interweb

Generated on 15/11/2026 00:13

KNOWLEDGE BASE · PUBLIC DOCUMENT

Receiving a webhook in PHP and checking its signature

Webhooks, APIs & Tokens

CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-1064	v2026.10.18	Angola · English

Interweb Angola
RLX Angola, Lda. · Group RLX Holdings
Tax ID (NIF) 500 260 5724 · Commercial registration no. 26464-25/250724

https://www.meu.interweb.ao/knowledgebase/1064

Receiving a webhook in PHP and checking its signature

Webhooks, APIs & Tokens · Created on **October 10, 2026** · Updated on **October 18, 2026** · 3 min read



A webhook is a request another service makes to you when something happens. In a PHP file on your site you receive it in four steps: **read the raw body, check the signature, answer quickly and handle it afterwards**. The signature step is what stops anyone who discovers the address from sending you fake data.

The skeleton, in four steps

1. **Read the raw body** with `file_get_contents('php://input')`. Do not use `json_decode` yet: the signature is calculated over the text exactly as it arrived.

2. **Calculate the expected signature** with `hash_hmac('sha256', $body, $secret)`, where the secret is the one you agreed with the service. Read the signature that was sent from the header the service names, for example `$_SERVER['HTTP_X_SIGNATURE']`. The header name changes from one service to another.

3. **Compare with** `hash_equals`, not with `==`. If they differ, answer 401 and stop.

4. **Answer 200 quickly.** Many services give up and resend if there is no reply within a few seconds. Store the event in a file or a table and let a cron job handle it. See **running a PHP script from a cron job**.

Proving it works, without waiting for the service

1. **Calculate a test signature** in the terminal: `echo -n '{"test":1}' | openssl dgst -sha256 -hmac "SECRET"`.

2. **Make the request with it**, using `curl -i -X POST`, the signature header and the same body. See **what a webhook is and how to test one with curl**.

3 **Repeat with a slightly different body.** You must get 401. If you get 200, the check is not working.

When the webhook does not arrive or is refused

Symptom	Likely cause
The signature never matches	A different secret, or the body was changed before you calculated (parsed and rewritten).
The service uses a different format	Some add a prefix to the value, such as the algorithm name. Follow the service's documentation.
The request arrives as GET, with no body	A redirect from http to https, or from www to non-www, turns the POST into a GET. Use the final address.
403 before it reaches PHP	A security rule or an IP block. See **website errors explained**.
The same event twice	Services resend when they do not get a 200. Store the event identifier and ignore repeats.

A secret in the code is a secret in plain sight.
Keep it in a file outside `public_html`, and do not put it in a repository. See **keeping passwords out of your PHP code**. And never trust the data of an unsigned webhook as if it were your own.

Log each request received
(time, result of the check, size of the body), without storing sensitive data. When the service swears it sent and you see nothing, the log tells you which side the problem is on.

RECOMMENDED
Want a plan with PHP to receive your webhooks? See the hosting plans.
See the plans →

Where to see it in your account

The PHP versions you can use and the time a script may run depend on your account, so read them in cPanel. Under **Select PHP Version** (or **MultiPHP Manager**) you see the versions available, and the **Options** tab of Select PHP Version shows values such as `max_execution_time` and `memory_limit`, which you can also change there. A webhook handler should answer in a few seconds anyway, as the article says: store the event and let a cron job do the slow work, so the execution time hardly matters. If the receiver is refused or stopped at busy hours, **Resource Usage** shows whether the account hit a limit at that moment.

SEE ALSO

- **What a webhook is, and how to test one with curl**
- **Receiving messages: Evolution API webhooks**
- **Webhooks from Meta: verifying and receiving**

interweb

Interweb Angola

RLX Angola, Lda. · Group RLX Holdings

TAX ID (NIF)	500 260 5724
COMMERCIAL REGISTRATION NO.	26464-25/250724
REGISTERED OFFICE	Rua Direita do Patriota, Bairro Dangereux Belas, Luanda, Angola
CONTACT	info@interweb.ao · +244 942 011 191 · www.interweb.ao
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-AO-KB-1064 · v2026.10.18 · Generated on 15/11/2026 00:13