

---

# User Guide

**Linux Mint**

**Jul 20, 2020**



---

## Software

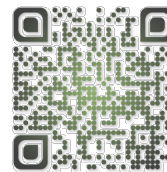
---

<b>1</b>	<b>Snap Store</b>	<b>3</b>
<b>2</b>	<b>Chromium</b>	<b>5</b>
<b>3</b>	<b>Grub Boot Menu</b>	<b>9</b>
<b>4</b>	<b>How to upgrade to Linux Mint 20</b>	<b>13</b>



This is the Linux Mint User Guide.

This Guide is not final. Content is being added slowly but surely :)





The **Snap Store**, also known as the *Ubuntu Store*, is a commercial centralized software store operated by **Canonical**.

Similar to AppImage or Flatpak the Snap Store is able to provide up to date software no matter what version of Linux you are running and how old your libraries are.

## 1.1 Criticism

### 1.1.1 Centralized control

Anyone can create APT repositories and distribute software freely. Users can point to multiple repositories and define priorities. Thanks to the way APT works, if a bug isn't fixed upstream, Debian can fix it with a patch. If Debian doesn't, Ubuntu can. If Ubuntu doesn't Linux Mint can. If Linux Mint doesn't, anyone can, and not only can they fix it, they can distribute it with a PPA.

Flatpak isn't as flexible. Still, anyone can distribute their own Flatpaks. If Flathub decides they don't want to do this or that, anyone else can create another Flatpak repository. Flatpak itself can point to multiple sources and doesn't depend on Flathub.

Although it is open-source, Snap on the other hand, only works with the Ubuntu Store. Nobody knows how to make a Snap Store and nobody can. The Snap client is designed to work with only one source, following a protocol which isn't open, and using only one authentication system. Snapp is nothing on its own, it can only work with the Ubuntu Store.

This is a store we can't audit, which contains software nobody can patch. If we can't fix or modify software, open-source or not, it provides the same limitations as proprietary software.

### 1.1.2 Backdoor via APT

When Snap was introduced Canonical promised it would never replace APT. This promise was broken. Some APT packages in the Ubuntu repositories not only install snap as a dependency but also run snap commands as root without your knowledge or consent and connect your computer to the remote proprietary store operated by Canonical.

## 1.2 Disabled Snap Store in Linux Mint 20

Following the decision made by Canonical to replace parts of APT with Snap and have the Ubuntu Store install itself without users knowledge or consent, the Snap Store is forbidden to be installed by APT in Linux Mint 20.

---

**Note:** For more information read the announcements made in [May 2020](#) and [June 2019](#).

---

## 1.3 How to install the Snap Store in Linux Mint 20

Recommended or not, if you want to use the Snap Store, re-enabling and installing it is very easy.

```
sudo rm /etc/apt/preferences.d/nosnap.pref
apt update
apt install snapd
```



Chromium is an open-source web browser developed by Google.

Google also develops a proprietary browser based on Chromium called Chrome. This is the browser Google promotes and provides repositories for. There are no Google repositories for Chromium.

This page lists the different ways to install Chromium in Linux Mint.

### 2.1 Default Linux Mint repositories

In LMDE, Linux Mint 18.x and Linux Mint 19.x you can install Chromium from the repositories:

```
apt install chromium-browser
```

However, in Linux Mint 20.x, as announced in the [May 2020 Monthly News](#), *chromium-browser* is a dummy package and Chromium is no longer available in the repositories.

### 2.2 Unofficial repositories and PPAs

Unofficial repositories and PPAs provide easy access to packages and updates. As always, with unofficial sources, you need to trust the maintainer when it comes to software stability, security and frequency of updates.

---

**Hint:** Create a Timeshift system snapshot before adding any unofficial software sources or installing software from third parties. This will allow you to revert any potential mistakes or undesired consequences.

---

**Warning:** Do not use PPAs or third party sources unless you trust its maintainer.

### 2.2.1 Saikrishna Arcot's Chromium BETA PPA

This **BETA PPA** contains the latest Chromium Beta builds, with hardware video decoding enabled (hidden behind a flag), and support for Widevine (needed for viewing many DRM-protected videos) enabled.

As root, add the following to `/etc/apt/preferences.d/saiarcot895-chromium-beta.pref`

```
# Ensure packages from saiarcot895-chromium-beta PPA have priority
Package: *
Pin: release o=LP-PPA-saiarcot895-chromium-beta
Pin-Priority: 800
```

Then in a terminal window run

```
apt remove --purge chromium-browser
sudo add-apt-repository ppa:saiarcot895/chromium-beta
apt update
apt install chromium-browser
```

### 2.2.2 Saikrishna Arcot's Chromium DEV PPA

This **Dev PPA** contains the latest Chromium Dev builds, with hardware video decoding enabled (hidden behind a flag) and support for Widevine (needed for viewing paid videos on Netflix and Youtube) enabled.

As root, add the following to `/etc/apt/preferences.d/saiarcot895-chromium-dev.pref`

```
# Ensure packages from saiarcot895-chromium-dev PPA have priority
Package: *
Pin: release o=LP-PPA-saiarcot895-chromium-dev
Pin-Priority: 800
```

Then in a terminal window run

```
apt remove --purge chromium-browser
sudo add-apt-repository ppa:saiarcot895/chromium-dev
apt update
apt install chromium-browser
```

### 2.2.3 Ungoogled Chromium

**Ungoogled Chromium** provides a version of Chromium which doesn't use Google Web services.

```
echo 'deb http://download.opensuse.org/repositories/home:/ungoogled_chromium/Ubuntu_
↳Focal/ /' | sudo tee /etc/apt/sources.list.d/home:ungoogled_chromium.list
wget -nv https://download.opensuse.org/repositories/home:ungoogled_chromium/Ubuntu_
↳Focal/Release.key -O - | sudo apt-key add -
apt update
apt remove --purge chromium-browser
apt install ungoogled-chromium
```

## 2.3 APT Pinning

APT pinning consists in pointing to repositories which are not designed for Linux Mint 20 but with APT preferences which restrict the usage of these repositories to Chromium only and nothing else.

### 2.3.1 Debian Buster

As root, add the following to */etc/apt/sources.list.d/pinned-debian-chromium.list*

```
deb https://deb.debian.org/debian buster main
deb https://deb.debian.org/debian buster-updates main
deb http://security.debian.org/ buster/updates main
```

And the following to */etc/apt/preferences.d/pinned-debian-chromium.pref*

```
# Don't install anything other than chromium from the Debian repos
Package: *
Pin: origin "deb.debian.org"
Pin-Priority: -10

# Don't install anything other than chromium from the Debian repos
Package: *
Pin: origin "security.debian.org"
Pin-Priority: -10

# Exclude the game chromium-bsu
Package: chromium-bsu*
Pin: origin "deb.debian.org"
Pin-Priority: -10

# Exclude the game chromium-bsu
Package: chromium-bsu*
Pin: origin "security.debian.org"
Pin-Priority: -10

# Pattern includes 'chromium'
Package: chromium*
Pin: origin "deb.debian.org"
Pin-Priority: 700

# Pattern includes 'chromium'
Package: chromium*
Pin: origin "security.debian.org"
Pin-Priority: 700

# Chromium dependencies only in buster
Package: /libevent-2.1-6/ /libc6/ /libicu63/ /libjpeg62-turbo/ /libvpx5/
Pin: origin "deb.debian.org"
Pin-Priority: 1

# Chromium dependencies only in buster
Package: /libevent-2.1-6/ /libc6/ /libicu63/ /libjpeg62-turbo/ /libvpx5/
Pin: origin "security.debian.org"
Pin-Priority: 1
```

Then run the following commands:

```
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com:443 --recv-keys_  
↳DCC9EFBF77E11517  
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com:443 --recv-keys_  
↳648ACFD622F3D138  
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com:443 --recv-keys_  
↳112695A0E562B32A  
apt update  
apt remove --purge chromium-browser  
apt install chromium
```

## 2.4 Alternatives to Chromium

### 2.4.1 Google Chrome

Google makes it very easy to install Chrome. Visit the [Google Chrome Website](#), download and install the provided package and it will automatically add the Google repositories to your operating system.

**Warning:** Only use this browser if you trust [Google](#) with your privacy and data. Chrome is proprietary software. It cannot be audited or modified.

### 2.4.2 Firefox

Firefox is the most popular open-source Web browser and also the default browser in Linux Mint. It's available in the repositories.

```
apt install firefox
```

## 2.5 Alternatives to APT

### 2.5.1 The Snap Store

Chromium is available from the Snap Store.

```
apt install snapd  
sudo snap install chromium
```

**Warning:** Only use this store if you trust both [Canonical](#) and the maintainers of the snaps you install. Similar to proprietary software, software delivered by and code run by the Snap Store cannot be audited or modified. This store is disabled in Linux Mint 20.x. For more information read [Snap Store](#).

---

## Grub Boot Menu

---

Grub is the boot menu.

If you have more than one operating system installed, it allows you to select which one to boot.

Grub is also useful for troubleshooting. You can use it to modify the boot arguments or to boot from an older kernel.

### 3.1 How to make the Grub menu always visible

If you only run Linux Mint and there are no other operating systems on the computer, the menu is hidden by default.

To make it visible, as root, add these lines to */etc/default/grub.d/90\_custom.cfg*:

```
GRUB_TIMEOUT="5"  
GRUB_TIMEOUT_STYLE="menu"
```

Then type the following commands in a terminal:

```
sudo update-grub
```

### 3.2 How to theme the Grub menu

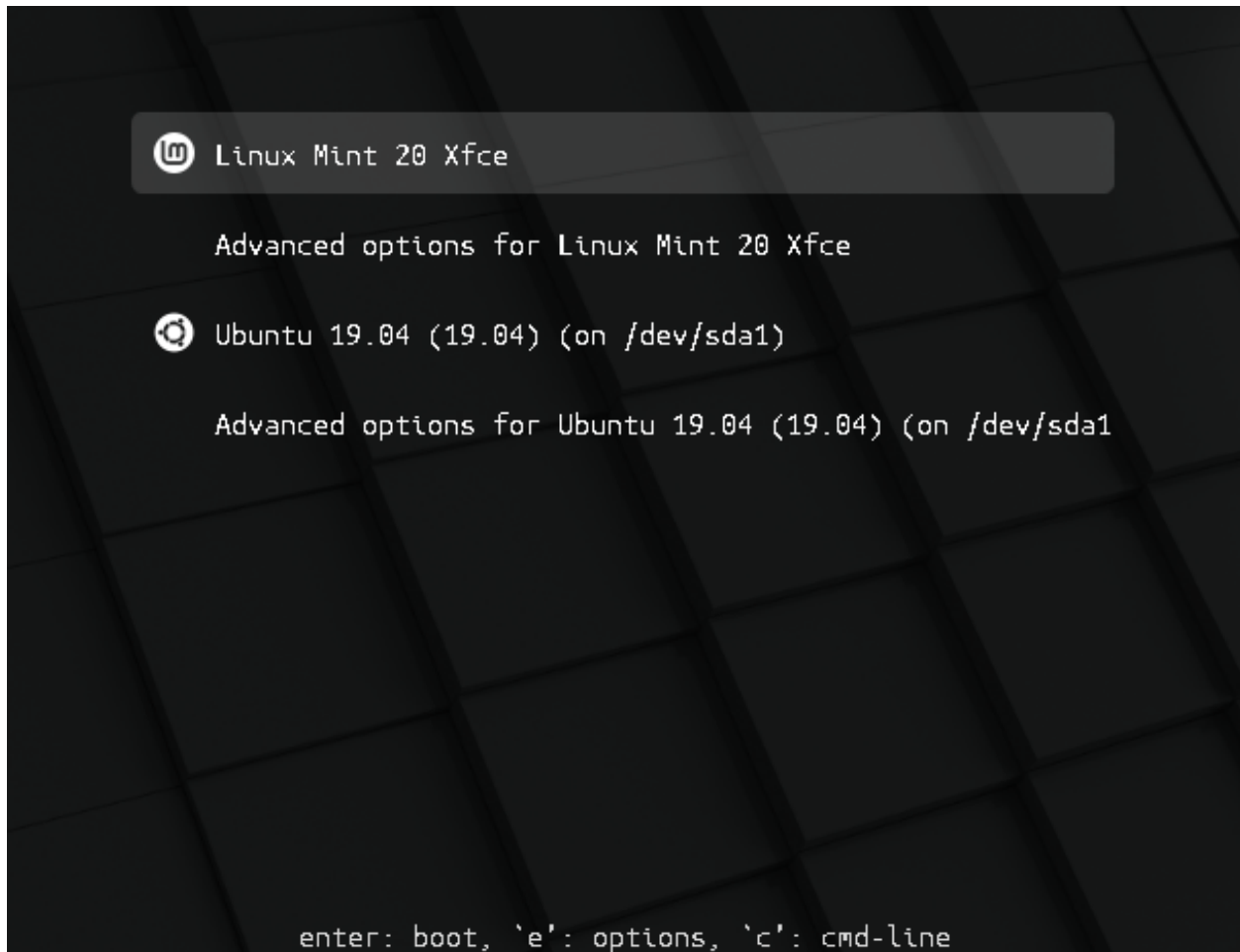
For compatibility reasons, some releases sometimes ship without a Grub theme:

```
GNU GRUB version 2.04

*Linux Mint 20 Xfce
Advanced options for Linux Mint 20 Xfce
Ubuntu 19.04 (19.04) (on /dev/sda1)
Advanced options for Ubuntu 19.04 (19.04) (on /dev/sda1)

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.
```

You can make it look like this:



To do so, open a terminal and type:

```
apt install --reinstall -o Dpkg::Options::="--force-confmiss" grub2-theme-mint
```

Or if you have a HiDPI screen, type this instead:

```
apt install --reinstall -o Dpkg::Options::="--force-confmiss" grub2-theme-mint-2k
```





---

## How to upgrade to Linux Mint 20

---

This page explains how to upgrade from Linux Mint 19.3 to Linux Mint 20.

### 4.1 Requirements

#### 4.1.1 64-bit architecture

Although both 32-bit and 64-bit versions of Linux Mint 19.3 are supported until April 2023, new releases of Linux Mint, including 20, are only available in 64-bit.

To upgrade to Linux Mint 20 you need to be running the 64-bit version of Linux Mint 19.3.

To check which version you're running type:

```
dpkg --print-architecture
```

If it says **amd64** you can upgrade to Linux Mint 20.

If it says **i386**, it means you're using the 32-bit version. In this case you cannot upgrade and you need to stick with Linux Mint 19.3.

#### 4.1.2 Experience with APT

To upgrade to Linux Mint 20 you need experience with APT and the command line.

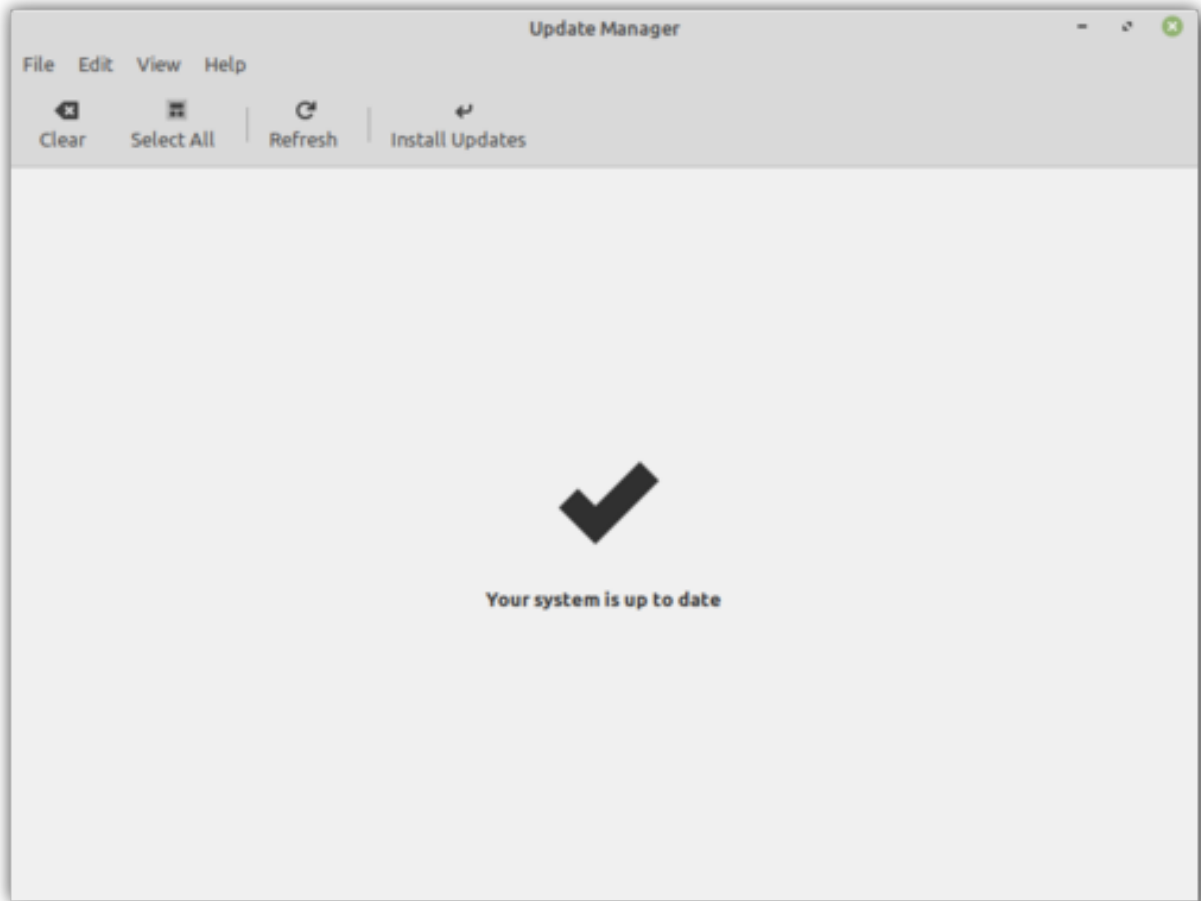
Upgrading to a newer package base is not trivial and it should not be performed by novice users.

You need to know how to type commands and read their output.

You also need to be experienced with APT. During the upgrade you'll need to understand the output of APT commands. You'll need to understand if a package needs to be removed, if it blocks the upgrade or if it conflicts with another package.

## 4.2 Preparation

### 4.2.1 Apply all package updates



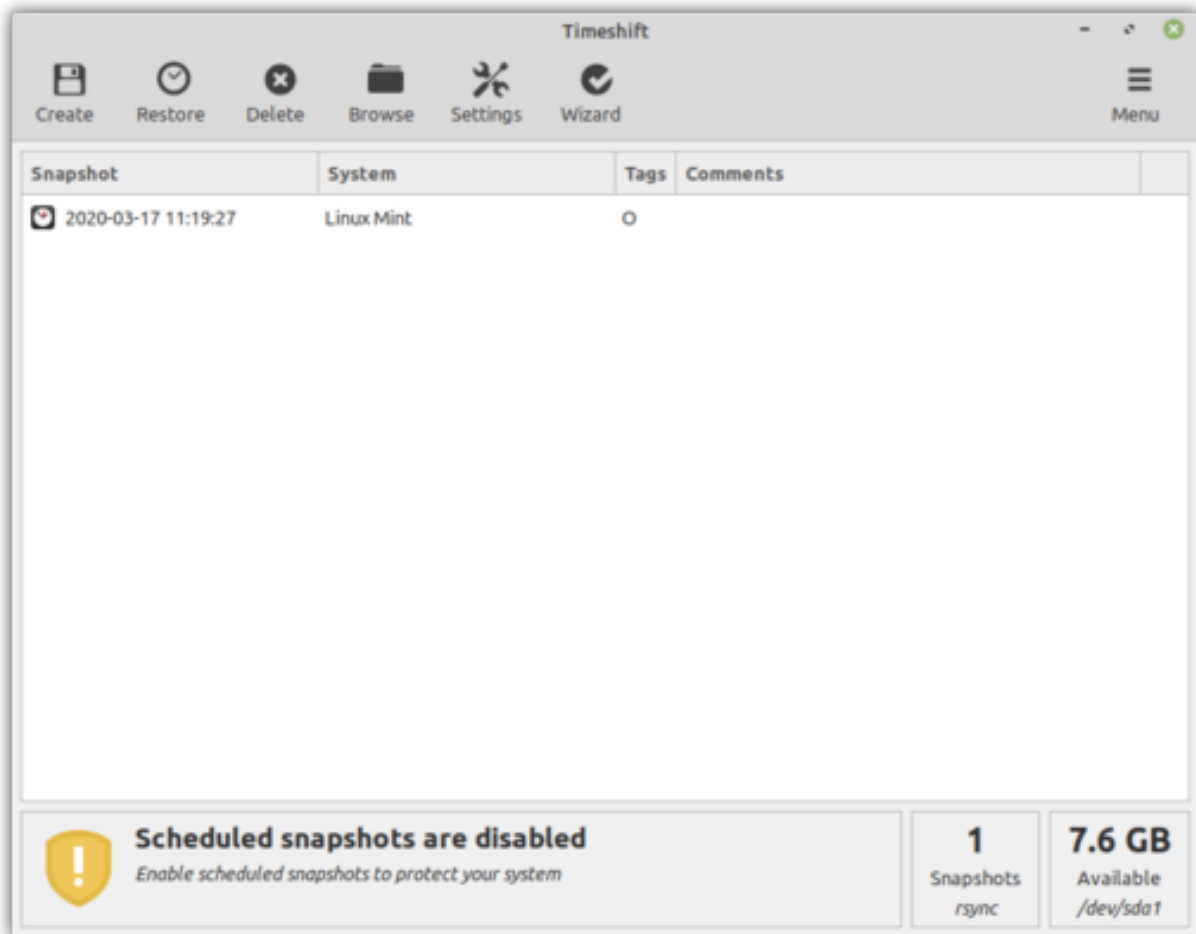
To apply all updates:

- Launch the Update Manager with *Menu* → *Administration* → *Update Manager*.
- Press the *Refresh* button to update the cache.
- Press the *Select All* button to select all updates.
- Press the *Install Updates* button.
- Follow the instructions from the Update Manager and install all updates until the manager tells you the system is up to date.
- Reboot the computer.

### 4.2.2 Create a system snapshot

If anything breaks or if anything goes wrong during the upgrade, you can go back in time and revert all changes by restoring your latest system snapshot. Whatever happens, you're covered. You'll be able to restore your operating

system to this current state, either from within Linux Mint, or by launching Timeshift from a live Mint session (live DVD or live USB).



To create a system snapshot:

- Launch Timeshift with *Menu* → *Administration* → *Timeshift*.
- Follow the wizard to select a destination for your snapshots.
- In the toolbar, click on the *Create* button to make a manual snapshot of your operating system.

### 4.2.3 Purge PPAs and 3rd party repositories

PPAs or 3rd party repositories can introduce issues during the upgrade if the versions of the packages they provide are higher than in Linux Mint 20. This can result in unmet dependencies, held packages or resolver issues.

To purge 3rd party packages follow these steps:

- Launch the Software Sources tool from *Menu* → *Administration* → *Software Sources*.
- Open the *Additional repositories* tab and disable all additional repositories.
- Open the *PPA* tab and disable all PPAs.
- Click on the button to refresh your APT cache.

- Open the *Maintenance* tab and click on *Downgrade Foreign Packages*.
- Select all foreign packages and click *Downgrade*.
- Click on *Remove Foreign Packages*.
- Select all foreign packages and click *Remove*.
- Reboot the computer.
- Create another Timeshift snapshot.

---

**Note:** This step is optional but it is strongly recommended. Some PPAs are perfectly fine, some aren't. Some only add packages and don't impact the upgrade process, others introduce dependencies which cannot be resolved.

You can leave some foreign packages installed or in their 3rd party version and try to upgrade if you want. If it works, then great. If it doesn't, you can always restore the previous snapshot and follow the steps above to purge them before trying again.

---

## 4.3 Upgrade

### 4.3.1 Install the upgrade tool

To install the upgrade tool, open a terminal and type:

```
apt install mintupgrade
```

### 4.3.2 Check the upgrade

To simulate an upgrade, open a terminal and type:

```
mintupgrade check
```

Then follow the instructions on the screen.

This command temporarily points your system to the Linux Mint 20 repositories and calculates the impact of an upgrade.

Note that this command doesn't affect your system. After the simulation is finished, your original repositories are restored.

The output shows you if the upgrade is possible, and if it is, which packages would be upgraded, installed, removed and kept back.

---

**Note:** It is extremely important that you pay close attention to the output of this command.

Keep using **mintupgrade check** and do not proceed to the next step, until you're happy with the output.

---

---

**Hint:** If this steps fails half-way through type **mintupgrade restore-sources** to go back to your original APT configuration.

---

### 4.3.3 Download the package updates

To download the packages necessary to upgrade, type the following command:

```
mintupgrade download
```

Note that this command doesn't actually perform the upgrade itself, but just downloads the packages.

### 4.3.4 Apply the upgrades

**Warning:** This step is non-reversible. Once you perform it, the only way to go back is by restoring a system snapshot.

To apply the upgrades, type the following command:

```
mintupgrade upgrade
```

### 4.3.5 Downgrade foreign packages

Some of your packages might have a lower version in Linux Mint 20 than in Linux Mint 19.3. To guarantee they function properly, they need to be downgraded.

- Launch the Software Sources tool from *Menu* → *Administration* → *Software Sources*.
- Open the *Maintenance* tab and click on *Downgrade Foreign Packages*.
- Select all foreign packages and click *Downgrade*.

### 4.3.6 Delete foreign packages

Some packages no longer exist in Linux Mint 20 and can safely be removed.

- Launch the Software Sources tool from *Menu* → *Administration* → *Software Sources*.
- Open the *Maintenance* tab and click on *Remove Foreign Packages*.
- With the exception of packages you want to keep, select all foreign packages and click *Remove*.

## 4.4 Troubleshooting

### 4.4.1 Installing mintupgrade

If you can't find *mintupgrade* in the repositories, switch to the default Linux Mint mirror and refresh the APT cache.

### 4.4.2 Restoring a snapshot

A known issue affects Timeshift. When restoring a snapshot, if the *Disclaimer* window is empty, wait for about 2 minutes for the text to appear. Once the disclaimer text is there you can press *Next* and restore your snapshot. Pressing *Next* before the text appears results in a failure to restore. If you did, reboot and try to restore again.

### 4.4.3 Skipping the timeshift requirement

If you're using another snapshot tool and would rather not use Timeshift, you can skip the Timeshift requirement with the following command:

```
echo "{}" | sudo tee /etc/timeshift.json
```

Don't forget to remove that file after the upgrade if you want Timeshift to work properly.

### 4.4.4 Computer freeze

On some computers the upgrade can be quite intensive and it can temporarily freeze the desktop. This can last for up to 10 minutes at times or even take hours on slow computers. This is OK, be patient and give it time.

If this becomes a problem, logout completely, drop to console with `CTRL+ALT+F2` and run **mintupgrade upgrade** from there instead.

### 4.4.5 Broken boot

If the computer no longer boots, boot from the live Linux Mint 19.3 ISO.

From the live session, launch *Boot Repair* and use it to fix the boot sequence.

If this doesn't work, boot from the live Linux Mint iso again, and launch *Timeshift*.

Timeshift is able to scan your drives from the live session and restore your snapshot from there.

### 4.4.6 Boot warnings

You can ignore boot warnings related to *ACPI* or *initramfs unpacking*. They're cosmetic and do not affect the boot sequence.

## 4.5 Notes

The upgrade overwrites files in */etc/* with default configuration files. You can restore files individually by the Timeshift snapshot you made prior to upgrading.

To restore your lightDM settings, run the Login Window configuration tool (**sudo lightdm-settings**).

## 4.6 Alternatives

If you cannot upgrade perform a [fresh installation](#).

Generic instructions on [fresh upgrades](#) are also available.