

CVE-2026-32144

OCSP designated-responder authorization bypass via missing signature verification

[« Back to all CVEs](#)[See on OSV.dev »](#)

Weakness Type (CWE)

[CWE-295 — CWE-295 Improper Certificate Validation](#)

CAPEC

[CAPEC-459 — CAPEC-459 Creating a Rogue Certification Authority Certificate](#)

CVSS 4.0 Score

7.6

HIGH

[CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:L/SI:L/SA:N](#)

Vulnerability description

Improper Certificate Validation vulnerability in Erlang OTP `public_key` (`pubkey_ocsp` module) allows OCSP designated-responder authorization bypass via missing signature verification.

The OCSP response validation in `public_key:pkix_ocsp_validate/5` does not verify that a CA-designated responder certificate was cryptographically signed by the issuing CA. Instead, it only checks that the responder certificate's issuer name matches the CA's subject name and that the certificate has the OCSPSigning extended key usage. An attacker who can intercept or control OCSP responses can create a self-signed certificate with a matching issuer name and the OCSPSigning EKU, and use it to forge OCSP responses that mark revoked certificates as valid.

This affects SSL/TLS clients using OCSP stapling, which may accept connections to servers with revoked certificates, potentially transmitting sensitive data to compromised servers. Applications using the `public_key:pkix_ocsp_validate/5` API directly are also affected, with impact depending on usage context.

This vulnerability is associated with program files `lib/public_key/src/pubkey_ocsp.erl` and program routines `pubkey_ocsp:is_authorized_responder/3`.

This issue affects OTP from OTP 27.0 until OTP 28.4.2 and 27.3.4.10 corresponding to `public_key` from 1.16 until 1.20.3 and 1.17.1.2, and `ssl` from 11.2 until 11.5.4 and 11.2.12.7.

Affected

pkg:otp/public_key

Module	Source File	Routine	
pubkey_ocsp	<code>src/pubkey_ocsp.erl</code>	pubkey_ocsp:is_authorized_responder/3	
Status	Type	Version	Changes / Fixed in
Affected	otp ⓘ	1.16	<ul style="list-style-type: none"> unaffected at 1.20.3 unaffected at 1.17.1.2

pkg:otp/ssl

Module	Source File
ssl_stapling	<code>src/ssl_stapling.erl</code>

Status	Type	Version	Changes / Fixed in
Affected	otp ^①	11.2	<ul style="list-style-type: none"> unaffected at 11.5.4 unaffected at 11.2.12.7

[pkg:github/erlang/otp](https://pkg.github.com/erlang/otp)

Module	Source File	Routine
pubkey_ocsp	lib/public_key/src/pubkey_ocsp.erl	pubkey_ocsp:is_authorized_responder/3

Status	Type	Version	Changes / Fixed in
Affected	otp ^①	27.0	<ul style="list-style-type: none"> unaffected at 28.4.2 unaffected at 27.3.4.10
Affected	git ^①	601a012837ea	<ul style="list-style-type: none"> unaffected at ac7ff528be85 unaffected at 49033a6d93a5

Workarounds

For SSL users:

- Do not enable OCSP validation setting (current default is {stapling, no_staple})
- Use CRL-based revocation checking by setting the {crl_check, true} SSL option instead

For applications using public_key:pkix_ocsp_validate/5 directly:

- Pass {is_trusted_responder_fun, Fun} option with a function that validates trusted responder certificates
- Restrict OCSP responder access to trusted endpoints via network controls (only applicable if you control the OCSP infrastructure)

References

- <https://github.com/erlang/otp/security/advisories/GHSA-gxrm-pf64-99xm> vendor-advisory related
- <https://osv.dev/vulnerability/EEF-CVE-2026-32144> related
- <https://github.com/erlang/otp/commit/ac7ff528be857c5d35eb29c7f24106e3a16d4891> patch
- <https://github.com/erlang/otp/commit/49033a6d93a5be0ee0dce04e1fb8b4ae7de1e0c0> patch

Credits

- Reporter:** Igor Morgenstern / Aisle Research
- Remediation developer:** Jakub Witczak
- Remediation reviewer:** Ingela Anderton Andin

CVE record as JSON: [GET /cves/CVE-2026-32144.json](#)

OSV record as JSON: [GET /osv/EEF-CVE-2026-32144.json](#)



ERLANG ECOSYSTEM
FOUNDATION

Supporting the BEAM community

ABOUT THE EEF

[Membership details](#)

[Join us!](#)

[Sponsors](#)

[Working Groups](#)

[Stipends](#)

[Bylaws](#)

[FAQ](#)

STAY UP-TO-DATE

[News](#)

[Events](#)

GET IN TOUCH

[Contact Us](#)