

# App configs

- [Firefox RAM usage](#)
- [NextCloud customization](#)
- [Tauri apps with blank screen on Linux](#)

# Firefox RAM usage

## Post found on a forum

-> [Firefox forum](#)

Recently I asked chatgpt about a memory problem I was having when opening more than a hundred tabs and he suggested these two procedures:

1. it is possible to configure Firefox to load tabs only when you click on them, instead of loading them immediately after opening. This setting can help save memory on your computer and improve Firefox's performance. To configure Firefox this way, follow these steps:
  1. Open Firefox and type `about:config` in the address bar.
  2. Press Enter and then click 'I accept the risk!'.
  3. In the search bar, type `browser.tabs.loadDivertedInBackground`.
  4. Double-click the option to set the value to `true`.

From now on, when you open multiple tabs at once, they will not load automatically. Instead, only the active tab will load, while the others will remain in a suspended state until you click on them. When you click on a dropdown tab, it will load immediately. Remember that if you click on a link in a dropdown tab, the link will open in a new tab, which will remain in the dropdown until you click on it. If you prefer that all new tabs be loaded immediately, you can go back to the `about:config` settings and set the 'browser.tabs.loadDivertedInBackground' option to 'false'.

2. It is possible to configure Firefox so that tabs opened from a link do not load automatically.

To do this, follow these steps:

1. Open Firefox and type `about:config` in the address bar and press Enter.
2. Click 'Accept the risk and continue' to confirm that you want to access Firefox's advanced settings.
3. In the search bar, type `browser.tabs.loadInBackground`.
4. Double-click the `browser.tabs.loadInBackground` to change the value to `false`. This will cause tabs opened from a link to not load automatically.

From now on, when you open a link in a new tab, it will be displayed in the background and will not load automatically until you select it.

With these two simple procedures, you can save a lot of memory in Firefox, especially for people who open a lot of tabs like me. And that almost nobody knows, that's why it's important to make these two settings default in the browser, ok?

Thanks.

# NextCloud customization

## Introduction

The solution indicated here works on a [turnkeylinux LXC image for NextCloud](#).

## Change the data location

To change the data location, you need to do a few things:

1. copy the previous storage location to the new one:

```
sudo cp -R /var/www/nextcloud-data/* /path/to/your/data/directory/
```

2. create a `.ncdata` file in the new data location:

```
vim .ncdata
```

And write the following inside:

```
# Nextcloud data directory
```

3. change the ownership of the new data directory to `www-data`:

```
sudo chown -R www-data:www-data /path/to/your/data/directory
```

4. edit the `/var/www/nextcloud/config/config.php` file:

```
<?php
$CONFIG = array (
    'passwordsalt' => 'salt',
    'secret' => 'secret',
    'trusted_domains' =>
    array (
        0 => 'localhost',
        1 => 'example.com',
    ),
```

## data directory path

#####

#####

'datadirectory' => '/var/www/nextcloud-data',# <---- change that to '/path/to/your/data/directory'

'dbtype' => 'mysql',

'version' => '30.0.0.14',

'overwrite.cli.url' => 'http://example.com',

'overwriteprotocol' => 'https',

'dbname' => 'nextcloud',

'dbhost' => 'localhost',

'dbport' => '',

'dbtableprefix' => 'oc\_',

'mysql.utf8mb4' => true,

'dbuser' => 'nextcloud',

'dbpassword' => 'password',

'installed' => true,

'instanceid' => 'id-number',

'memcache.local' => '\\OC\\Memcache\\Redis',

'redis' =>

array (

    'host' => '/var/run/redis/redis.sock',

    'port' => 0,

    'timeout' => 0.0,

),

'filelocking.enabled' => true,

'memcache.locking' => '\\OC\\Memcache\\Redis',

'log\_type' => 'file',

## log file path

#####

#####

'logfile' => '/var/www/nextcloud-data/nextcloud.log',# <----- change that to

'/path/to/your/data/directory/nextcloud.log'

'loglevel' => 3,

'maintenance' => false,

'theme' => '',

'updater.secret' => 'secret',

);

# Disable the HTTP to HTTPS redirection

⚠ ONLY DO THAT IF YOU HAVE A VERY VALID REASON TO!

In my case, the NextCloud server is behind `HAProxy` who is set to offload the SSL encryption from the server and managing the certificates at the same time.

If NextCloud is served by `Apache2`, then this is the solution:

Edit the file `/etc/apache2/sites-available/nextcloud.conf`:

```
ServerName localhost

<VirtualHost *:80>
    UseCanonicalName Off
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/nextcloud/ # <----- Add this line
    # RewriteEngine On <----- Delete or comment this line like it is here
    # RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301] <----- Delete or comment this
line like it is here
</VirtualHost>

<VirtualHost *:443>
    SSLEngine on
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/nextcloud/

    <IfModule mod_headers.c>
        Header always set Strict-Transport-Security "max-age=15552000; includeSubDomains"
    </IfModule>
</VirtualHost>

<Directory /var/www/nextcloud/>
    Options +FollowSymLinks
    AllowOverride All
    Require all granted
```

</Directory>

Restart `apache2` service:

```
sudo systemctl restart apache2
```

# Tauri apps with blank screen on Linux

## Description

Multiple times, with my desktop computer running Archlinux with `bspwm`, I couldn't correctly open a Tauri app, it only showed a blank window with nothing inside. The app seemed to function well though, because I could see the cursor change in certain areas of the blank window as if there were buttons or links, or text inputs.

The issue seemed to be purely in the graphical rendering of the app.

## Simple workaround (launch from CLI)

After some research, I found a workaround, setting an environment variable to true:

```
WEBKIT_DISABLE_COMPOSITING_MODE=1
```

I'll give an example with `hoppscotch-desktop`, an app like `postman`, but open source and available to self host:

```
WEBKIT_DISABLE_COMPOSITING_MODE=1 hoppscotch-desktop
```

## Persistent workaround (launch from GUI)

It's not very practical to run a graphical application from the terminal sometimes though (except when you want to see all the logs in real time), so I looked for a persistent solution without having



to remember that variable's name every single time. That solution lies in the `.desktop` file.

In the `Exec=` section, you need to add, prior to the command to launch the app, `env` `WEBKIT_DISABLE_COMPOSITING_MODE=1`.

For our current example, `hoppscotch-desktop`, it is like this:

```
[Desktop Entry]
Categories=Development,API
Comment=Desktop App for hoppscotch.io
Exec=env WEBKIT_DISABLE_COMPOSITING_MODE=1 hoppscotch-desktop
Icon=hoppscotch-desktop
Name=Hoppscotch
Terminal=false
Type=Application
MimeType=x-scheme-handler/io.hoppscotch.desktop
```