interweb

Generated on 21/10/2026 17:25

KNOWLEDGE BASE · PUBLIC DOCUMENT

cURL error 60 in PHP: SSL certificate problem, and why not to switch verification off

PHP & Development

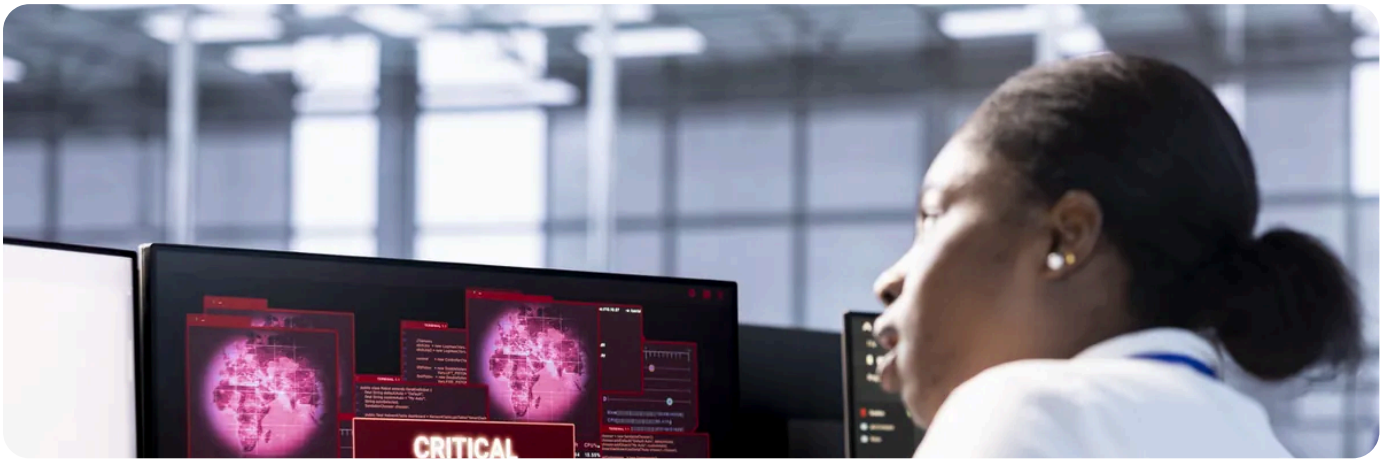
CODE	VERSION	COUNTRY · LANGUAGE
RLX-MZ-KB-1444	v2026.10.10	Mozambique · English

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/1444

cURL error 60 in PHP: SSL certificate problem, and why not to switch verification off

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



"cURL error 60: SSL certificate problem" means PHP connected to a service over HTTPS and **did not trust the certificate** that service presented. PHP is doing what it should: refusing to talk to a server whose identity it cannot prove. The fix is almost never to switch verification off. It is to find **which side holds the fault**: the other service's certificate, or the trusted certificates on your side.

The messages and what they point to

The message	Likely cause	Whose it is
certificate has expired	The destination service's certificate has expired.	The destination's. Tell them.
unable to get local issuer certificate	An intermediate certificate is missing on the destination server, or your side has an old or incomplete list of trusted certificates.	Either: see the steps below.
self signed certificate	The destination uses a certificate it made itself, not one issued by a recognised authority.	The destination's. Common in test environments.
no alternative certificate subject name matches	The certificate is for a different name than the address called.	The code's (wrong address) or the destination's.

Finding which side holds the fault

1. **Open the service address in a browser.** If the browser warns too, the problem is the destination's certificate. A browser sometimes "patches" a missing intermediate certificate that PHP does not: if the browser is quiet and PHP complains, suspect that.

2. **See the details with cURL**, if your account has a Terminal: `curl -vI https://service.tld/`. The output shows the chain and the reason for the refusal.

3. **Use a public SSL checker** on the destination address. It shows whether the certificate chain is complete and when it expires.

4. **Check your own side.** Here PHP's stock configuration does not fix a certificate file (the `curl.cainfo` and `openssl.cafile` options are empty), so PHP uses the system's. If the error is from one application only, it may ship its own old certificate file: update the application, or the plugin making the call. See **updating themes and plugins without breaking anything**.

The fix, by cause

Cause	What to do
Destination certificate expired or badly installed	Tell whoever runs the service. There is no honest repair on your side.
Application with old certificates	Update the application or plugin. See also **which PHP version to use**.
A private (test) certificate you do want to accept	Tell PHP which certificate to trust, with `CURLOPT_CAINFO` pointing at the authority's file. Do not switch verification off.
On a VPS of your own	Update the system's certificate package (`ca-certificates`). The commands are yours; support does not handle them. See **how far our support goes**.

Do not set `CURLOPT_SSL_VERIFYPEER` to `false` and leave it.
That makes PHP accept any certificate, including that of someone who slipped into the middle of the connection. It is fine, at most, for a minute, in a test, to confirm the diagnosis, and it comes out straight away. If a plugin or a tutorial tells you to leave it that way, be suspicious.

Did it appear out of nowhere?
The commonest cause is a certificate that expired or was renewed with a new chain. Check the service's certificate expiry date first. For your own site, see **what an SSL certificate is**.

NEED A HAND?

Want us to look at the server side? Send us the address of the service called and the full error message.

Open a support ticket →

SEE ALSO

- cURL error 28, 7 and 6 in PHP →
- Where the PHP error log is →
- What is an SSL certificate and why does it matter? →
- Updating themes and plugins without breaking anything →

interweb

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

Online version of this document

RLX-MZ-KB-1444 · v2026.10.10 · Generated on 21/10/2026 17:25