



Firefox contained in Flatpak vs Snap comparison

Updated: 2020-04-14

Published: 2019-12-19

11–14-minute read

[Share](#)

The Linux desktop has seen great advances in desktop app containerization and process-isolating sandbox-technologies. Keeping programs from getting hold of each other's environments and files can greatly improve security if something were to go wrong with a program. Flatpak and Snap are the two leading implementations on the desktop.

I've previously praised [Skip to main content](#) get when [playing Steam games inside a Flatpak container](#). I'm overdue to extend that added layer of protection to the web browser. I believe I'll sleep better at night knowing there was an additional security layer between my web browser and the rest of my system.

To this end, I wanted to compare the current versions of Firefox Flatpak and Firefox Snap on Fedora Linux 31. Due to the nature of containerized apps, you can expect my experiences to translate to other Linux distributions.

I'll test Firefox 71 from the Fedora Flatpak repository, and Firefox 71 stable from the Canonical Snapcraft Store. I'll refer to these as "Fedora Flatpak" and "Firefox Snap" throughout this article. Please note that you can use alternative Flatpak repositories such as Flathub. These alternate repositories may come different variants of Firefox with other default limitations and capabilities.

Update (2020-04-14): There's now an official version of Firefox 75 packaged with Flatpak from Mozilla. It's published in the Flathub repository. It eliminates the performance problems mentioned in this article, and ships with an MPEG-4/H.264 decoder. The article has been updated with this version. It's referred to as "Mozilla Flatpak" throughout the article to differentiate it from the version from Fedora Linux.

The below table is a quick comparison of the features and limitations I'll go through in greater detail in this article.

	Firefox 75	Firefox 71	
Feature	Mozilla Flatpak	Fedora Flatpak	Snap
File system protection			
Home directory	~/Downloads only	Read-only, can revoke	Read-only, irrevocable
/etc, /var	Sandboxed		Unrestricted
Device access			
USB/U2F security token	Unrestricted access, revokable	Blocked	Allowed
Webcam / mic		Blocked	Allowed
Desktop integration			
UI & web font	Skip to main content		FreeMono-only
HiDPI (4 K) support	Good		Tiny mouse cursor
Notifications	FreeDesktop		Custom
Multimedia			
H.264 codec	Built-in	Unsupported	Built-in

	Firefox 75	Firefox 71	
Feature	Mozilla Flatpak	Fedora Flatpak	Snap
Performance			
UI responsiveness	Consistently fast		Some slow-downs
Start-up time	700 ms	720 ms	~11 sec
Speedometer 2	-0 %	-24,9 %	-4,7 %
JetStream 2	-0,7 %	-11,3 %	-4,6 %
MotionMark 1.1	-5,8 %	-65,2 %	-10,4 %

File system access restrictions

Firefox Flatpak sandboxes system-level directories like `/etc/` and `/var/`. Firefox Snap has the same permissions to system-level directories as the user.

Mozilla Flatpak only has access to `~/Downloads/`.

By default, Fedora Flatpak has read-access to the entire home directory. It also has read and write access to `~/.mozilla/`, `~/.cache/firefox/`, and `~/Downloads/`. This is where Firefox normally keeps its user data plus the default ephemeral downloads directory.

You can revoke Fedora Flatpak's access to the home directory (while maintaining specific permissions mentioned above) by running the below command. You can still access files in your home directory using the File: Open dialog and when choosing which files to upload.

```
flatpak override --user \
  --nofilesystem=home org.mozilla.Firefox
```

Firefox Snap has irrevocable read-only access non-hidden files and directories in the home directory. Hidden-files are files and directories whose name starts with a dot.

[Skip to main content](#)

In other words, it restricts access to sensitive areas like `~/.ssh/` and `~/.bitcoin/` but does nothing to restrict access to your `~/Documents/` and `~/Pictures/`. At least one of these directories have been the target of [an old Firefox exploit](#).

Anything it wants to write to the home directory or read from a dot-directory is redirected to `~/snap/firefox/common/`.

Both variants sandboxes `/tmp/` (boot-cycled temporary files) but only Flatpak sandboxes `/var/tmp/` (reboot-persistent ephemeral files).

Firefox Snap has the most restrictive default home directory protection out of the two. However, only Flatpak limits access to system directories. Flatpak can also be configured to have the most restrictive configuration by revoking access to the home directory.

Device access

Your web browser will need access to your webcam and microphone for video conferencing and the like. It will also need access to U2F devices if you've set-up any online accounts with two-factor authentication with a [USB](#) security key token.

Snap comes with dedicated hardware device access policies for specific device categories such as [USB](#) security key tokens and webcams. Firefox Snap, by default, has access to your webcam, microphone, and [USB](#) security tokens.

Mozilla Firefox has full access to all hardware devices. Fedora Flatpak is blocked from accessing any of your hardware devices. Blocking offers stronger security and privacy protections but comes at a loss of functionality. This can be crippling if use services that require U2F like [Google Advanced Protection](#).

Flatpak only has two access policies for devices: allow access to all devices or block access to all devices. You can flip this policy using the following command:

```
flatpak override --u
--device=all org.mozilla.firefox
```

[Skip to main content](#)

You can change the device-access policies under both Flatpak and Snap. You'll need to memorize commands and restart the

browser if you want to turn these on and off depending on your needs. This is probably too much of a bother for most people.

This comes down to how paranoid you want to be about your device security. Blocking the browser from accessing your camera and microphone may be considered a net benefit unless you use them every day.

Firefox Snap has out-of-the-box support for experience- and security-enhancing hardware devices. This can be a double-edged sword as it also means a potential exploit could get access to your device's microphone and camera. Flatpak doesn't have granular controls for device access. You either give it access to all your devices or none at all (the default).

Desktop integration

Flatpak has the clear advantage when it comes to desktop environment integration. It also uses the expected default system font for user interface elements and looks like any other GTK+ apps. It looks and behaves as Firefox running unconstrained directly on the host system.

Firefox for Flatpak integrates with the FreeDesktop notification system (used by GNOME and KDE) for web notifications. The Snap variant doesn't integrate with the system-wide notification system.

Its custom notification pop-overs doesn't respect system settings for notifications (quiet hours, etc..) The custom notifications are easy to miss as they aren't shown on top of other windows.

Firefox Snaps looks good as well but it has a few minor styling differences from other GTK+ apps. One of the more severe issues is that it uses FreeMono Regular (a fixed-width/monospace font) instead of a sans-serif font.

The font issue extends on to the web as well. Every webpage that doesn't supply custom Web Fonts uses FreeMono Regular instead. This causes many website designs to break and text

legibility goes right out the window compared to a sans-serif font. This issue is fixed in the beta release channel.



Flatpak (left) vs Snap (right) mouse cursor sizes.

You can compare how fonts look in Flatpak versus Firefox Snap in the above video. You'll also notice that the Snap version has an issue with the mouse cursor shrinking when entering the window.

The cursor changes the pointer-icon when moving it over the Firefox window. It also shrinks to half the expected size on a high-definition (HiDPI) display. Unfortunately, this isn't just a cosmetic problem. You need to move the mouse twice the distance to get across the screen.

Multimedia

MPEG-4 (H.264/AAC) is a popular proprietary multimedia codec. Most videos you watch online — outside of YouTube — will probably be encoded with this codec. This is also the most common codec for live-streaming video.

Firefox Snap comes with H.264 support built-in. So does Mozilla Flatpak. You shouldn't have any problems watching videos on the web.

The main Fedora Linux package repository doesn't come with H.264 video codec support for applications. This codec is patent- and license-encumbered which prevents Fedora Linux from distributing it by default. The repository has the same limitation.

[Skip to main content](#)

However, you can install additional codecs on Fedora Linux. The Flatpak sandbox prevents Firefox from using these extra multimedia codecs. This is the sandbox doing its job. Unfortunately, you can't install additional codecs into the Flatpak sandbox environment.

The lack of H.264 support is probably the key issue that will get someone to stop using Fedora Flatpak in favor of the unconstrained system default. Luckily, Mozilla Flatpak has built-in support for the codec.

Performance

Every Snap package I've tested have been slow to launch. It generally takes 5–12 seconds from you start a program until it appears on the screen. Firefox Snap takes roughly 11 seconds to start.

Earlier this year, [Snap fixed a font-caching issue that caused slow start-up times](#). I'm running a version with that fix. However, Firefox Snap's font-cache is notably still broken. This is also what causes the problem with the monospace font. This could still be the root cause of the start-up performance issue.

For comparison, Flatpak starts up in less than a second. Flatpak also feels more responsive when scrolling, switching tabs, and other UI operations. Firefox Snap has noticeable lag when performing the same operations.

The [Speedometer 2 benchmark](#) tries to measure the responsiveness of web apps. Fedora Flatpak is almost 25% slower than Firefox installed unconstrained on the system. Firefox Snap is only penalized 25% of that figure.

The [JetStream 2 benchmark](#) tests JavaScript and WebAssembly performance. Firefox ~~Snap has a 460% performance penalty compared to Firefox running directly on the host system.~~ Fedora Linux Flatpak is 11,3% slower.

Fedora Flatpak tanks in the [MotionMark 1.1 benchmark](#). It's a full 65,2% slower than Firefox running on the host system.

Firefox Snap is only 10,4% slower. Both variants see the biggest difference in the Canvas painting tests. Fedora Flatpak nearly half the speed (-98,63 %) of Firefox running on the host system.

Perceived performance undoubtedly *feels* better in Fedora Flatpak despite Firefox Snap crushing it in both graphical and computational synthetic benchmarks.

Conclusions

You may want to stick with an unconstrained installation of Firefox if you're using older hardware that would be severely affected by the decreased performance. The relaxed sandboxing of Firefox Snap would still protect some of your files. Fedora Flatpak set to block home directory access will likely yield the highest level of protection, however.

Firefox Snap is the clear winner when it comes to capabilities with support for video-conferencing, USB security token, and video playback. These are arguably attack-surfaces you may be better without. You can use a separate browser for these tasks only. It comes down to whether you need these features regularly or not.

The performance story is quite interesting. The Flatpak sandbox hurts Firefox's web performance badly. But it's the Snap version that *feels* slow to use. The perceived performance may vary greatly depending on your hardware and which type of delays annoys you.

Update (2019-12-20): Alexander Larsson from Red Hat pointed out that Fedora Flatpak isn't built with PGO. This could account for the performance difference. It's also something concrete that Fedora Linux can improve on to help bring performance on par with Firefox running directly on the host.

[Skip to main content](#)

I think I'll migrate to Fedora Flatpak at the start of the new year.

I may find the time to test [Qubes OS](#) first, though. It's a Linux distribution where *many tasks of the operating system* is independently isolated from everything else. Although, I'm not sure whether I'd sleep better or lose sleep over jumping feet-first into something as complex as Qubes OS. ■

Thanks to Maciej Borzecki from Canonical Ltd. for guiding me through troubleshooting the font and mouse cursor issues with Firefox Snap.

I used Firefox 71, Flatpak 1.4.3, and snapd 2.42.2 on Fedora Linux 31 for this article. The article was updated to compare Firefox 75 from Flathub (the official Mozilla release) and Firefox 75 from the Firefox repository.

[Buy Me A Coffee](#)[Read 141 comments](#)

[Web Browsers](#)[Linux](#)[Security](#)[Codecs](#)

Abbreviations

HiDPI	high-definition
PGO	profile-guided optimization
USB	Universal Serial Bus

Related reading

[Skip to main content](#)



Steam for Linux is more secure
than on MacOS and Win32



How to back up Flatpak
user-data and apps



How to install additional spell
check dictionaries for Flatpak apps



Flatpak, Steam Cloud,
XDG Base, and why Linux
doesn't sync progress

Get the weekly Ctrl blog newsletter

Email:

[About](#)

[Privacy Policy](#)

[Colophon](#)

[Licensing](#)

The image “Flatpak and Snap” is used under a CC0 1.0 License. Ctrl blog by © 2025
Daniel Aleksandersen.

Hosting by Hetzner and Linode. CDN by Bunny.