Mozout

Generated on 15/11/2026 00:38

KNOWLEDGE BASE · PUBLIC DOCUMENT

Fixing “Maximum execution time exceeded”

PHP Errors & Logs

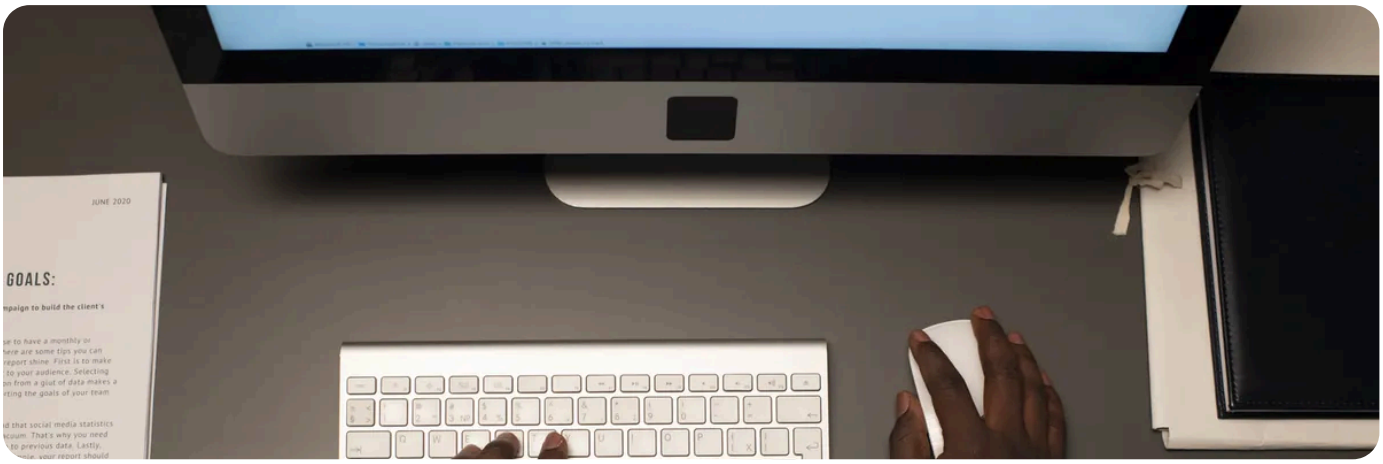
CODE	VERSION	COUNTRY · LANGUAGE
RLX-INT-KB-2478	v2026.10.10	International · English

MozOut International
RLX Holdings (Pty) Ltd. · Group RLX Holdings
Tax no. (SARS) 943 907 6234 · Registration no. (CIPC) 2025/667953/07

https://www.my.mozout.com/knowledgebase/2478

Fixing “Maximum execution time exceeded”

PHP Errors & Logs Created on **October 10, 2026** 3 min read



“Maximum execution time of N seconds exceeded” means the script took longer than PHP’s `max_execution_time` allows and was cut off halfway. Raising that value helps when the work is legitimately slow. When the slowness is a **defect** (a slow query, an outside service that does not answer), raising the value only hides the problem.

Why it takes so long

Cause	How to recognise it	What to do
A legitimately long task (import, export, building a report)	It always happens on the same button, and only there.	Raise the limit for that folder, or move the work to a scheduled task.
A slow database query	The whole site is slow, not one action.	See **slow queries**. The limit is not the problem.
A call to an outside service that does not answer	The delay varies, and it fails without warning.	Put a time limit on the call itself, and handle the failure.
An endless loop in the code	It fails at once, every time.	Fix the code. No limit is enough.

Raising the limit, if that is the case

1. In `Select PHP Version`, **Options** tab, find `max_execution_time` and choose a value above the current one. It is in seconds.
2. For a single application, put a line `max_execution_time = 120` (an example) in a `.user.ini` in its folder. See **where to set each value**.
3. If the work uploads a file, look at `max_input_time` too, which counts the time spent receiving the data.

4. Repeat the action. If it now goes through, stop. If it still fails with a much higher value, the cause is another one from the table above.

Finding where the time goes

1. **Repeat it with less data.** Import a smaller file, export a shorter period. If it goes through, the problem is volume, and the answer is to split the work into batches.

2. **Look at the line in the error.** The “Maximum execution time” message names the file and the line PHP was on when it was cut off. It is nearly always close to the slow spot.

3. **Switch plugins off one by one**, if it is WordPress, and repeat the action. A plugin that calls an outside service is a usual suspect.

PHP is not the only clock.
The web server and the account’s limits have their own. If you raised the value and the page still dies, the error you see is no longer PHP’s: it may be a 503, 504 or 508. See **website errors explained** and **the limits nobody advertises**.

Setting the time to zero is not a solution.
A script with no limit that gets stuck holds one of your account’s processes until somebody stops it, and spends the resources your site’s other requests need.

What is long runs outside the browser.
From the command line or from cron the web request disappears, and with it most of these cut-offs. Split the work into small batches that repeat: it is sturdier than one huge batch. See **running a PHP script from a cron job**.

NEED A HAND?

The page still gets cut off with a high limit? Send us the address, the time it happened and the message.

Open a support ticket →

SEE ALSO

- PHP limits: memory, time, upload size and the one that fails silently
- Running a PHP script from a cron job
- Website errors explained: 500, 403, 404, 508
- Where the PHP error log is

Mozout

MozOut International

RLX Holdings (Pty) Ltd. · Group RLX Holdings

TAX NO. (SARS)	943 907 6234
REGISTRATION NO. (CIPC)	2025/667953/07
REGISTERED OFFICE	1111 Olive Street Saint Louis 63101, Missouri, United States
CONTACT	info@mozout.com · +27 78 823 3325 · www.mozout.com
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-INT-KB-2478 · v2026.10.10 · Generated on 15/11/2026 00:38