

SECURITY ADVISORY

Critical PHP Runtime Remote Code Execution Flaws

Two critical PHP vulnerabilities can enable unauthenticated server compromise and exposure of sensitive data.

SEVERITY	PUBLISHED	CATEGORY	STATUS
CRITICAL	23 OCT 2026	PHP / RCE	PATCH AVAILABLE
URGENT ACTION Patch affected PHP installations immediately to address two critical unauthenticated remote code execution flaws. CERT-In reports exploitation in the wild. Temporary mitigations do not provide complete protection. [1, 2]			

01 INCIDENT SUMMARY

On 21 October 2026, the PHP project released emergency security fixes for CVE-2026-109618 and CVE-2026-109619. Each vulnerability is rated CRITICAL (CVSS v4.0: 9.2) and may allow unauthenticated attackers to execute arbitrary code on vulnerable PHP web servers. The issues arise from memory-handling flaws in the PHP runtime, not from a specific application plug-in. [1, 3]

The vulnerabilities affect PHP-FPM, Apache mod_php, php-cgi, FrankenPHP and LiteSpeed deployments running affected PHP versions. The PHP team identifies Apache mod_php as particularly exposed. Public proof-of-concept research is available, and CERT-In reported large-scale exploitation on 22 October 2026. [1-3]

02	9.2	21 OCT
Critical vulnerabilities	CVSS v4.0 (each)	Security fixes released

02 POTENTIAL IMPACT

Server compromise — An attacker may execute commands with the permissions of the PHP worker, potentially enabling persistence, website defacement or further intrusion.

Data exposure — Successful attacks could expose application data, credentials or sensitive information held in server memory.

Widespread exposure — Public-facing websites, APIs, content management systems, containers, hosting environments and products embedding PHP may be at risk. [3-5]

03 PRIORITY ACTIONS FOR CUSTOMERS

- 1. Identify exposed PHP systems.** Inventory internet-facing sites, APIs, web hosts, containers and third-party services. Check the actual PHP version serving each application, not only the command-line version.
- 2. Patch and confirm.** Update to PHP 8.5.12, 8.4.27, 8.3.36 or 8.2.35 (or a vendor-confirmed patched backport). Restart/redeploy affected workers and verify the live runtime and application functionality.
- 3. Investigate for compromise.** Review web/PHP logs, abnormal requests, unexpected child processes, suspicious file changes and outbound connections. Preserve evidence and activate incident response if indicators are found.
- 4. Contain systems that cannot be patched.** Restrict internet access or temporarily take vulnerable services offline until fixed. WAF rules and vendor mitigations may reduce exposure but are not a complete fix. [1, 3, 4]

Scope note: This is a vulnerability advisory; it does not establish compromise of any recipient. For managed hosting or bundled software, obtain vendor confirmation of the deployed security fix.

TECHNICAL & RESPONSE DETAILS

Guidance for infrastructure, development, operations and security teams

04 TECHNICAL INFORMATION

CVE / severity	CVE-2026-109618 and CVE-2026-109619; both rated CRITICAL, CVSS v4.0 9.2.
CVE-2026-109618	Out-of-memory handling can leave request data partially initialized; cleanup may corrupt memory and enable remote code execution.
CVE-2026-109619	Stale request-owned state may be accessed after its backing memory is freed (use-after-free), potentially enabling code execution.
Affected deployments	PHP-FPM, Apache mod_php, php-cgi, FrankenPHP and LiteSpeed; PHP considers mod_php the highest-risk configuration.
Exploitation status	Public exploit research exists; CERT-In reported large-scale exploitation in the wild on 22 October 2026. [2, 3]
Detection focus	Review abnormal PHP errors, unexpected shell processes, unfamiliar PHP files, web shells, unusual network traffic and data-access activity.

05 PATCHED VERSIONS & EXPOSURE

PHP 8.5 Upgrade to 8.5.12 or later.

PHP 8.4 Upgrade to 8.4.27 or later.

PHP 8.3 Upgrade to 8.3.36 or later.

PHP 8.2 Upgrade to 8.2.35 or later; this branch reaches end of life on 31 December 2026.

PHP 8.1 / older Upstream support has ended; upgrade or obtain a verified vendor-supported security backport.

Managed services Ask providers to confirm whether their running PHP build includes both fixes, including bundled appliances and hosting plans.

**Vendor-backported packages may retain an older PHP version number. Verify the security patch through the supplier advisory. [1, 3]*

06 VALIDATION CHECKLIST

Discover all runtimes: Confirm PHP versions and SAPIs in web workers, PHP-FPM pools, container images, managed platforms and embedded products.

Verify deployment: Confirm patched builds, restart PHP workers and services as required, and test authentication, uploads, payments and other critical application paths.

Hunt for malicious activity: Correlate recent web and PHP error logs with endpoint telemetry; prioritize unexpected shell execution, altered web content and anomalous outbound sessions.

Escalate suspected breaches: If suspicious activity is confirmed, isolate affected systems, retain evidence, assess data exposure and rotate affected secrets as appropriate.

07 PRIMARY SOURCES

[1] [PHP Security Team • 21 Oct 2026](#)

[2] [CERT-In Security Advisory • 22 Oct 2026](#)

[3] [CERT-EU Advisory 2026-018 • 21 Oct 2026](#)

[4] [UK NCSC Security Update • 21 Oct 2026](#)

[5] [Google Cloud Technical Guidance • 21–22 Oct 2026](#)

[6] [NCSC-NL PHP Dependency Tracker](#)

Advisory basis. Based on publicly available information as of 23 October 2026. This independent 443 Labs advisory is for customer awareness and does not indicate that any recipient is compromised. Prioritize internet-facing PHP systems, verify remediation with providers and investigate suspected exploitation promptly.