The simplest way to speed up Linux startup: Disable unnecessary services.

More and more operating systems are becoming cumbersome with numerous background services. While these may not use much CPU time, they do increase boot times and RAM usage.

More and more operating systems are becoming cumbersome with numerous background services. While these may not use much CPU time, they increase boot times and RAM usage . Fortunately, open-source operating systems allow you to do what you want. This means you can disable or remove anything you don't need. This article will guide you on how to disable unnecessary services to improve Linux boot times.

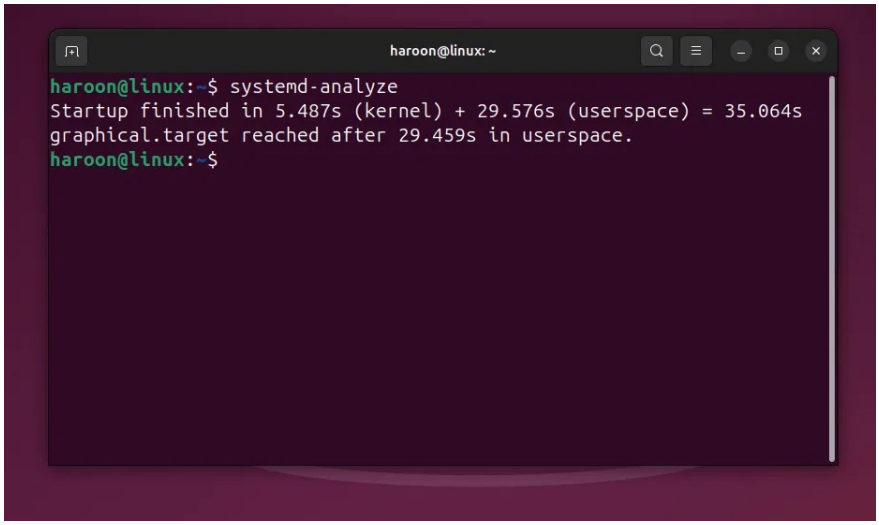
Note : This article guides you on how to disable services, not delete them. Disabling services carries less risk of causing permanent damage. And you can revert to the original state by simply re-enabling a service if you notice a useful function has stopped working.

Analyze the load time of each service.

Most Linux-based operating systems use Systemd by default. Among the utilities it includes, there's a program that allows you to analyze your system's boot speed. Specifically, it shows you the total time it takes to boot and the load time of each service. Note that some services are loaded in parallel. So, if one service takes 2 seconds to load and another takes 3 seconds, that doesn't necessarily mean the total takes 5 seconds. The time could be much less.

Open a terminal emulator and enter this command:

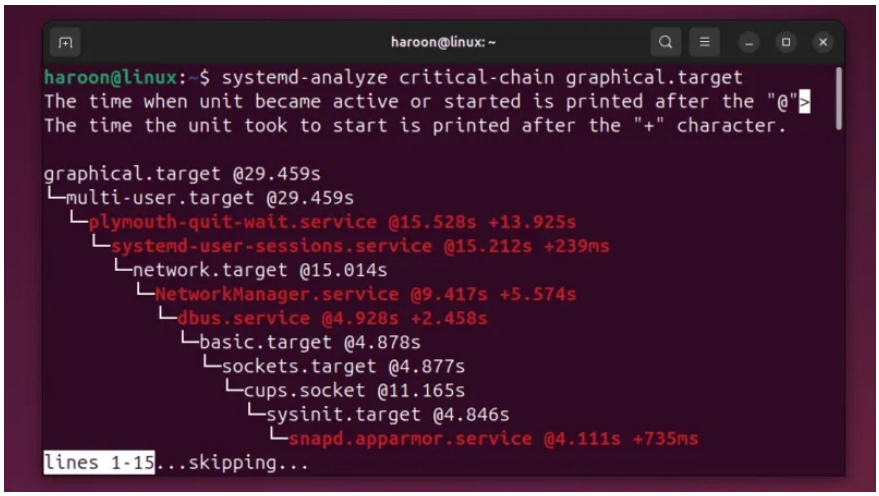
```
systemd-analyze
```



This command shows the time it takes for the Linux kernel and core system services to initialize. It does not take into account the time it takes for on-screen animations or user applications to complete their loading process.

However, you can check the time it takes for the graphical interface to initialize using this command:

```
systemd-analyze critical-chain graphical.target
```

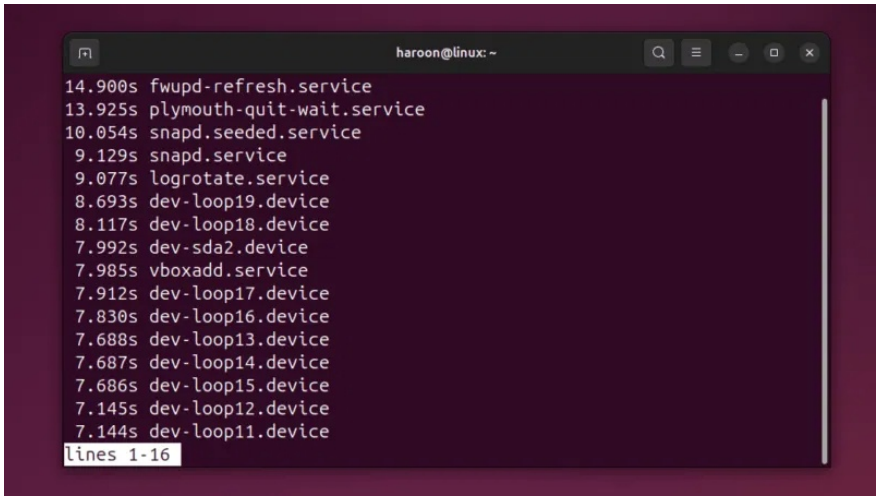


This command shows the sequence of services leading to the achievement of the graphics goal. While useful, this represents a technical milestone rather than the point at which the desktop is completely idle, as some background components may continue loading afterward.

Note : Programs that automatically launch in the desktop environment are usually managed through desktop-specific startup settings or systemd user services.

Finally, perhaps the most useful command for the purposes of this guide is:

```
systemd-analyze blame
```



You can navigate the list using the arrow keys or **PAGE UP** and **PAGE DOWN** . Press **q** to exit.

Use Systemctl to disable unnecessary services.

As you can see in the previous image, some services like snapd take a few seconds to load. On an SSD , this is negligible. But on a traditional hard drive, these times will amount to several seconds, and eventually they will accumulate significantly.

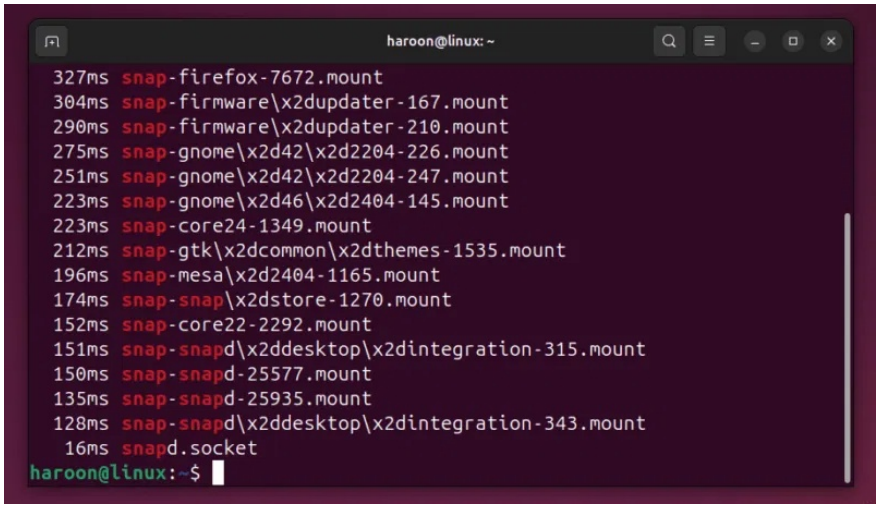
Assuming you don't need the snapd service, which provides access to snap applications packaged within containers, you can disable it using this command:

```
sudo systemctl disable snapd.service
```

After a restart, you may still see snapd start under certain conditions. This happens because snapd uses socket activation, allowing it to start on demand even if the service itself has been disabled.

You can identify the relevant units by:

```
systemd-analyze blame | grep snap
```



This helps you identify which Snap-related units are contributing to the startup activity.

Handle services enabled by sockets.

Some services, including snapd, use socket activation. This means the service can still start when something tries to communicate with it, even if it's disabled.

If snapd continues to appear after a restart, related units such as snapd.socket or snapd.seeded.service may be the cause. In such cases, hiding the service and its socket would be more effective:

```
sudo systemctl mask snapd.service sudo systemctl mask snapd.socket
```

Hiding will completely block the service so that dependencies or sockets cannot start it. However, use this method with caution, especially on distributions that rely on Snap packages for system or application updates.

Additionally, be careful not to confuse "disabling" and "hiding" in this context. Disabling a service tells systemd not to automatically start it. Hiding goes further by preventing the service from starting at all. In most cases, disabling is sufficient. Hiding (masking) is useful for services that keep restarting or are clearly unnecessary on the system.

Furthermore, depending on the configuration, you may find other services that you can safely disable, such as:

```
avahi-daemon.service ModemManager.service thermald.service
```

However, always make sure you thoroughly research a service before disabling it. What's unnecessary on the desktop might be essential on a laptop or server.