Mozout

Generated on 21/10/2026 15:56

KNOWLEDGE BASE · PUBLIC DOCUMENT

Viewing and adjusting your PHP configuration: version per folder, extensions and phpinfo

PHP & Development

CODE	VERSION	COUNTRY · LANGUAGE
RLX-INT-KB-1128	v2026.09.28	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/1128

Viewing and adjusting your PHP configuration: version per folder, extensions and phpinfo

PHP & Development Created on **September 11, 2026** Updated on **September 28, 2026** 4 min read



Changing a whole site's PHP version is easy and is covered in **how to change your PHP version**. This article is the next step: **different versions in different folders**, extensions, and how to confirm what is actually running - because what the panel shows and what the page uses are not always the same thing.

The three tools, and what each is for

In cPanel	What for
MultiPHP Manager	Chooses the version **per domain**. Every domain and subdomain on the account can have its own.
PHP Selector	There you choose the version **and** switch extensions on or off, and adjust options. It is the complete tool.
MultiPHP INI Editor	Changes *php.ini* directives per domain, in basic mode or by editing the text.

Always start with the panel.
There is an `.htaccess` trick that also changes the version, and we explain it below - but it **does not carry your extensions or options** chosen in the panel. When the two cross, the site runs on a version without the pieces it needs, and the error that appears does not point here. Use the panel; use `.htaccess` only when there is no alternative.

A different version in one folder only

This is asked mainly in two cases: an old shop in a subfolder that will not yet run on PHP 8, or a new area that already needs a newer version than the rest of the site.

1. In **cPanel**, open **File Manager** and go into the folder. Turn on *Settings* → *Show Hidden Files*.

2. Open that folder's `.htaccess`, or create it. **Make a copy first**: duplicate the file and call it `.htaccess-copy`.

3. Add this at the **top** of the file, swapping the number for the version you want:

```
<FilesMatch "\.(php|phtml)$">
    SetHandler application/x-lsphp82
</FilesMatch>
```

4. Save and confirm with the test in the next section. If the site errors, delete the lines - everything returns to how it was.

The `82` is PHP 8.2; for 8.3 it would be `x-lsphp83`, and so on. Only use versions that appear in the panel: writing one that does not exist gives a 500 error across the whole folder - see **website errors: 500, 403, 404 and 508**.

> **If the folder is a subdomain, there is a better route.**
> If the folder corresponds to a subdomain, do not touch `.htaccess` - **MultiPHP Manager** treats subdomains as domains and lets you pick the version from the list, extensions and all. See **parked domains and subdomains**.

Confirming what is actually running

The panel says what *should* be there. To see what the page *uses*, ask it directly. Create a file in the folder you want to test:

```
<?php phpinfo();
```

1. Give it a name nobody would guess - `info-8f3a2.php`, not `info.php`.

2. Open it in the browser. The version is right at the top; further down are the loaded extensions and every directive in force.

3. **Delete the file immediately afterwards.**

Do not leave phpinfo online.
It shows system paths, loaded modules, exact versions and sometimes configuration values - in other words, a map for anyone wanting to attack the site. Automated scanners look for `info.php` and `phpinfo.php` by default, every day. Use it and delete it, always.

Running PHP inside .html files

This comes up when you inherit a site in `.html` and want to add one small dynamic part without changing every address. One line in `.htaccess` does it:

```
AddHandler application/x-httpd-lsphp .html .htm
```

It has a cost, and it is worth knowing.
From then on **every** `.html` file goes through PHP, even the ones with no code at all - which makes them slower and stops them being served from cache as static files. See **the site is slow: what actually makes a difference**. If it is only one page, it is nearly always better to rename it to `.php` and leave a redirect from the old address.

PHP extensions

This is the second reason for «I changed the version and the site broke»: each version has its own list of extensions, and the list does not follow by itself. A shop needing `intl` or `soap` stops working if the new version has them switched off.

1. In the PHP Selector, choose the version **first**.
2. Go through the extension list and enable the ones your site was using. If you do not know which, `phpinfo()` on the old version shows them.
3. Enable only what you need. Every loaded extension costs memory on every request.

If the site throws a critical error after the change, **common WordPress errors** helps read the message; and to go back, simply pick the previous version again on the same screen.

NEED A HAND?

Need a version or an extension you cannot find?

Talk to us →

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-1128 · v2026.09.28 · Generated on 21/10/2026 15:56