Mozout

Gerado a 15/11/2026 02:05

BASE DE CONHECIMENTO · DOCUMENTO PÚBLICO

O admin-ajax a consumir tudo: como descobrir e reduzir

Site lento e carga do servidor

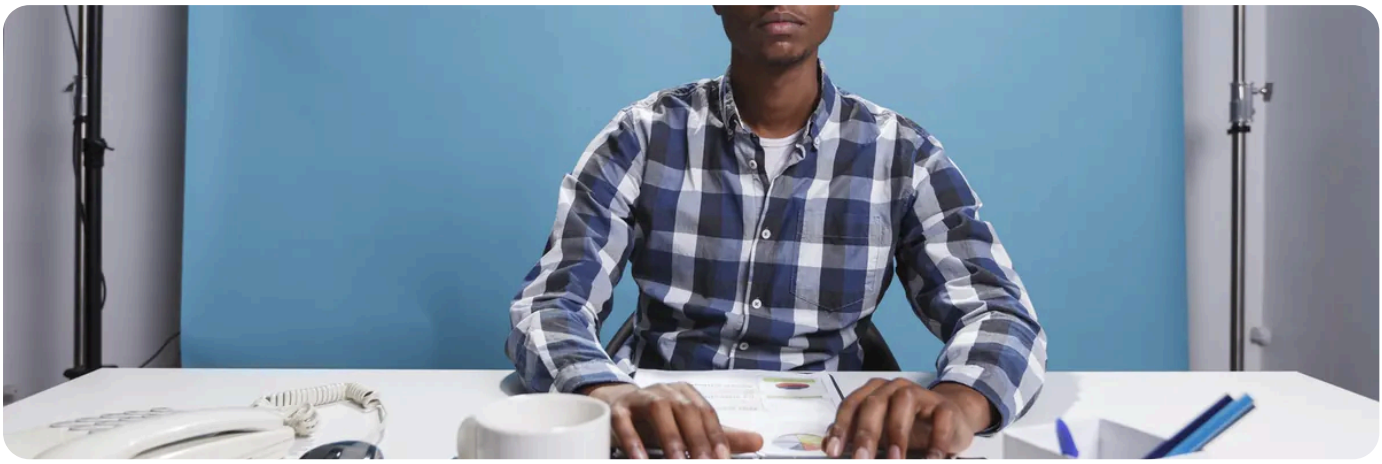
CÓDIGO	VERSÃO	PAÍS · IDIOMA
RLX-AO-KB-1988	v2026.09.27	Angola · Português

MozOut Angola
RLX Angola, Lda. · Grupo RLX Holdings
NIF 500 260 5724 · Matrícula 26464-25/250724

https://www.my.mozout.com/knowledgebase/1988

O admin-ajax a consumir tudo: como descobrir e reduzir

Site lento e carga do servidor · Criado em **27 de setembro de 2026** · 4 min de leitura



O `admin-ajax.php` é a porta por onde os *plugins* falam com o servidor em segundo plano, sem recarregar a página. Serve o painel e serve também o lado público do site: carrinhos, filtros, formulários, pesquisa ao vivo. Não é um defeito, e não se desliga.

O problema é outro: **cada chamada arranca o WordPress inteiro**, nunca é guardada em cache, e ocupa uma vaga de pedido em simultâneo, exactamente como um visitante. Num site onde alguma coisa chama o `admin-ajax.php` de cinco em cinco segundos, isso são milhares de arranques por dia a competir com quem está a tentar ver o site.

Como saber se é isto

Três maneiras, da mais rápida para a mais completa:

1. **No navegador.** Abra o site, carregue na tecla F12, vá ao separador **Rede** (Network) e escreva `admin-ajax` na caixa de filtro. Deixe a página aberta um minuto sem tocar em nada. Se a lista continua a crescer, encontrou o problema.

2. **No painel.** Repita o mesmo com o `/wp-admin` aberto, sobretudo no editor de páginas. Aí é normal haver alguns pedidos; o que não é normal é serem constantes.

3. **No cPanel.** Na secção **Métricas**, o registo de acessos em bruto mostra os pedidos todos. Conte quantas linhas têm `admin-ajax.php` num dia e compare com o número de páginas servidas. Se são da mesma ordem de grandeza, metade do seu servidor está a trabalhar para isto.

Cruze com o consumo.

Se o **Resource Usage** mostra picos de pedidos em simultâneo às mesmas horas em que o `admin-ajax.php` dispara, está provado: **acompanhar o desempenho do site no cPanel** e **os limites que não vêm no anúncio**.

Os culpados do costume

Quem chama	O que se vê e o que fazer
O batimento do editor	Enquanto uma página está aberta no editor, o WordPress fala com o servidor de poucos em poucos segundos para gravar sozinho e avisar que ela está a ser editada. Um separador esquecido aberto uma tarde inteira são centenas de pedidos. É a causa número um em sites pequenos.
Fragmentos de carrinho	Numa loja, o total do carrinho no cabeçalho é actualizado por uma chamada em **todas** as páginas, mesmo nas que não têm carrinho nenhum. É o caso mais pesado que vemos.
Estatísticas dentro do WordPress	*Plugins* que registam cada visita na sua base de dados fazem uma chamada por visita, e escrevem uma linha de cada vez. Custam duas vezes: ao servidor e à base.
Pesquisa ao vivo e filtros	Uma chamada por cada letra escrita na caixa de pesquisa. Parece elegante e é caro.
Carrosséis e contadores	Contadores de visitas, «X pessoas estão a ver isto», sondagens, avisos de stock.

O que fazer, por ordem

1. **Identifique quem chama.** No separador **Rede**, carregue num dos pedidos e veja o campo `action` no corpo dele. O nome dessa acção costuma ter o nome do *plugin* lá dentro, e aí já sabe com quem falar.

2. **Trave o batimento.** Há *plugins* feitos só para isso: deixam-no escolher de quanto em quanto tempo é que o editor fala com o servidor, e desligam-no nas páginas onde não faz falta. É a mudança mais rápida e a de menor risco.

3. **Desligue os fragmentos de carrinho** nas páginas que não são de loja. A maioria dos temas de loja tem essa opção; se não tiver, há *plugins* que a acrescentam.

4. **Tire as estatísticas de dentro do WordPress.** Um serviço de estatísticas externo não gasta nada do seu servidor. Se prefere não usar serviços de fora, o cPanel já conta as visitas por si: **estatísticas de visitas**.

5. **Reveja as funcionalidades ao vivo.** A pesquisa instantânea, o contador de pessoas online e o carrossel automático raramente pagam o que custam.

Não bloqueie o `admin-ajax.php`.
É a primeira coisa que aparece nos fóruns e parte o site: formulários que deixam de enviar, carrinhos que deixam de somar, filtros que não filtram, e nada disto dá erro visível. O mesmo se aplica à protecção da pasta `wp-admin` por senha, que o apanha pelo caminho: **proteger a entrada do WordPress**.

O primo que se confunde com este: o `wp-cron.php`

Se no registo de acessos vê muito `wp-cron.php` em vez de `admin-ajax.php`, o problema é parecido mas a cura é outra: o agendador interno do WordPress corre à boleia das visitas. Trata-se em **trocar o WP-Cron pelo cron do servidor**.

E se o que vê é muito `/wp-login.php` ou `xmlrpc.php`, isso não é o seu site a trabalhar: são tentativas de entrada de fora. Essas tratam-se em **proteger a entrada do WordPress**.

O que isto resolve, e o que não resolve

Reduzir as chamadas em segundo plano não torna as páginas mais rápidas a construir: torna o servidor mais livre para as construir. O ganho vê-se nas horas de ponta, nos erros que desapareciam sozinhos e no painel que deixa de arrastar. Para o resto, a lista por ordem de retorno está em **o site está lento: o que faz mesmo diferença**, e a cache das páginas públicas em **instalar e afinar uma cache de páginas**.

PRECISA DE AJUDA?

Quer que vejamos o registo de acessos consigo? Diga-nos o domínio e a hora do pico.

Abrir um pedido →

VEJA TAMBÉM

- **Hospedagem WordPress** →
- **Planos de hospedagem e os limites de cada um** →
- **Política de Suporte** →

Mozout

MozOut Angola

RLX Angola, Lda. · Grupo RLX Holdings

NIF	500 260 5724
MATRÍCULA	26464-25/250724
SEDE	Rua Direita do Patriota, Bairro Dangereux Belas, Luanda, Angola
CONTACTOS	info@mozout.com · +244 942 011 191 · www.mozout.com
HORÁRIO	2ª. à 6ª. Feira - 08 à 16 Horas Sábado & Domingo - Fechado

Versão online deste documento

RLX-AO-KB-1988 · v2026.09.27 · Gerado a 15/11/2026 02:05