Firefox/Tweaks

< Firefox

This page contains advanced Firefox configuration options and performance tweaks.

Contents

Related articles

Firefox

Browser plugins

Firefox/Profile on RAM

Firefox/Privacy

Performance

Improving Firefox's performance is divided into parameters that can be inputted while running Firefox or otherwise modifying its configuration as intended by the developers, and advanced procedures that involve foreign programs or scripts.

Note: Listed options may only be available for the latest version of Firefox.

This section contains advanced Firefox options for performance tweaking. For additional information see **these MozillaZine articles (http://kb.mozillazine.org/Category:Tweaking_preferences)**.

Change Performance settings

Firefox 56 automatically uses settings based on the computer's hardware specifications **[1] (https://support.mozilla.org/en-US/kb/performance-settings)**.

Adjusting these settings can be done in Preferences or by changing the `dom.ipc.processCount` value to `1-7` and `browser.preferences.defaultPerformanceSettings.enabled` to `false` manually in `about:config`.

However you may want to manually adjust this setting to increase performance even further or decrease memory usage on low-end devices.

In this case the **Content process limit** for the current **user** has been increased to *4*:

```
$ ps -e | grep 'Web Content'
```

```
13991 tty1     00:00:04 Web Content
14027 tty1     00:00:09 Web Content
14031 tty1     00:00:20 Web Content
14040 tty1     00:00:26 Web Content
```

Enable OpenGL Off-Main-Thread Compositing (OMTC)

Basic software OMTC is enabled by default.

To check if OpenGL OMTC is enabled, go to `about:support` and under the "Graphics" section look for "Compositing". If it reports "Basic", OpenGL OMTC is disabled; if it reports "OpenGL" it is enabled.

If OpenGL OMTC is disabled, you can force-enable it by going to `about:config` and enabling `layers.acceleration.force-enabled`. Restart Firefox for changes to take effect.

Warning: If OpenGL OMTC is disabled for a specific hardware, it may be due to stability issues, high system resources consumption, driver bugs or a number of different variables,

and so instead of speeding things up it might slow them down. Proceed with force-enabling it at your own risk, benchmark if you aren't sure.

For more information on OMTC in Firefox read here: **https://wiki.mozilla.org/Platform/GFX/OffMainThreadCompositing**

Enable Accelerated Azure Canvas

Warning: Accelerated Azure Canvas may cause invalid/corrupt rendering of images on unsupported devices and/or drivers, see **#Enable OpenGL Off-Main-Thread Compositing (OMTC)**.

Go to `about:config`, accept the warning, right click and create a new boolean value. Set the name as `gfx.canvas.azure.accelerated` and set it to `true`.

To verify restart Firefox then go to `about:support` and search for `AzureCanvasAccelerated` which should be set to *1*.

The acceleration efficiency can be tested by comparing the speed of a **javascript demo (http://js1k.com/2016-elemental/demo/2445)** with and without the setting.

Stop urlclassifier3.sqlite from being created again

Removing all `urlclassifier*` files can prevent the use of megabytes of storage in your firefox profile. If you remove all the `urlclassifier*` files, you may find out that `urlclassifier3.sqlite` keeps growing again after a certain time. Here is a simple solution to avoid it for now and ever.

```
$ cd ~/.mozilla/firefox/<profile_dir>
$ echo "" > urlclassifier3.sqlite
$ chmod 400 urlclassifier3.sqlite
```

This effectively makes the file empty and then read-only so Firefox cannot write to it anymore.

Turn off the disk cache

Every object loaded (html page, jpeg image, css stylesheet, gif banner) is saved in the Firefox cache for future use without the need to download it again. It is estimated that only a fraction of these objects will be reused, usually about 30%. This because of very short object expiration time, updates or simply user behavior (loading new pages instead of returning to the ones already visited). The Firefox cache is divided into memory and disk cache and the latter results in frequent disk writes: newly loaded objects are written to memory and older objects are removed.

The disk cache can be turned off by enabling *Preferences > Privacy & Security > Cached Web Content - Override automatic cache management* and setting to `0` *Limit cache to ... MB of space*.

An alternative approach is to use `about:config` settings:

- Set `browser.cache.disk.enable` to `false`
- Verify that `browser.cache.memory.enable` is set to `true`, more information about this option can be found in the **browser.cache.memory Mozilla article (http://kb.mozillazine.org/Browser.cache.memory.enable)**
- Add the entry (*right click* > *new* > *integer*) `browser.cache.memory.capacity` and set it to the amount of KB you want to spare, or to `-1` for **automatic (http://kb.mozillazine.org/Browser.cache.memory.capacity#-1)** cache size selection (skipping this step has the same effect as setting the value to `-1`)

This method has some drawbacks:

- The content of currently browsed webpages is lost if the browser crashes or after a reboot, this can be avoided using **anything-sync-daemon** or any similar periodically-syncing script so that cache gets copied over to the drive on a regular basis
- The settings need to be configured for each user individually

Longer interval between session information record

Firefox stores the current session status (opened urls, cookies, history and form data) to the disk on a regular basis. It is used to recover a previous session in case of crash. The default setting is to save the session every 15 seconds, resulting in frequent disk access.

To increase the save interval to 10 minutes for example, change in `about:config` the setting of `browser.sessionstore.interval` to `600000`

To disable completely this feature, change `browser.sessionstore.resume_from_crash` to `false`

Referrer header control

The HTTP `Referer` header can be extensively configured via `about:config`. See **Security/Referrer (https://wiki.mozilla.org/Security/Referrer)** on the Mozilla wiki for the available preferences.

Defragment the profile's SQLite databases

Warning: This procedure may damage the databases in such a way that sessions are not saved properly.

Firefox keeps bookmarks, history, passwords in SQLite databases. SQLite databases become fragmented over time and empty spaces appear all around. But, since there are no managing processes checking and optimizing the database, these factors eventually result in a performance hit. A good way to improve start-up and some other bookmarks and history related tasks is to defragment and trim unused space from these databases.

You can use **profile-cleaner (https://aur.archlinux.org/packages/profile-cleaner/)**[AUR] to do this, while Firefox is **not** running:

profile-cleaner usage example:

SQLite database	Size Before	Size After	% change
urlclassifier3.sqlite	37 M	30 M	19 %
places.sqlite	16 M	2.4 M	85 %

Firefox provides a tool to defragment and optimize the places database, which is the source of most slowdowns and profile corruptions. To access this tool, open the `about:support` page, search for `Places Database` and click the `Verify Integrity` button.

Cache the entire profile into RAM via tmpfs

If the system has memory to spare, `tmpfs` can be used to **cache the entire profile directory**, which might result in increased Firefox responsiveness.

Turn off sponsored content and tiles

In `about:config`, set the string value to a blank for both of these: `browser.newtabpage.directory.source` and `browser.newtabpage.directory.ping`. Consider also disabling the tile feature from the tools on a new tab page. A **Wireshark** session demonstrates the level of chatter created by these features.

Enable Electrolysis

Note: Electrolysis should always be enabled in Firefox 57 and later

Electrolysis (multi-process) may be enabled to improve performance and security by setting `browser.tabs.remote.autostart` to *true* in `about:config`. It may be needed to force-enable Electrolysis **[2] (https://wiki.mozilla.org/Electrolysis#Force_Enable)**, although this is generally not recommended and may cause issues.

To check if Electrolysis is enabled, go to `about:support` and under the "Application Basics" section look for "Multiprocess Windows". If it reports "0/1 (Disabled)", Electrolysis is disabled; if it reports "1/1 (Enabled by user)" it is enabled. Note that the given numbers */* indicate the number of open Firefox windows, e.g. 0/2 meaning non of the two Firefox-windows are using Electrolysis, and 2/2 means it is enabled for both windows.

Disable Pocket

If you don't use the Pocket-service, you may want to disable it by setting `extensions.pocket.enabled` to *false* in `about:config`.

Appearance

Fonts

See the main article: **Font configuration**

Configure the DPI value

Modifying the following value can help improve the way fonts looks in Firefox if the system's DPI is below 96. Firefox, by default, uses 96 and only uses the system's DPI if it is a higher value. To force the system's DPI regardless of its value, type `about:config` into the address bar and set `layout.css.dpi` to **0**.

Note that the above method only affects the Firefox user interface's DPI settings. Web page contents still use a DPI value of 96, which may look ugly or, in the case of high-resolution displays, may be rendered too small to read. A solution is to change `layout.css.devPixelsPerPx` to system's DPI divided by 96. For example, if your system's DPI is 144, then the value to add is 144/96 = 1.5. Changing `layout.css.devPixelsPerPx` to **1.5** makes web page contents use a DPI of 144, which looks much better.

See also **HiDPI#Firefox** for information about HiDPI displays and **[3] (https://www.sven.de/dpi/)** for calculating the DPI.

Default font settings from Microsoft Windows

Below are the default font preferences when Firefox is installed in Microsoft Windows. Many web sites use the Microsoft fonts.

```
Proportional: Serif Size (pixels): 16
Serif: Times New Roman
Sans-serif: Arial
Monospace: Courier New Size (pixels): 13
```

General user interface CSS settings

Firefox's user interface can be modified by editing the `userChrome.css` and `userContent.css` files in `~/.mozilla/firefox/<profile_dir>/chrome/` (*profile_dir* is of the form *hash.name*, where the *hash* is an 8 character, seemingly random string and the profile *name* is usually *default*).

Note: The `chrome/` folder and `userChrome.css` / `userContent.css` files may not necessarily exist, so they may need to be created.

This section only deals with the `userChrome.css` file which modifies Firefox's user interface, and not web pages.

Change the interface font

The setting effectively overrides the global GTK+ font preferences, and does not affect webpages, only the user interface itself:

~/.mozilla/firefox/<profile_dir>/chrome/userChrome.css

```
* {
    font-family: "FONT_NAME";
}
```

Hide button icons

Enables text-only buttons:

~/.mozilla/firefox/<profile_dir>/chrome/userChrome.css

```
.button-box .button-icon {
    display: none;
}
```

Hiding various tab buttons

These settings hide the arrows that appear to the horizontal edges of the tab bar, the button that toggles the "all tabs" drop-down list, and the plus sign button that creates a new tab.

~/.mozilla/firefox/<profile_dir>/chrome/userChrome.css

```
/* Tab bar */

.tabbrowser-strip *[class^="scrollbutton"] {
    /* Hide tab scroll buttons */
    display: none;
}

.tabbrowser-strip *[class^="tabs-alltabs"] {
    /* Hide tab drop-down list */
    display: none;
```

```
}

.tabbrowser-strip *[class^="tabs-newtab-button"] {
    /* Hide new-tab button */
    display: none;
}
```

Horizontal tabs

To place the tab bar horizontally stacked along the sides of the browser window:

```
~/.mozilla/firefox/<profile_dir>/chrome/userChrome.css
```

```
/* Display the tabbar on the left */
#content > tabbox {
    -moz-box-orient: horizontal;
}

.tabbrowser-strip {
    -moz-box-orient: vertical;
    /*
     * You can set this to -moz-scrollbars-vertical instead,
     * but then the scrollbar will *always* be visible.  this way
     * there is never a scrollbar, so it behaves like the tab bar
     * normally does
     */
     overflow: -moz-scrollbars-none;
}

.tabbrowser-tabs {
    -moz-box-orient: horizontal;
    min-width: 20ex;   /* You may want to increase this value */
    -mox-box-pack: start;
    -moz-box-align: start;
}

.tabbrowser-tabs > hbox {
    -moz-box-orient: vertical;
    -moz-box-align: stretch;
    -moz-box-pack: start;
}

.tabbrowser-tabs > hbox > tab {
    -moz-box-align: start;
```

```
    -moz-box-orient: horizontal;
}
```

Hide window border and title bar

Install the **Hide Caption Titlebar Plus (https://addons.mozilla.org/firefox/addon/hide-caption-titlebar-plus-sma/)** extension and set the following settings (leaving other settings to default):

Option	Value
Show Custom Caption/TitleBar	Never
Activate custom borders and corner resizers	Deactivate
Enable Customizable Buttons (min,max,close)	Using a Glass-like window background
Custom Minimize, Max, Close Buttons	Auto. Current theme's skin (fixed position)
Drag Fx window using Tab-bar background	Enable
Alternative hide-titlebar feature	Use

The extension **Classic Theme Restorer (https://addons.mozilla.org/firefox/addon/classicthemerestorer/)** provides more tweaking options to get the best result (e.g. set tab height to 28px, enable/disable toolbars/buttons, etc.).

Auto-hide Bookmarks Toolbar

```
~/.mozilla/firefox/<profile_dir>/chrome/userChrome.css
```

```
#PersonalToolbar {
    visibility: collapse !important;
}
```

```
#navigator-toolbox:hover > #PersonalToolbar {
    visibility: visible !important;
}
```

Remove sidebar width restrictions

```
~/.mozilla/firefox/<profile_dir>/chrome/userChrome.css
```

```
/* remove maximum/minimum  width restriction of sidebar */
#sidebar {
    max-width: none !important;
    min-width: 0px !important;
}
```

Unreadable input fields with KDE Breeze Dark theme

If you are using **KDE** desktop in conjunction with Breeze Dark theme, you might find that some input fields have dark background, which makes text unreadable. A solution to this issue is to use a non-dark GTK theme along with a dark Firefox theme. Similarly, by using a dark Firefox theme your browser will look the way you want (requires Firefox 56 or later).

Web content CSS settings

This section deals with the `userContent.css` file in which you can add custom CSS rules for web content. Changes to this file will take effect once the browser is restarted.

This file can be used for making small fixes or to apply personal styles to frequently visited websites. Custom stylesheets for popular websites are available from sources such as **userstyles.org (http://userstyles.org/)**. You can install an add-on such as **superUserContent (https://addons.mozilla.org/firefox/addon/superusercontent/)** to manage themes. This add-on creates the directory `chrome/userContent.css.d` and applies changes to the CSS files therein when the page is refreshed.

Import other CSS files

```
~/.mozilla/firefox/<profile_dir>/chrome/userContent.css
```

```
@import url("./imports/some_file.css");
```

Block certain parts of a domain

```
~/.mozilla/firefox/<profile_dir>/chrome/userContent.css
```

```
@-moz-document domain(example.com) {
    div#header {
        background-image: none !important;
    }
}
```

Add [pdf] after links to PDF files

```
~/.mozilla/firefox/<profile_dir>/chrome/userContent.css
```

```
/* add '[pdf]' next to links to PDF files */
a[href$=".pdf"]:after {
    font-size: smaller;
    content: " [pdf]";
}
```

Block ads

See **floppymoose.com (http://www.floppymoose.com)** for an example of how to use `userContent.css` as a basic ad-blocker.

Mouse and keyboard

Mouse wheel scroll speed

To modify the default values (i.e. speed-up) of the mouse wheel scroll speed, go to `about:config` and search for `mousewheel.acceleration`. This will show the available options, modifying the following:

- Set `mousewheel.acceleration.start` to `-1`.
- Set `mousewheel.acceleration.factor` to the desired number (`10` to `20` are common values).

Mozilla's recommendation for increasing the mousewheel scroll speed is to:

- Set `mousewheel.default.delta_multiplier_y` between `200` and `500` (default: `100`)

Alternatively you can install the **SmoothWheel add-on (http://smoothwheel.mozdev.org/)**.

Pixel-perfect trackpad scrolling

To enable one-to-one trackpad scrolling (as can be witnessed with GTK3 applications like Nautilus), set the `MOZ_USE_XINPUT2=1` **environment variable** before starting Firefox.

If scrolling is undesirably jerky, try enabling Firefox's "Smooth Scrolling" option in Preferences > Advanced.

Enable touchscreen gestures

Make sure `dom.w3c_touch_events.enabled` is either set to 1 (*enabled*) or 2 (*default, auto-detect*).

Run `export MOZ_USE_XINPUT2=1` before launching Firefox. To make this change persistent, add that command to `/etc/profile.d/firefox.sh`.

Mouse click on URL bar's behavior

To make the url bar behaves like in Windows regarding mouse clicks: a single click selects everything, a double click selects a single word until a punctuation sign and a triple click selects everything again, set the following in `about:config`:

```
browser.urlbar.clickSelectsAll; true
browser.urlbar.doubleClickSelectsAll; false
layout.word_select.stop_at_punctuation; true (default)
```

Set backspace's behavior

See **Firefox#Backspace does not work as the 'Back' button**.

Disable middle mouse button clipboard paste

See **Clipboard#Firefox**.

Miscellaneous

Enable additional media codecs

Before continuing, remember there is a reason some of these variables are not enabled by default, e.g. stability, memory leaks, etc. Go to `about:config` and check the following options:

Suggested values

Key	Value	Description
media.mediasource.enabled	true (default)	Enable **Media Source Extensions** (MSE)
media.mediasource.mp4.enabled	true (default)	Enable **MP4** MSE
media.mediasource.webm.enabled	true (default)	Enable **WebM** MSE.
media.mediasource.ignore_codecs	true	Enable H.264 MSE, amongst other things (This boolean key has to be created!)

Widevine and Netflix/Amazon Video

Widevine is a digital rights management tool that Netflix, Amazon Prime Video, and others use to protect their video content.

The first time you visit a Widevine-enabled page Firefox will display a prompt below the address bar asking for permission to install DRM. Approve this and then wait for the "Downloading" bar to disappear, you are now able to watch videos from Netflix, Amazon Prime Video, or any other Widevine protected site.

Note: Please make sure to check the *Play DRM content* option under Content-Preference.

Remove full screen warning

Warning about video displayed in full screen mode ("… is now fullscreen") can be disabled by setting `full-screen-api.warning.timeout` to `0` in `about:config`.

Change the order of search engines in the Firefox Search Bar

To change the order search engines are displayed in:

- Open the drop-down list of search engines and click *Manage Search Engines...* entry.
- Highlight the engine you want to move and use *Move Up* or *Move Down* to move it. Alternatively, you can use drag-and-drop.

"I'm Feeling Lucky" mode

Some search engines have a "feeling lucky" feature. For example, Google has "I'm Feeling Lucky", and DuckDuckGo has "I'm Feeling Ducky".

To activate them, search for `keyword.url` in `about:config` and modify its value (if any) to the URL of the search engine.

For Google, set it to:

```
https://www.google.com/search?btnI=I%27m+Feeling+Lucky&q=
```

For DuckDuckGo, set it to:

```
https://duckduckgo.com/?q=\
```

Secure DNS with DNSSEC validator

You can enable **DNSSEC** support for safer browsing.

Adding magnet protocol association

In `about:config` set `network.protocol-handler.expose.magnet` to `false`. In case it does not exist, it needs to be created, right click on a free area and select *New* > *Boolean*, input `network.protocol-handler.expose.magnet` and set it to `false`.

The next time you open a magnet link, you will be prompted with a *Launch Application* dialogue. From there simply select your chosen torrent client. This technique can also be used with other protocols.

Prevent accidental closing

The **Disable Ctrl-Q Shortcut (https://addons.mozilla.org/firefox/addon/disable-ctrl-q-shortcut/)** extension can be installed to prevent unwanted closing of the browser.

An alternative is to add a rule in your window manager configuration file. For example in **Openbox** add:

```
<keybind key="C-q">
  <action name="Execute">
    <execute>false</execute>
```

```
  </action>
</keybind>
```

in the <*keyboard*> section of your `~/.config/openbox/rc.xml` file.

Note: This will be effective for every application used under a graphic server.

Plugins do not work with latest version

There can be compatibility issues with plugins not working with the latest Firefox version (e.g. **Pentadactyl (http://5digits.org/pentadactyl/index)**). If possible, try installing the nightly/beta builds available, or temporarily **downgrade** Firefox.

Disable Add-on Compatibility Checks (https://addons.mozilla.org/firefox/addon/checkcompatibility/) plugin should take care of spurious compatibility issues when the plugins get disabled, even though they work just fine with the new version.

Jerky or choppy scrolling

Scrolling in Firefox can feel "jerky" or "choppy". A post on **MozillaZine (http://forums.mozillazine.org/viewtopic.php?f=8&t=2749475/)** gives settings that work on Gentoo, but reportedly work on Arch Linux as well:

1. Set `mousewheel.min_line_scroll_amount` to 40

2. Set `general.smoothScroll` and `general.smoothScroll.pages` to **false**
3. Set `image.mem.min_discard_timeout_ms` to something really large such as 2100000000 but no more than 2140000000. Above that number Firefox will not accept your entry and complain with the error code: "The text you entered is not a number."
4. Set `image.mem.max_decoded_image_kb` to at least 512K

Now scrolling should flow smoothly.

Run Firefox inside an nspawn container

Note: on newer systems, you must enable local access to the user xserver with 'xhost local:root'

To run as PID 1

```
# systemd-nspawn --setenv=DISPLAY=:0 \
             --setenv=XAUTHORITY=~/.Xauthority \
             --bind-ro=$HOME/.Xauthority:/root/.Xauthority \
             --bind=/tmp/.X11-unix \
             -D ~/containers/firefox \
             firefox
```

Else rather boot the container, with systemd ideally setting up your networking with **systemd-networkd**:

```
# systemd-nspawn --bind-ro=$HOME/.Xauthority:/root/.Xauthority \
             --bind=/tmp/.X11-unix \
```

```
            -D ~/containers/firefox \
            --network-veth -b
```

Note: There is **a bug (https://github.com/systemd/systemd/issues/7093)** in systemd version 235 that causes /tmp/.X11-unix to disappear from the filesystem when doing this. If you're having trouble, try binding /tmp/.X11-unix read-only instead: `--bind-ro=/tmp/.X11-unix`

Once your container is booted, run the Xorg binary like so:

```
# systemd-run -M firefox --setenv=DISPLAY=:0 firefox
```

Show search matches position in scroll bar

This chrome feature can be achieved via **FindBar Tweak (https://addons.mozilla.org/firefox/addon/findbar-tweak/)** extension.

Disable WebRTC audio post processing

If you are using the PulseAudio **module-echo-cancel**, you probably don't want Firefox to do additional audio post processing.

To disable audio post processing, change the value of the following preferences to `false`:

- `media.getusermedia.aec_enabled` (Acoustic Echo Cancellation)

- `media.getusermedia.agc_enabled` (Automatic Gain Control)
- `media.getusermedia.noise_enabled` (Noise suppression)

Fido U2F authentication

Since version 57, Firefox supports Fido U2F authentication protocol. However, it's disabled by default. To enable it set the following settings to `true` in `about:config`.

- `security.webauth.u2f`
- `security.webauth.webauthn_enable_usbtoken`

Note: Firefox does not inplement the entire U2F protocol **[4] (https://www.yubico.com/2017/11/how-to-navigate-fido-u2f-in-firefox-quantum/)**. Some site might not work correctly

If you're using a Firefox version lower than 57 (e.g. Firefox 52 ESR), U2F isn't supported natively, but there is an **extension (https://addons.mozilla.org/en-US/firefox/addon/u2f-support-add-on/)** to add this functionality.

See also

- **MozillaZine Wiki (http://kb.mozillazine.org/Knowledge_Base)**
- **about:config entries MozillaZine article (http://kb.mozillazine.org/About:config_entries)**

- **Firefox touch-ups that might be desired (http://linuxtidbits.wordpress.com/2009/08/01/better-fox-cat-and-weasel/)**

Retrieved from "https://wiki.archlinux.org/index.php?title=Firefox/Tweaks&oldid=510417"
