interweb

Generated on 19/10/2026 15:06

KNOWLEDGE BASE · PUBLIC DOCUMENT

Reducing memory use on shared hosting: plugins, PHP settings and what to switch off

cPanel & WHM

CODE	VERSION	COUNTRY · LANGUAGE
RLX-MZ-KB-2216	v2026.10.19	Mozambique · English

Interweb Mozambique
RLX Holdings, S.A. · Group RLX Holdings
Tax ID (NUIT) 401 191 607 · Legal entity no. (NUEL) 105 045 974 · Economic activity code (CAE) 62022

https://www.meu.interweb.co.mz/knowledgebase/2216

The short answer: the account's memory is the **sum** of everything running at the same moment: every PHP request, every cron job, every script, every app. So you bring it down in two ways: make **each process use less** (fewer heavy plugins, fewer loaded extensions, big jobs split into parts) and make **fewer processes run at once** (caching, fewer bots, jobs spaced out). Raising PHP's memory_limit gives the account no new memory: it only lets each request spend more of what is already there, as **PHP limits** explains.

What uses memory, and what to do

Cause	What you see	What to do
Heavy plugins: page builders, security plugins that scan the site, backups, statistics inside WordPress	High memory in the WordPress dashboard, and spikes when scans or backups run	Switch off the ones you do not use; replace heavy ones with lighter ones; move scans to a quiet hour
Many PHP requests at the same time	Memory follows the visits, often together with entry processes	Page caching: a cached page does not start PHP
Imports, exports and backups done in one go	They stop halfway, with **Allowed memory size** in the log, or memory faults at the hour they run	Split them up; run them from the command line, as in **importing a large database over SSH**
Huge photos processed on the server	Image uploads that fail, and a memory spike with each upload	Shrink images on your computer before uploading them
PHP extensions switched on that the site does not use	Nothing visible: each PHP process is a little larger, multiplied by the number of processes	Switch off only the ones you are sure you do not use, in Select PHP Version
A Node.js or Python app always running	Memory taken even at night, with no visitors	Check it is still needed; if it is, and it is big, it belongs on a VPS

Bringing memory down, step by step

1. **See when it fails.** In cPanel, **Resource Usage**: on the memory graph, the **faults** mark the moments the account hit the limit. At those moments requests fail, with a blank page or an error. Match the times with your jobs and backups.

2. **Take a backup before touching plugins.** Then switch off the ones you do not use and, to find the heavy one, switch off half, measure, and repeat with the half that is left. If the dashboard will not open: **disabling plugins when you cannot get in**. A diagnostic plugin such as Query Monitor shows how much memory each page used.

3. **Switch page caching on.** Fewer requests starting PHP means less memory taken at the same time. See **LiteSpeed Cache for WordPress**.

4. **Review your PHP settings.** In **Select PHP Version**, Options tab, keep memory_limit at a sensible value, not «huge just in case»: a high value only lets one badly written request eat the memory of the others. On the extensions tab, switch off the ones you are sure the site does not use, one at a time, testing the site after each: **switching a PHP extension on or off**.

5. **Move big jobs out of busy hours.** Backups, imports, security scans and reports go to a quiet hour, and the big ones are done in parts or from the command line.

6. **In WordPress, cut the background calls.** An open editor, cart fragments and statistics inside WordPress keep processes alive without anyone noticing: **WordPress Heartbeat**.

7	**Measure again** after a few ordinary days, at the same hours.

Optimise or upgrade?

Situation	Optimise or upgrade?
One plugin alone uses a large share of the memory	**Optimise.** Replace or configure the plugin. An upgrade only postpones the problem.
Memory rises with visits and there is no cache	**Optimise.** Switch caching on first.
The faults only appear during a backup or an import	**Optimise.** Split the job, move it to another hour.
You see «Allowed memory size exhausted» on a request	That is PHP's memory_limit, not the plan: **fixing Allowed memory size exhausted**.
Site already optimised, with caching, and memory runs into the limit at your busiest hours	**Upgrade.**
An app that needs a lot of memory all the time (queues, automations, AI)	**VPS or Cloud**, with memory of your own: **shared hosting vs VPS**.

Bigger plans

Each plan up gives the account more memory: from **1 GB on Pro to 8 GB on Business Ultra** (table below). It is the memory for everything running in the account at the same moment; yours is the limit line on the memory graph in Resource Usage. Note that an upgrade raises the **account's** memory, not the memory_limit of each request, which you change in Select PHP Version. You upgrade in Meu Interweb, pay only the difference until renewal, and nothing moves: **how to upgrade your plan**. Which plan comes after yours: **which hosting plan comes next**.

Plan	Memory (RAM)
Pro	1 GB
Plus	2 GB
Ultra	3 GB
Business Pro	4 GB
Business Plus	6 GB
Business Ultra	8 GB

Our warnings: when the account reaches **75%**, **85%** and **95%** of the limit of a plan resource, you get an e-mail with every resource on the account, each with its limit and the percentage you use, and the upgrade plans (between 75 and 85%, every 15 days; between 85 and 95%, every 7 days; from 95%, every 3 days). If several resources are close to the limit, they all come in the same e-mail. The limit is 100%. Above it the account **may be suspended**, and we always warn you the day before. To bring it back, upgrade (the account reactivates by itself straight after) or ask for **24 hours to sort it out** from the link in the e-mail: the account stays active for 24 hours and, if it is still over the limit at the end, it is suspended again straight away (you can ask for the 24 hours once every 7 days). If usage drops back below the limit, the account becomes active again by itself. What each warning means is in **usage warnings at 75%, 85% and 95%**.

Switching off a PHP extension the site uses breaks the site straight away, nearly always with error 500. If that happens, go back to the same screen and switch it on again. If in doubt, leave it: the ones the site uses (the database one, cURL, the image ones, mbstring, for example) stay on.

Deleting a plugin is not the same as switching it off. Switch it off first, use the site for a few days, and only then delete it, with a backup taken: **making and keeping your own backup**.

Removed the heavy plugins and memory is still at the limit? Tell us the domain and the times of the faults.

Open a support ticket

SEE ALSO

PHP limits: memory, time, upload size and the one that fails silently

Giving WordPress more memory

CPU, RAM and disk I/O in plain words: what each one limits on your site

CPU usage at your plan limit: caching, bots and cron jobs that bring it down

Disabling plugins, including when you cannot get in

How to upgrade or downgrade your plan

interweb

Online version of this document

Interweb is a leading internet services company, with a mission to help businesses to be born and prosper through technology.

2020	37,000+	130+
FOUNDED	CLIENTS	SIGN-UPS A DAY

Interweb Mozambique

RLX Holdings, S.A. · Group RLX Holdings

TAX ID (NUIT)	401 191 607
LEGAL ENTITY NO. (NUEL)	105 045 974
ECONOMIC ACTIVITY CODE (CAE)	62022
REGISTERED OFFICE	Bairro 7 de Setembro Vilankulo 1304 · CEP 0411-02, Inhambane, Mozambique
CONTACT	info@interweb.co.mz · +258 84 635 1111 · +258 86 635 1111 · www.interweb.co.mz
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

RLX-MZ-KB-2216 · v2026.10.19 · Generated on 19/10/2026 15:06