANGE
Inhalt:	Microsoft Basic Data
UUID des Volumes:	55D08E27-091B-3904-9BC6-BEAD6F6C9DF8

disk0s8:

Kapazität:	200 GB (199.999.000.064 Byte)
Dateisystem:	MS-DOS
BSD-Name:	disk0s8
Inhalt:	Microsoft Basic Data

disk0s9:

Kapazität:	20 GB (19.998.441.472 Byte)
Dateisystem:	MS-DOS
BSD-Name:	disk0s9
Inhalt:	Microsoft Basic Data

disk0s10:

Kapazität:	779,25 GB (779.249.254.400 Byte)
Dateisystem:	MS-DOS
BSD-Name:	disk0s10
Inhalt:	Microsoft Basic Data

HighSierra:

Verfügbar:	165,13 GB (165.130.338.304 Byte)
Kapazität:	199,35 GB (199.349.997.568 Byte)
Mount-Point:	/
Dateisystem:	Journaled HFS+
Beschreibbar:	Ja
Eigentümer ignorieren:	Nein
BSD-Name:	disk0s5
UUID des Volumes:	0A45E64C-8C04-3003-8554-C85AEEF5F67B
Physisches Laufwerk:	
Gerätename:	Samsung SSD 870 EVO 2TB
Mediename:	Samsung SSD 870 EVO 2TB Media
Art des Mediums:	SSD
Protokoll:	SATA
Intern:	Ja
Partitionstabellentyp:	GPT (GUID-Partitionstabelle)
S.M.A.R.T.-Status:	Überprüft

Figure 1 shows an overview of the hard drive partitioning on the iMac, which runs three versions of Mac OS-X. Linux is installed on disk0s8 (root), disk0s9 (swap), and disk0s10 (home).

The “XCHANGE” partition is used for file sharing between Mac OS-X and Linux. Data used on both systems can be stored here. FAT32, because both systems can read from and write to it. One alternative is exfat. The advantage of this is that it allows you to store files larger than 4 GB.

Once the formatting and partitioning are complete, clone the backed-up Mac OS-X back onto your computer. Mac OS-X must be installed before Linux!
Restart the Mac, and verify that everything is working and that the partitions are correct.

Installation

Now you are ready to install Linux.

Turn **off** the computer! Connect the USB drive containing the Linux live system to the Mac.

It is best to connect the computer to the Internet using an Ethernet cable. This allows any missing components (such as codes) to be installed directly.

Turn on the computer while holding down the Alt key until you see the FIE boot menu. Select the USB drive from the menu. Wait until the live system has loaded.

Now click the “Install” icon. The installer will launch and guide you through the installation process. When prompted to select an installation type, choose “Something else” (see Fig. 2).

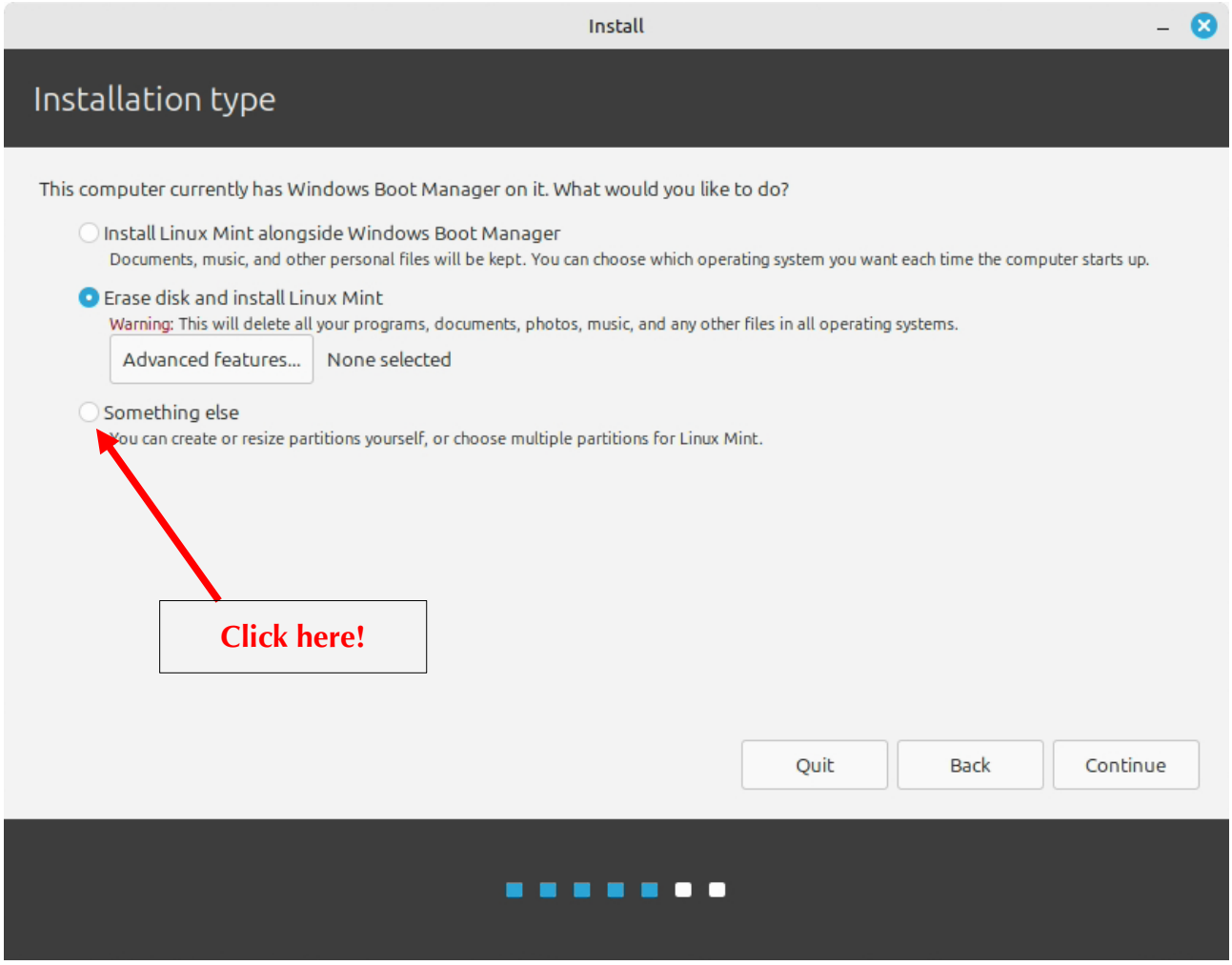


Fig. 2) Installation type

In the following window (see Fig. 3), locate the partitions you created earlier. To do so, check the size of your partitions.

– Your Mint (system) partition should be formatted as ext4 and mounted under “ **/** ”.

– Your home partition should also be formatted as ext4 and mounted under “ **/home** ”.

– Your swap partition should be formatted as swap.

– The destination for grub can be left as “ **/dev/sda** ” (assuming this corresponds to your hard drive).

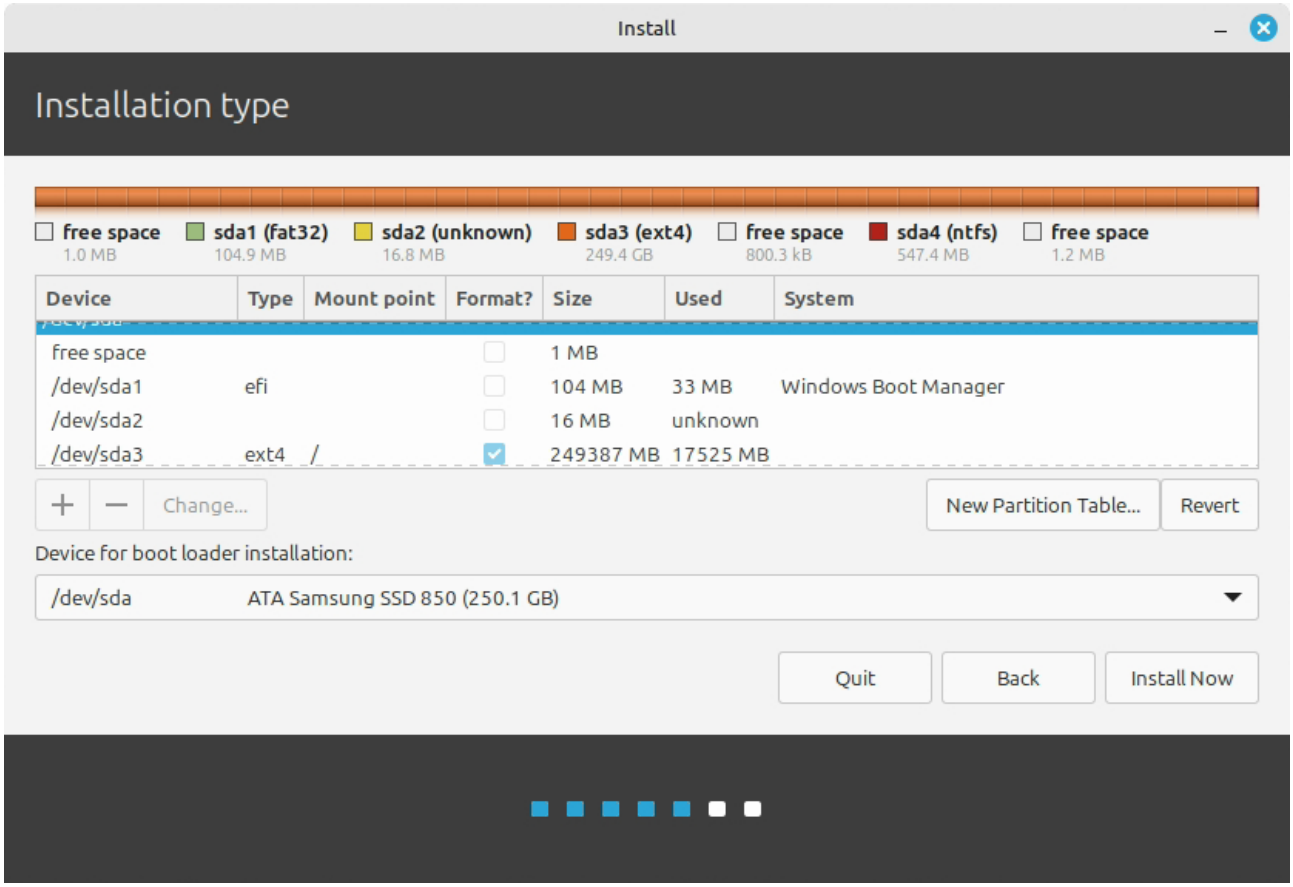


Fig. 3) The partitions

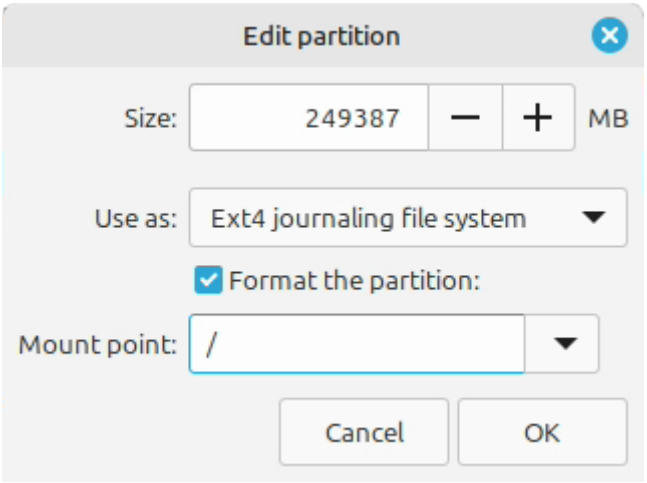


Fig. 4: Example of formatting and mounting the system partition

Do NOT restart the computer at the end of the installation. If you have already restarted it, boot the computer again from the USB drive. You do not need to reinstall the system.

Defining the boot order

You don't need a fancy boot manager. Now that you've installed Linux Mint, there are just two things you need to fix:

– By default, a Mac boots into macOS. However, we need to make sure that the Grub menu loads instead.

– Grub recognizes macOS but doesn't know how to boot it; so we need to fix that as well.

Open a terminal and enter the following command to install **efibootmgr** (this works whether you are connected to the Internet or not):

- `apt install efibootmgr`

Next, run the following command to view your EFI entries and boot order:

- `sudo efibootmgr`

For technical reasons (Linux Mint is based on Ubuntu), Linux Mint uses the name “ubuntu” as its EFI identifier. As a result, both of your operating systems will appear in the efibootmgr output as follows:

- Mint (“ubuntu”) is located at the address „**Boot0000**“
- Mac OS X is located at the address „**Boot0080**“

Now check the “BootOrder”. This determines what is booted and in what order. The value is set to “0080,” so that Mac OS X is booted first.

Let’s change this using the following command so that Mint (located at “Boot0000” or “0”) boots first, followed by Mac OS X (located at “Boot0080” or “80”):

- `sudo efibootmgr -o 0,80`

In the terminal now simply type

- `efibootmgr`

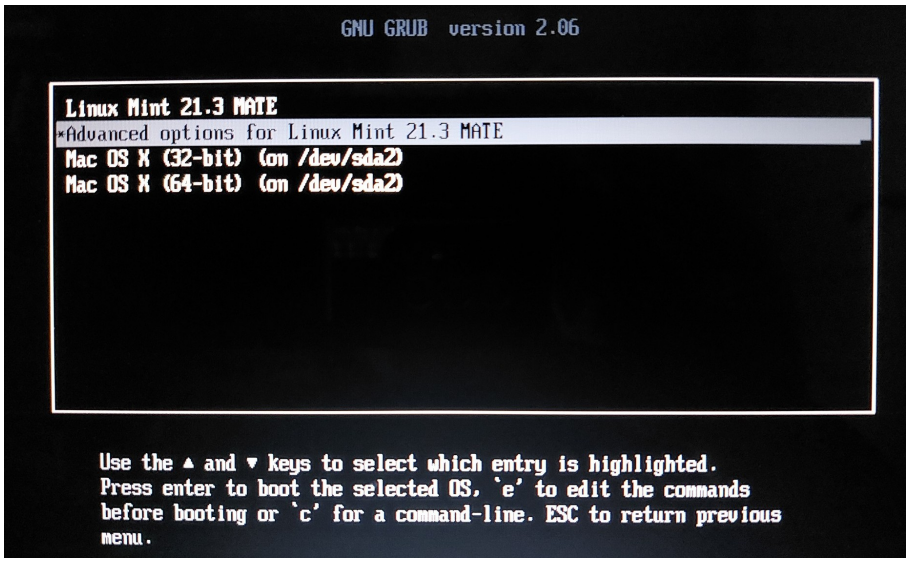
to check the settings. The output should look something like this:

The boot order should now show that Mint (Ubuntu) boots first (BootCurrent: 0000), and if that ever fails, Mac OS X would boot next (in theory ;-)). In other words: The Mac now boots into Grub. From there, you can select Mint or press the Escape key and type “exit” to start OS X.

So restart the computer now, and it should boot directly into the Mint GRUB menu.

Addendum:
There have been some changes to GRUB in the meantime. It is not certain that GRUB will recognize OS-X correctly and boot OS-X properly when you press “Escape” → “exit”.
When I start my Mac, it boots directly into the GRUB menu. Here, you can select options (e.g., a different kernel) or press the Enter key, and the computer will start Linux Mint immediately or automatically after 10 seconds.
But if I want to work with OS-X, I start up the Mac while holding down the ALT key. That way, I can select the system I want right away (including Linux).

The GRUB-Menu



If the GRUB menu does not appear when you start up your Mac, you will need to edit the following file: `/etc/default/grub`

The file should then look something like this:

grub ×

```
# 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_TIMEOUT_STYLE="menu"
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR="`( . /etc/os-release; echo ${NAME:-Ubuntu} ) 2>/dev/null || echo Ubuntu`"
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash acpi_backlight=native"
GRUB_CMDLINE_LINUX=""

# If your computer has multiple operating systems installed, then you
# probably want to run os-prober. However, if your computer is a host
# for guest OSes installed via LVM or raw disk devices, running
# os-prober can cause damage to those guest OSes as it mounts
# filesystems to look for things.
GRUB_DISABLE_OS_PROBER=false

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

# Uncomment to disable graphical terminal
GRUB_TERMINAL="console"

# The resolution used on graphical terminal
# note that you can use only modes which your graphic card supports via VBE
# you can see them in real GRUB with the command `vbeinfo'
GRUB_GFXMODE="640x480"

# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to Linux
#GRUB_DISABLE_LINUX_UUID="true"

# Uncomment to disable generation of recovery mode menu entries
#GRUB_DISABLE_RECOVERY="true"

# Uncomment to get a beep at grub start
#GRUB_INIT_TUNE="480 440 1"

```

On a Mac, you may need to change the line `#GRUB_TERMINAL=console` to `GRUB_TERMINAL=console`

After making changes, you must update the GRUB configuration file in the terminal using `sudo update-grub` and then restart the computer.

Driver

You may need to install proprietary drivers (Wi-Fi, graphics card) for your system. To do so, go to the menu → Control Center → Driver Management. If you are online, all the necessary drivers will be displayed here.
If you are offline, try using the USB drive you used to install the system. Launch the Driver Manager, connect the USB drive, and once it is recognized, click "OK" in the Driver Manager and follow the instructions.

What needs to be done next…

1) Update the system using the Update Manager

2) Switch to local (faster) mirror servers

3) If necessary, install any missing graphics drivers (Driver Manager)

4) Install full multimedia support. To do this, enter the following command in the Terminal:
   `sudo apt-get install mint-meta-codecs`

5) Install your programs

6) Back up the system using "Timeshift"

7) Deactivate unnecessary startup programs (Menu → Control Center → Startup Programs)

8) Enable the firewall (Menu → Control Center → Firewall Configuration)

9) Enable NumLock automatically at startup: To do this, you must first install "**numlockx**". In the terminal, enter: `sudo apt-get install numlockx`
   Now go to Menu → Control Center → Login Window. Under the "Settings" tab, enable the numeric keypad.
   Restart the computer.

Optimize SSD (optional)

1) noatime

By adding "noatime" to `/etc/fstab`, you disable the "access timestamp" operation, which the operating system sets every time it reads a file. "noatime" is better for an SSD. This has virtually no negative side effects (only in rare cases that most users will never encounter).

Open a terminal window. Type: `xed admin:///etc/fstab`

Note: The three consecutive slashes are not a typo—they are intentional! (For Ubuntu: use gedit instead of xed)

Press Enter

Now, in this text file, insert the word `noatime` into the line for your root partition and your other Linux partitions, immediately before `errors=remount-ro`. Use a comma as a separator, not a space.

Note: Do **not** add `noatime` to the line for the swap file or partition (if present), the line for the /boot/efi partition (if present), or the line for the /boot partition (if present)!

If you have other EXT4 partitions, you will likely see „`defaults`“ instead of „`errors=remount-ro`“ in their fstab entries. In that case, insert `noatime` immediately before `defaults`. Again, use a comma as a separator, not a space.

Here is an example of a customized line for a root partition, where you can see exactly where the `noatime` option needs to be set:

```
UUID=xxxxxxx / ext4 noatime,errors=remount-ro 0 1
```

Following is an example of a modified line for another (non-root) partition, which is mounted under /storage in the example:

```
UUID=xxxxxxx /storage ext4 noatime,defaults 0 1
```

Note: There **must be no space** after the comma following “`noatime`”! Otherwise, a boot error will likely occur. See the example lines above.

Save the modified text file and close it.

Now proceed to the next step.

2) Select the trim method

– Check TRIM compatibility:

```
sudo hdparm -I /dev/sda | grep TRIM
```

– Check the status of the TRIM timer/schedule:

```
systemctl status fstrim.timer
```

– Check TRIM status:

```
systemctl status fstrim
```

Using the TRIM function is recommended to ensure your SSD continues to perform well over the long term. Otherwise, it may start to slow down after a while. This applies even to new SSDs: Although new SSDs these days typically have some form of garbage collection built into their firmware, it’s always better to run TRIM regularly as well.

Note: Very old SSDs manufactured before 2010 generally do not support TRIM.

In Linux Mint and Ubuntu, automatic TRIM is enabled by default when you install them on an SSD. This is done via a weekly “job.” But in many (most?) cases, once a week isn't enough: waiting a whole week for recoverable storage space to become available again is simply too long.

Recommended method: daily job

Trimming the database daily is better than doing it weekly, since you don't have to wait as long for the recovered storage space to become fully available again. You can schedule it as follows:

a) Copy the following command and paste it into the Terminal to create a new folder:

```
sudo mkdir -v /etc/systemd/system/fstrim.timer.d
```

Press Enter. Enter your password when prompted. Press the Enter key again.

b) Copy the following command line and paste it into the Terminal to create a new file in this new folder:

```
sudo touch /etc/systemd/system/fstrim.timer.d/override.conf
```

Press Enter.

c) Copy the following command line and paste it into the Terminal to edit the new file:

```
xed admin:///etc/systemd/system/fstrim.timer.d/override.conf
```

Note: The three consecutive slashes are not a typo—they are intentional! (For Ubuntu: use gedit instead of xed)

Press Enter.

d) Now copy this blue text and paste it into the blank text document:

```
[Timer]
OnCalendar=
OnCalendar=daily
```

Note: The duplicate entry for OnCalendar is not an error; it is intentional!
Save the modified file and close it.

e) Restart your Computer.

f) Verify that you have successfully edited the fstrim configuration by running this terminal command:

```
systemctl cat fstrim.timer
```

The output should look something like this:

```
# /lib/systemd/system/fstrim.timer
[Unit]
Description=Discard unused filesystem blocks once a week
Documentation=man:fstrim
ConditionVirtualization=!container

[Timer]
OnCalendar=weekly
AccuracySec=1h
Persistent=true

[Install]
WantedBy=timers.target

# /etc/systemd/system/fstrim.timer.d/override.conf
[Timer]
OnCalendar=
OnCalendar=daily
```

Let's take a look at this output. The first part shows the default setting (weekly), and the second part shows the override setting you applied. This override setting consists of two parts: First, the existing OnCalendar setting (weekly) is deleted, and then a new OnCalendar setting (daily) is applied.

If you ever want to check whether fstrim was actually executed and when it happened, you can use this terminal command:

```
journalctl | grep fstrim
```

Undo change (return to weekly job)

Would you like to undo the change? It's very easy to change the daily job back to a weekly job. Here's how:

a) Copy the following command line and paste it into the Terminal to remove the modified settings:

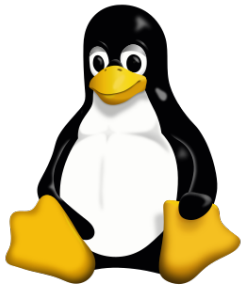
```
sudo rm -v /etc/systemd/system/fstrim.timer.d/override.conf
```

You are deleting the file `override.conf`.

Press Enter. Enter your password when prompted. Press Enter again.

b) Restart your Computer.

And now, enjoy Linux!



Greetings from Minetosh

https://www.minetoshsoft.com