

## Easy Linux tips project

Since 2009 this website is a complete quick guide for Linux, both for beginners and for advanced users!



Linux Mint and Ubuntu:  
Linux at its easiest.



### Donate and get free goodies!

If you wish to support this website, you can [make a donation and get free goodies](#). Thanks in advance!

### Most popular

10 things to do first in Linux Mint 18.3 Cinnamon  
10 things to do first in Ubuntu 16.04.x LTS  
Linux Mint: what it is and how to select the right flavour for you  
Linux Mint Xfce: a fine replacement for Windows 7, Vista and XP  
Wireless security: four popular myths and 12 tips  
Avoid 10 fatal mistakes in Linux Mint and Ubuntu  
Sitemap

This is the ground squirrel (xerus), after

## SSD: how to optimize your Solid State Drive for Linux Mint 18.x, Ubuntu 16.04 and Debian

[Back to the home page](#)



Below you'll find a complete how-to for optimizing your SSD for Ubuntu 16.04, Linux Mint 18.3 and Debian. So that you'll be able to enjoy your SSD for many years!

**Note:** this how-to is only for Ubuntu, Linux Mint and Debian; the how-to for openSUSE is [on another page](#).

### Contents

- 1 Avoid exaggerated measures
  - 1.1 Alignment is not necessary anymore
- 2 BIOS and UEFI: set it to AHCI
- 3 Check for updated firmware
- 4 Avoid quick wear: reduce write actions
- 5 Optional: reserve some space for overprovisioning
- 6 During installation: select EXT4
- 7 After the installation: noatime
- 8 Still relevant for Ubuntu 16.04 and Linux Mint 18.x: select your TRIM method
  - 8.1 Outdated method (no longer recommended): by rc.local
  - 8.2 Preferred method: daily by cron
    - 8.2.1 How to undo (back to a weekly cron job)
  - 8.3 Not advised: by discard
- 9 How to execute TRIM manually
- 10 Limit swap wear
- 11 Almost ready after reboot
- 12 Limit the write actions of Firefox
- 13 Limit the write actions of Chrome
- 14 Check whether the scheduler is set to deadline

**This website is being sponsored by Google Ads.**

**Are you using an ad blocker? Then you're also blocking my earnings from advertisements....**

**If you wish to support my website, you can configure your ad blocker [to make an exception for this website](#). Or you can [make a donation \(and get free goodies\)](#).**

**Thanks in advance....**

### Not advised: by discard

8.3. Unfortunately, another widely used method for automatic TRIM is to add the option [discard](#) to /etc/fstab, for the line for your root partition and for potential other Linux partitions in /etc/fstab. This isn't a good method, which I will explain below.

**Note:** don't ever add it to the line for the swap partition, as that's already being trimmed automatically by the system by default, during the boot process!

The disadvantage of the discard method is, that it may cause the system to slow down a lot. Because it forces the system to apply TRIM instantly on every file deletion. That's why I advise against this method.

**Note:** as already mentioned, old SSD's made before 2010 usually don't support TRIM. In that case it's of course not useful to apply the option [discard](#) in fstab.

which Ubuntu 16.04 Xenial Xerus has been named....



For your own free copy, [click here!](#)

A star of the Ubuntu family: Xubuntu!



In for a pleasant surprise? Experience the elegant simplicity of [Xubuntu!](#)

You can get your own free copy [here](#).

# 141

days until  
Ubuntu 18.04 LTS  
Bionic Beaver

Release schedule of  
Ubuntu 18.04 LTS  
Bionic Beaver

<https://wiki.ubuntu.com/Bio>

A fine replacement for  
Windows XP and  
Vista



Linux Mint Xfce is a fine replacement for

15 Do NOT enable hibernation

16 Also for an SSD: prevent fragmentation, and DO NOT defrag

17 Dual boot? Don't let Windows kill your SSD

18 Enjoy your SSD carefree for years and years

19 Want more tips?

## Avoid exaggerated measures

1. First a word of caution: **don't exaggerate!** There's a lot of exaggeration to be found on the web on this issue.

On the one hand you have people who don't take any special measures for SSD's at all, and on the other hand you see people who take all kinds of extreme and complicated precautions. Neither side is behaving sensibly.

This how-to is meant to be a "best practice". It'll show you how to achieve a very good result by applying just a few rather simple measures. With those, you'll be able to enjoy your SSD carefree, for years and years to come!

Taking into account the long warranty periods that the manufacturers are giving, probably for more than five years (10 years should be a reasonable expectation). Considerably longer than an ordinary platter hard disk, anyway...

## Alignment is not necessary anymore

1.1. A common misunderstanding is, that the partitions on the SSD should be properly aligned first. That used to be true years ago, but no longer.

In the current versions of Ubuntu and Linux Mint, and in Debian since Wheezy, all tools automatically align filesystems and partitions to the

If you want to do it anyway, then this is how:

a. First make sure that you have installed the applications [gksu](#) and [leafpad](#).

b. Then type in the terminal ([use copy/paste](#)):

```
gksudo leafpad /etc/fstab
```

Press Enter.

c. Now add the word **discard** to the line for your root partition and your other Linux partitions, just before [noatime](#) (assuming that you've already added the [noatime](#) switch). **Note:** don't add it to the line for the swap partition!

Here's an example of an adapted line, in which you can see the exact spot where the **discard** switch has to be put:

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

**Note:** there should be no space after the comma after [discard](#)! Otherwise boot failure will probably occur. See the example line above.

**Note:** if you have a separate partition for [/boot/efi](#), don't add a trim command for that partition in [fstab](#)! It's useless, as that partition is seldom being written to. And it may even cause serious malfunctions.

d. Save the modified text file and close it.

e. Reboot your computer.

f. Now proceed to item 9.

## How to execute TRIM manually

9. You can execute TRIM manually as well, namely as follows:

Windows XP and Vista!  
[Read here why.](#)

optimal 4096 byte page size. So there's no need to do anything for alignment on an SSD.

## BIOS and UEFI: set it to AHCI

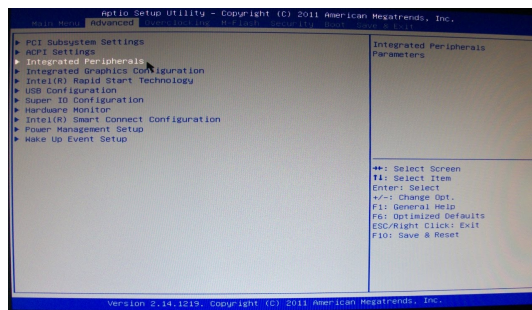
2. A Solid State Disk, or rather Drive (SSD), is sometimes only recognized properly by the BIOS or UEFI, when in the BIOS/UEFI the feature AHCI has been activated for SATA (instead of IDE).

This feature may be hard to find in the BIOS/UEFI, because there's absolutely no standardization at all in menu structures for BIOS and UEFI (sigh....).

That's why I've made two screen shots of the BIOS of my computer, in which you can see this particular feature. Hopefully it'll help you to find it in your own BIOS/UEFI.... The motherboard of my computer is, by the way, an MSI H61MA-E35 (B3).

**Note:** doesn't the BIOS or UEFI of your computer offer the option to switch to AHCI? Then the BIOS/UEFI might possibly detect the SSD automatically and automatically choose the right BIOS settings for it.

First of all, in the BIOS of my computer I go to the tab **Advanced**, and there I expand the section **Integrated Peripherals** (click on the image below to enlarge it):



Then, under the header "SATA Configuration", I change IDE into AHCI Mode. Namely for SATA Mode.

Launch a terminal window.  
*(You can launch a terminal window like this: [\\*Click\\*](#))*

Type ([use copy/paste](#)):

`sudo fstrim -v /`

Press Enter. Type your password when prompted; this remains entirely invisible, not even dots will show, this is normal. Press Enter again.

This operation may last for minutes; it then looks as if the terminal has frozen. That's not true, however; simply wait patiently....

**Note: on a few SSD models (specifically two models from Crucial), executing this command when there's high disk activity (I/O activity), might cause problems. So only apply it when there's not much activity going on. Preferably with all other applications closed.**

The above terminal command is enough when you have only one Ubuntu or Mint partition and a swap partition (because for the swap it's unnecessary; the system takes care of that automatically).

When you have more mounted EXT4 partitions, you'll have to adapt the command line accordingly. For example, if you have a separate home partition (although that's [not very useful](#)), then the command is: `fstrim -v /home`

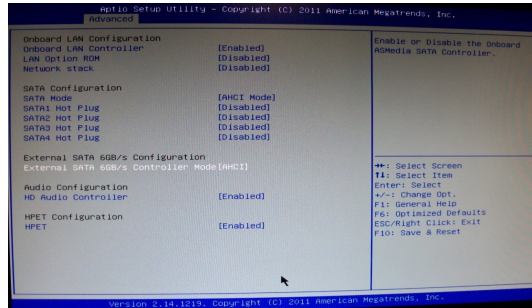
Now proceed to the next item.

## Limit swap wear

10. With the action described below, you limit the use of the swap partition (the virtual memory on the SSD). Without disabling it entirely, because that would go too far: in case of *extreme* RAM load, your Linux has to

Then, under the header "External SATA 6GB/s Configuration", I also change IDE into AHCI. Namely for External SATA 6gb/s Controller Mode.

See the image below (click on it to enlarge it):



### Check for updated firmware

3. Check the website of the SSD manufacturer, whether there's an updated version available of the firmware for your SSD. If so, apply it.

For most SSD's it doesn't matter that you're running Linux, because many manufacturers offer such firmware updates by means of an iso file with which you can [create a bootable CD, much like Ubuntu and Linux Mint do](#).

In the first year after purchase, repeat this check every month or so.

### Avoid quick wear: reduce write actions

4. A Solid State Drive is worn down relatively quickly by write actions. Especially the oldest generations of SSD's were vulnerable in that aspect, but to a lesser degree that's still the case for the newer generations.

Below you'll find some tips on keeping wear down to a reasonable minimum, by limiting the write actions. Within reason, because a modern SSD is already pretty wear-

be able to "swap".

Linux's inclination to use the swap, is determined by a setting. The lower the setting number, the more system load is required before your Linux starts using the swap.

On a scale of 0-100, the default setting is 60. Which is much too high for normal desktop use, and only fit for servers. For SSD's, it's just crazy.

A detailed explanation can be found [here](#) (link dead? Then download [this pdf file](#) with the same content).

Now the how-to:

a. First make sure that you have installed the applications [gksu](#) and [leafpad](#).

b. Now check your current swappiness setting. Type in the terminal ([use copy/paste](#)):

```
cat /proc/sys/vm/swappiness
```

Press Enter.

The result will probably be 60.

c. Now type in the terminal ([use copy/paste](#)):

```
gksudo leafpad /etc/sysctl.conf
```

Press Enter.

d. Add the following blue lines, at the very end of the existing text in that file ([use copy/paste](#) to avoid errors):

```
# Sharply reduce the inclination to swap
vm.swappiness=1
```

e. Save the file and close it.

f. Now proceed to the next item.

### Almost ready after reboot



resistant by default.

When you apply my tips, a lifespan of more than 10 years should be a normal expectation for your SSD. Which is considerably longer than the average lifespan of an old-fashioned platter disk.

**Note: these directions are only meant for a Solid State Drive (SSD), not for an ordinary conventional spinning hard disk!**

**Optional: reserve some space for overprovisioning**

5. Over-provisioning (with or without a dash) is a technique that's being used to improve the performance and life span of an SSD. I won't explain it here, but it boils down to **not** formatting a part of the SSD, which then remains unallocated space (unused capacity).

General opinion used to be, that it's wise to reserve as much as 28 percent of the storage capacity of an SSD for such unallocated space. That's of course a huge slice out of the total storage capacity, which makes nobody happy....

However, in the newer generations of SSD's the technology and firmware have improved so much, that such huge losses of storage capacity aren't needed anymore. Modern SSD's usually have some built-in unallocated space for overprovisioning anyway, which isn't accessible to the user.

But to be on the safe side, I still advise to reserve about seven percent for unallocated space. With a maximum of 10 GB, because more would be unnecessary in almost any case. Reason: commercial pressure might induce SSD manufacturers to be too stingy with built-in "useless" overprovisioning space....

11. Reboot your computer.

Your SSD is almost ready now, but not quite. So after the reboot, proceed to the next item.

**Limit the write actions of Firefox**

12. You can limit the write actions of Firefox as follows.

a. Set the cache to 0:  
Firefox menu button (with the three dashes on it) - Preferences - Advanced  
Tab [Network](#)  
section "Cached Web Content": tick [Override automatic cache management](#) and set the cache to 0 MB.

b. Type [about:config](#) in the url bar and press Enter. Click the button to accept the risk.

In the search bar, type: [sessionstore](#)

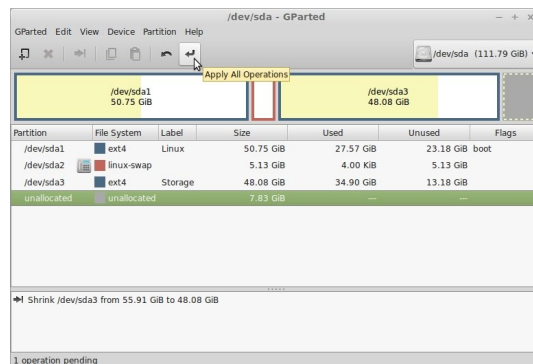
Double-click on the item called [browser.sessionstore.interval](#). The default interval is [15000](#), which means 15 seconds. Add three zeros to the existing value, so that it becomes: [15000000](#) and click the OK button.

Close Firefox and launch it again. Now you've practically disabled the session restore feature, which remembers what pages were opened if Firefox experiences an unexpected shutdown (read: crashes). This feature is neat, but causes many disk writes. Too many for an SSD.

c. If you have [installed Oracle Java](#), limit the write actions of the Java plugin:  
launch the Java Control Panel - Tab General:  
Temporary Internet Files - Settings...  
Remove the tick for: [Keep temporary files on my computer](#).

Seven percent with a maximum of 10 GB, is not a big loss of storage capacity. And it's beneficial for wear reduction on the very long term. What's more, on the short term it can be useful when your partitions are very full, because then your SSD won't slow down so much.

See the screenshot below of my own SSD, in which the seven percent unallocated space has been highlighted (click on the image to enlarge it):



I classify overprovisioning as optional, because it's not as important as the other measures that I advise and because it costs you storage space. Yet on my own SSD's I always apply this.

### During installation: select EXT4

6. The best file system (formatting) for an SSD, is the usual default EXT4. So you don't have to choose anything different in this respect.

The journaling that comes with EXT4 causes some write activity, but not very much. Plus journaling is a very important feature for system crash recovery, so you don't want to disable it.

**Note:** don't select the BTRFS file system! Because under certain circumstances, BTRFS causes a huge amount of write actions.

### Limit the write actions of Chrome

13. The write actions of Google Chrome can be limited as follows.

a. Click on the three dots (vertical line) in the upper right of the Chrome window - Settings. Down below, click on [Show advanced settings...](#)

Section Privacy: remove the tick for: [Use a prediction service to load pages more quickly](#)

Close Chrome and launch it again.

This way, you've disabled the "preload" feature, which can cause a lot of extra web traffic and also a lot of writes.

b. Currently, there's unfortunately no "neat" way of disabling the session restore feature in Chrome. This feature ensures that your opened web pages and tabs will be reopened automatically when launching Chrome after a crash.

Which is nifty, but it causes a lot of web traffic and a lot of disk writes. Too much so for an SSD.

A workaround is to launch Chrome in the "incognito" mode, with this terminal command:

[google-chrome-stable --incognito](#)

Some reports claim that this has the same effect, while keeping history and stored passwords available:

[google-chrome-stable --disable-session-crashed-bubble --disable-infobars](#)

Hopefully, this setting will become easier accessible in future versions of Chrome.

Now proceed to the next item.

**After the installation: noatime**

7. With "noatime" in /etc/fstab, you disable the write action "access time stamp", that the operating system puts on a file whenever it's being read by the operating system. For an SSD "noatime" is much better.

You can do that as follows:

a. First make sure that you have installed the applications [gksu](#) and [leafpad](#):

Launch a terminal window.  
(You can launch a terminal window like this: [\\*Click\\*](#))

Type (or [copy/paste](#)):  
`sudo apt-get install gksu leafpad`

Press Enter and submit your password. Please note that the password will remain invisible, not even asterisks will show, which is normal.

b. Then type in the terminal ([use copy/paste](#)):  
`gksudo leafpad /etc/fstab`

Press Enter.

c. Now add the word **noatime** to the line for your root partition and your other Linux partitions, just before `errors=remount-ro`. **Note:** don't add it to the line for the swap partition!

Here's an example of an adapted line, in which you can see the exact spot where the **noatime** switch has to be put:

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

**Note:** there should be no space after the comma after **noatime**! Otherwise boot failure will probably occur. See the example line above.

**Check whether the scheduler is set to deadline**

14. By default, Ubuntu and Linux Mint use the I/O scheduler Deadline, which is good for both SSD's and conventional platter hard disks. So for the scheduler, you don't have to change anything.

But because it's so important for the performance of your SSD, it's wise to check whether this setting is indeed as it should be.

You can check your current scheduler as follows: [use copy/paste](#) to transfer the following blue command line to the terminal:

```
cat /sys/block/sda/queue/scheduler
```

(if your drive isn't [sda](#), change the line accordingly)

Press Enter.

The output should be:  
`noop [deadline] cfq`

Which means: deadline is active, but noop and cfq are also supported. Is the outcome different? Then change it as follows:

a. First make sure that you have installed the applications [gksu](#) and [leafpad](#).

b. Then type (use copy/paste):

```
gksudo leafpad /etc/default/grub
```

Press Enter.

Find this line:

```
GRUB_CMDLINE_LINUX_DEFAULT="q
splash"
```

And replace it by this line:

```
GRUB_CMDLINE_LINUX_DEFAULT="e
quiet splash"
```

d. Save the modified text file and close it.

e. Now proceed to the next item.

### Still relevant for Ubuntu 16.04 and Linux Mint 18.x: select your TRIM method

8. The cleaning action TRIM is necessary for the good performance of your SSD in the long run. Otherwise it'll become very slow after some time.

All modern SSD's should support TRIM, but older SSD's from before 2010 usually don't. When in doubt, check it like this (example where your SSD is called **sda** by Linux):

Launch a terminal window.  
(You can launch a terminal window like this: [\\*Click\\*](#))

Type ([use copy/paste](#)):

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

Press Enter. Type your password when prompted; this remains entirely invisible, not even dots will show, this is normal. Press Enter again.

Your SSD supports TRIM when one of the messages you get, looks like this:

### Data Set Management **TRIM** supported

It's easiest to let the system perform an automatic TRIM. That can be done in several ways.

In Ubuntu 16.04 and in Linux Mint 18.x automatic TRIM is enabled by default, when you install Ubuntu 16.04 or Linux Mint 18.x on an SSD. Namely by a weekly "cron job".

But in many (most?) cases, once a week isn't often enough, so I advise

c. Save the modified file and close it.

d. Now update Grub for this change. In the terminal (use copy/paste):

```
sudo update-grub
```

Press Enter. Type your password when prompted; this remains entirely invisible, not even dots will show, this is normal. Press Enter again.

e. Reboot your computer.

f. Proceed to the next item.

**Note:** when you have only an SSD in your computer and no platter disk as well, some how-to's suggest that you may experience a slight performance benefit by choosing **noop** for scheduler, instead of **deadline**. This is not true, however. Deadline is always the best choice.

### Do NOT enable hibernation

15. Hibernation causes a huge amount of write actions, which is very bad for an SSD. [Make sure it's disabled!](#) (item 10, left column)

This is especially relevant for Linux Mint, because in Linux Mint hibernation is unfortunately enabled by default...

### Also for an SSD: prevent fragmentation, and DO NOT defrag

16. For an SSD, fragmentation of the file system is a smaller problem than for ordinary rotating hard disks. But it's nevertheless still a problem, so it's useful to prevent fragmentation as much as possible on an SSD, too. You can achieve that by preserving [a minimum of 20 % free space on each partition](#) (item 4, left column).

Prevention is all the more important,



you to read on.

### **Outdated method (no longer recommended): by rc.local**

8.1. You can add the TRIM command to `/etc/rc.local`. Then this command will be executed automatically on system boot. Before Ubuntu 16.04 and Linux Mint 18.x, this was a good solution that hardly slowed the boot process down.

However, in Mint 18.x and Ubuntu 16.04 there's a new boot process called `systemd`. For `systemd` it may cause a huge boot delay (minutes!) when you put the `fstrim` command in `/etc/rc.local`. So that's why I don't recommend this method anymore.

### **Preferred method: daily by cron**

8.2. Trimming daily is better than weekly, because then you don't have to wait so long before recoverable disk space becomes fully usable again.

You can schedule a daily trimming as follows.

a. [Copy/paste](#) the following command line into the terminal:

```
sudo mv -v /etc/cron.weekly/fstrim /etc/cron.daily
```

Press Enter. Type your password when prompted; this remains entirely invisible, not even dots will show, this is normal. Press Enter again.

With that command you've transferred the script file `fstrim` from the "weekly" directory of cron to the "daily" directory. Ubuntu or Linux Mint will now perform the daily cron job for `fstrim` automatically, at 06:25, or (when the computer isn't on at that time), automatically at a later time on the same day.

as defragmenting is out of the question anyway: your SSD would wear rapidly, because of the many write actions it causes.

The reason that fragmentation is also a problem for an SSD, is this:

The mechanical seek time of an SSD is always 0, regardless of the fragmentation. That's a big difference with rotating hard disks, on which the seek time increases as fragmentation grows. This is a significant improvement.

But mechanical seek time only makes up a part of total access time, or I/O time, of any single input/output request made to the disk.

I/O time is the time a computer system takes to complete a request cycle. All the way from application, operating system and driver down to disk hardware, memory cells, and then back again.

Zero mechanical seek time certainly does not mean zero I/O time. No matter how fast an SSD is, its I/O time can never be zero. File system fragmentation increases I/O time in an SSD, even when the mechanical seek time is zero.

To put it another way: the performance degradation as a result of fragmentation is not caused by the storage device alone (whether there's a mechanical moving part or not), but it's also a problem concerning the system as a whole.

The task for the system becomes heavier, when there are more files to be chopped up and more pieces of files to be glued together. The heavier the task, the longer the processing time.

A good and elaborate explanation of this fact [can be found here](#). Website

b. Reboot your computer.

c. Now proceed to item 9 in the column on the right.

### How to undo (back to a weekly cron job)

8.2.1. Want to undo? It's easy to change the daily cron job back to a weekly job, namely as follows.

a. [Copy/paste](#) the following command line into the terminal:

```
sudo mv -v /etc/cron.daily/fstrim
/etc/cron.weekly/fstrim
```

Press Enter. Type your password when prompted; this remains entirely invisible, not even dots will show, this is normal. Press Enter again.

With that command you've transferred the script file fstrim from the "daily" directory of cron to the "weekly" directory. Ubuntu or Linux Mint will now perform the cron job for fstrim once a week, as is the default setting.

b. Reboot your computer.

*(continued in the column on the right)*

down? Here's [a pdf file](#).

### Dual boot? Don't let Windows kill your SSD

17. Do you have a dual boot with Windows? Then don't let Windows kill your SSD by defragmenting it.

Defragmentation will kill your SSD in a very short time, because of the multitude of write actions that it causes.

Within Linux you don't have this problem, because Linux filesystems [hardly fragment in the first place](#) and so don't need to be defragmented anyway.

### Enjoy your SSD carefree for years and years

18. Now you'll be able to enjoy your SSD carefree!

As already said in the start of this how-to: taking into account the long warranty periods that the manufacturers are giving, your SSD will probably last for more than five years (10 years should be a reasonable expectation).

Considerably longer than an ordinary platter hard disk generally lasts, anyway...

### Want more tips?

19. Do you want more tips and tweaks for Ubuntu or Linux Mint? There's [a lot more of them](#) on this website! Like these ones:

[Security in Ubuntu and Linux Mint](#)

[Four popular myths and 11 tips about wireless security \(for wifi\)](#)

[Firefox: tweak it right](#)

[Chrome and Chromium: make them even better](#)

[Libre Office: configure it well](#)

[Make Linux Mint run faster](#)

[Clean your Linux Mint](#)

*To the content of this website applies a [Creative Commons license](#).*

[Back to the home page](#)

[Disclaimer](#)

## Comments

You do not have permission to add comments.

[Sign in](#) | [Report Abuse](#) | [Print Page](#) | Powered By [Google Sites](#)