

THREAT ADVISORY | OCTOBER 21, 2026 · VERSION 1.4

PHP2Shell: What the Critical PHP Vulnerabilities Mean for Your Organization

On October 21, 2026, the PHP project released patches for two critical vulnerabilities that allow adversaries to execute code on web servers without authentication. Every PHP installation that serves web requests is affected, from the WordPress blog to the firewall interface. This advisory tells you what to do and what to look for in your logs.

Audience

Operators of web applications and appliances in Germany, Austria and Switzerland, CISOs, heads of IT, SOC, hosting managers

Regulation

NIS2 / BSIG Section 30 · DORA · ISO 27001:2022

Classification

TLP:CLEAR, sharing permitted

As of

October 23, 2026, 9:00 a.m., version 1.4

Sources

PHP project, BSI, ENISA, CERT-EU, CIRCL, CERT-FR, NCSC UK, NCSC-NL, BACS, CERT.at, CISA, FBI, Amazon, Wiz, VulnCheck, Sansec, ELLIO, Dataminr, CrowdStrike, Datadog, Cloudflare, Imperva, Unit 42, Palo Alto, Fortinet, Cisco, Netgate, OPNsense, QNAP, Synology, Debian, Ubuntu, Red Hat

Changes

Version 1.4 of October 23, 2026, 9:00 a.m. New: NCSC-NL is investigating first signs of successful exploitation, exploit status from VulnCheck, vendor overview (PAN-OS, Fortinet, Cisco, pfSense, OPNsense, QNAP, Synology), Debian bookworm and Red Hat corrected, regression in 8.4.27 and 8.5.12. Earlier versions: 1.3 (Oct. 22, 5:30 p.m.), 1.2 (Oct. 22, 10:00 a.m.), 1.1 (Oct. 21, 10:30 p.m.), first version Oct. 21, 6:00 p.m.

SIX IMMEDIATE ACTIONS

1

Find PHP, even where it is not labeled

Web servers, container images, hosting packages, admin interfaces, appliances such as pfSense, OPNsense, PAN-OS or NAS devices from QNAP and Synology. Internet-facing systems first.

[CIS Control 1, 2](#)

2

Patch and restart every worker

PHP 8.5.12, 8.4.27, 8.3.36 or 8.2.35, or the backport package for distributions. Rebuild containers. Check what is actually running, not just `php -v`.

[CIS Control 7](#)

3

Cannot patch? Take it offline

Temporarily disconnect exposed PHP systems from the internet or put them behind a VPN until the patch is in place. A WAF alone is not enough.

[CIS Control 12](#)

4

The PHP project's interim measures

`session.upload_progress.enabled=0`, `memory_limit=-1` set administratively, `fatal_error_backtraces=0` for PHP 8.5, limit multipart at the proxy. Each closes only individual paths.

[CIS Control 4](#)

5

Review logs before and after the patch

Multipart anomalies, `PHP_SESSION_UPLOAD_PROGRESS`, crashing workers, memory exhaustion, child processes of the web worker, new files in the web root.

[CIS Control 8](#)

6

Replace old PHP branches

PHP 8.1 and older are unsupported, and 8.2 ends on December 31, 2026. The source backports for 7.4, 8.0 and 8.1 are a stopgap.

[CIS Control 2, 7](#)

Management Summary

What happened?

On October 21, the PHP project patched two bugs in the PHP core (CVE-2026-109618 and CVE-2026-109619, CVSS 4.0 of 9.2 each) that allow code execution with the privileges of the web worker without authentication. The BSI rates the situation as Orange, "proof of concept public". Amazon saw an attempt as early as October 17, and since the patches, several actors have been attacking in an automated way. As of October 23, nobody has reported a confirmed takeover: NCSC-NL has been investigating first signs of successful exploitation since October 22, the BSI knows of no compromise in regulated industry or the federal administration, and CISA's KEV catalog still does not list the vulnerabilities.

What does this mean?

The bugs are in the runtime, not in an application: an update of WordPress, TYPO3 or Nextcloud does not help; the PHP underneath must be patched. Every reachable PHP endpoint is affected, including login routes and health checks, plus firewalls and appliances (pfSense, OPNsense, PAN-OS) and NAS systems from QNAP and Synology, where nobody thinks of PHP running inside. Since October 22, a picture has been emerging: attempts are broad and automated, while successful attacks are difficult and so far unproven.

What do we need to do?

Immediately, the six actions above. Today: a patch window for all internet-facing PHP systems and confirmation from hosting providers and vendors. This week: internal systems and appliances, scheduling vendor firmware (PAN-OS hotfixes from October 26), patch evidence for each system, replacement of the 8.1 and older branches. An update does not remove a compromise that has already occurred, which is why the log review is part of it.

Why this advisory matters from a regulatory perspective

NIS2 (Art. 21(2)(e)) and the BSIG (Section 30(2) no. 5) require vulnerability management, and Section 30(1) sentence 3 BSIG requires documentation of compliance; DORA (Art. 9(4)(f)) requires documented policies for patches and updates, and ISO 27001:2022 requires the management of technical vulnerabilities (A.8.8). Patch evidence for each system with date, package version and restart is the proof for regulators and auditors.

Two bugs in the PHP core, not in an application

The vulnerabilities are not in WordPress, TYPO3 or pfSense but in the runtime underneath, which receives every request before the application runs. Confirmed paths: the session upload progress feature, which is enabled by default, and, in PHP 8.5, the backtraces of fatal errors. This is how the attack works:

Attack flow, simplified

Phase	Approach	Evidence
01 Unauthenticated HTTP request	The adversary sends a crafted request to any PHP endpoint, usually multipart/form-data. No account, no user interaction; the application on top can be flawless.	PHP advisories, Wiz
02 Two bug classes in the core	A (CVE-2026-109618): a memory allocation fails during parsing, PHP aborts with half-built request data, and the shutdown processes invalid state. B (CVE-2026-109619): a fatal abort leaves references to freed memory that the next request to the same worker uses.	GHSA-w8hw-w9fx-j79w, GHSA-6wmf-4rv5-p3rp
03 Memory corruption or reuse	Invalid or reused memory controls read and execution paths. mod_php: demonstrated before any application code. PHP-FPM: requires a matching application operation after startup.	PHP advisories, Datadog

04 Code execution or memory disclosure	Code runs with the privileges of the web worker within its OS. Wiz, Imperva, and container boundaries. Or, without code execution: around 67 KB of worker memory with passwords, tokens and keys as an overlong PHPSESSID cookie. Both cause damage.
05 Optional: further compromise	With code execution or stolen credentials: web shell, file changes, database access. Persistence, C2 and lateral movement require suitable privileges and egress.
06 Business impact	The memory disclosure alone breaches confidentiality. Further steps: skimming, fraud, extortion, manipulation, outage; ransomware requires additional capabilities and access.

On the state of exploitation as of October 23: Amazon confirms a honeypot attempt under embargo as early as October 17 and, since disclosure, has seen several actors with anonymized infrastructure, without knowing whether any of them would have succeeded against real systems. ELLIO saw Mirai loaders, DDoS booters and a Go remote access agent as payloads: botnet operators, not targeted adversaries. VulnCheck has its own exploit for PHP 8.4.10 to 8.4.26 and knows of no other that executes code on several versions; of the hundreds of PoCs on GitHub, only a few are genuine, and one bypasses ASLR on x86-64, but only on 8.4.26 with a standard heap. Sansec therefore considers mass exploitation less likely than assumed on the first day, and Dataminr sees only scans and unusable exploits. On the other hand, NCSC-NL has been investigating “first signs of successful exploitation” since October 22, without details. The BSI knows of no confirmed compromise in regulated industry or the federal administration and considers reports of bypassable patches refuted; the KEV catalog does not list the vulnerabilities. Quiet is still no proof: a successful attack leaves hardly any traces (see below), and according to the BSI, the number of vulnerable systems worldwide remains very high.

Chronology

Oct. 17	Amazon (MadPot) observes an exploitation attempt from an anonymized IP, four days before disclosure
Oct. 20	Details leak out, PHP announces patches for Oct. 21, 14:00 UTC; CERT-EU and CERT.at issue advance warnings in the evening
Oct. 21, 5:02 a.m.	Cloudflare activates WAF rules and observes attack traffic
Oct. 21, morning	BSI warning BITS-H 2026-297981-1232, Orange. NCSC-NL and CERT.at report exploit code in circulation
Oct. 21, 3:34 p.m.	The PHP project publishes both advisories with the fixed versions
Oct. 21, afternoon	Debian, Ubuntu, Red Hat, Netgate and OPNsense ship updates; Dutch authorities, the Charité hospital and parts of berlin.de go offline as a precaution. In the evening, BSI 1.1: “proof of concept public”; Wiz and Calif publish

Oct. 22, morning	Amazon confirms the attempt of Oct. 17 and several actors; CrowdStrike and Datadog: a successful attack leaves almost no trace
Oct. 22, afternoon	BSI 1.2: no confirmed compromises in regulated industry or the federal administration, patch bypass refuted. Sansec: public exploits fail against standard installations. From 4 p.m., NCSC-NL investigates first signs of successful exploitation
Oct. 23, 1:00 a.m.	VulnCheck: own exploit for PHP 8.4.10 to 8.4.26, one public PoC bypasses ASLR only on 8.4.26; Dataminr sees only scans. Palo Alto announces PAN-OS hotfixes from Oct. 26. Still no KEV entry

Every PHP installation that serves web requests

According to the PHP project, all installations that serve web requests are affected. The risk is not the same everywhere, and the order of patching should reflect that.

9.2 CVSS 4.0, both vulnerabilities, critical	42% of PHP hosts on the internet run on EOL branches without an official fix (VulnCheck, 2.6 million hosts)	22.6 million exposed instances according to Unit 42 (Cortex Xpanse)	3/4 BSI criticality Orange, level 3 of 4: act without delay
---	--	--	--

Runtime	Risk according to the PHP advisory		Why
Apache mod_php	Highest, code execution confirmed		The cleanup routine also runs after a failed request start-up; the attack happens before any application code.
PHP-FPM	Affected, session path confirmed		A child process dies on a failed startup; after a successful startup, both vulnerabilities remain reachable.
FrankenPHP, LiteSpeed/LSAPI, php-cgi	Affected, not validated		Named as at risk by the PHP project; FrankenPHP behaves like mod_php, LSAPI like FPM, no exploit demonstrated.

PHP branch	Patched version	Note
8.5	8.5.12	Also check fatal_error_backtraces=0; SimpleXML regression as in 8.4.27
8.4	8.4.27	Debian trixie: 8.4.27-1~deb13u1 (DSA-6574-1). Regression: SimpleXML crash (GH-24488), fix prepared for 8.4.28
8.3	8.3.36	Ubuntu 24.04: 8.3.6~Oubuntu0.24.04.13 (USN-8948-1)

8.2	8.2.35	Support ends on December 31, 2026. Debian bookworm (LTS): 8.2.35-1~deb12u1 (DLA-4848-1)
8.1 and older	no regular fix	Unsupported, vulnerable back to PHP 5. Source backports for 7.4, 8.0 and 8.1; Ubuntu 22.04: 8.1.2-1ubuntu2.28, 20.04 and older only with Ubuntu Pro

With distributions, the package version counts, not the major version: Debian and Ubuntu backport the fix, and php -v still shows 8.3.6. Red Hat (Important, CVSS 3.1: 8.1) published errata for RHEL 8, 9 and 10 on October 21; php:8.4 on RHEL 9 and RHEL 7 are still pending. Rebuild Docker images. For appliances, the vendor delivers the fix, and the status is mixed (Palo Alto, Fortinet, Cisco and Sophos know of no exploitation against their products; NCSC-NL and the BSI list further vendors):

Vendor	Status as of October 23
Palo Alto PAN-OS 10.2 to 12.2	Affected, hotfixes announced from October 26; Panorama and WildFire appliances without a date. Prisma Access, Cortex XDR, GlobalProtect not affected
Fortinet	FortiOS, FortiManager, FortiAnalyzer, FortiWeb not affected; FortiDDoS, FortiSOAR, FortiClient Cloud under review
Cisco	Expressway and URWB affected, fix pending; ISE Portal Builder, Catalyst Center under review; Secure Firewall, IOS, Webex not affected
pfSense, OPNsense	Updates available: pfSense only via shell (Plus 26.07, 26.03.1, CE 2.9.0 with PHP 8.5.12, CE 2.8.1 with 8.3.36), OPNsense 26.7.7 and 26.4.2p3; restart Captive Portal
QNAP	All QTS and QuTS hero branches affected, no fix; until then, disconnect from the internet, use VPN, stop services
Synology	PHP packages 7.4 to 8.4 fixed, Mail Station manually, PHP 7.3 without fix; according to Synology, low risk for DSM-internal PHP
F5, Citrix NetScaler, cPanel	According to the BSI at least partially affected, check the vendor advisory
Sophos	Not affected

Patch first, everything else only bridges the gap

Step	Implementation	Source
Inventory	All PHP installations: version, runtime, reachability, operator. Include container images, hosting packages, appliances and specialized applications; ask suppliers which products contain PHP.	NCSC UK, BACS, BSI

Apply the patch	Fixed version of the branch or the distribution's backport package. PHP project, Then restart every PHP service and every long-running PHP BACS, CIRCL process; rebuild and roll out containers.
Verify the fix	Check the version of the running worker and the package version BSI, BACS, against the distribution advisory, not <code>php -v</code> . Have hosting CIRCL, hosting providers and vendors confirm: IONOS, Hetzner, Mittwald, Host providers' Europe, domainfactory and all-inkl report completion for the plat- status pages, forms they operate themselves; VPS and dedicated servers remain Assetnote the customer's responsibility. Active scanners deliberately trigger the memory error and also flag patched systems with a small <code>memory_limit</code> . Record the patch level, build and restart for each system.
Reduce exposure	If you cannot patch at short notice: disconnect exposed systems BSI, CERT-EU, from the internet and restrict internal applications to VPN. This NCSC UK, costs availability but prevents takeover. Do not disable the PHP CERT-FR, handler, otherwise the web server delivers source code; use a CIRCL maintenance page or a network block.
Interim CVE-2026-109618	<code>memory_limit=-1</code> set administratively (<code>php_admin_value</code> in the PHP project, FPM pool or Apache); no pool, directory or application may reset a CIRCL, CERT-limit. <code>post_max_size</code> 8M without uploads, 32M with uploads, up- FR, BACS <code>load_max_filesize</code> below that, limit <code>max_input_nesting_level</code> . At the proxy, limit body size, number of fields, multipart parts, header lines and nesting, even without Content-Length; block multipart where it is not needed. No <code>RLIMIT_AS</code> or <code>RLIMIT_DATA</code> .
Interim CVE-2026-109619	<code>session.upload_progress.enabled=0</code> in <code>php.ini</code> , the FPM pool or the PHP project, server configuration (not <code>ini_set()</code> , not <code>.user.ini</code>), restart all workers. CIRCL, CERT-PHP 8.5: <code>fatal_error_backtraces=0</code> . Remove invalid INI values that FR generate warnings before initialization; suppressing them does not help. Fix fatal errors and timeouts on reachable paths. If necessary, <code>pm.max_requests=1</code> .
WAF	Cloudflare (including the free plan), Imperva, Akamai and Check BSI, BACS, Point have rules, and Cloudflare already sees variants designed to CIRCL, evade them. They only help if the server is reachable exclusively via Cloudflare the WAF, and they do not replace the patch (BSI, BACS, CIRCL).

Taking systems offline remains the fallback option for systems that receive neither the patch nor the interim measures; the BSI, NCSC UK, NCSC-NL, FBI and CISA recommend it this way, and QNAP explicitly does so for its NAS devices. Since October 22, Sansec no longer considers it necessary for online shops. If you can patch, patch and stay online.

Few traces, so search early and broadly

CrowdStrike describes how a successful attack leaves almost no trace. The worker remains stable, the response is HTTP 200, and the application logs stay empty because no application code runs. The only reliable server-side trace is said to be a single memory-limit error in the error log, which is dismissed as noise in operations. If you do not evaluate these lines, you will not see the attack.

There are still no confirmed indicators: the BSI does not recommend the IOC lists in circulation, and CIRCL calls its observation points hunting approaches. Amazon, VulnCheck and ELLIO have named source IPs but ask that they be investigated rather than blocked, because anonymization and hosting addresses rotate. What holds up is the behavior, not the address. A quick search is still worthwhile: VulnCheck saw scanners with self-named user agents (php2shell-scan, php2shell-verify, php2shell-poc), trivial to change, but found in the access log in seconds.

Instructions for IT, in this order

1. Secure the logs: web server access and error logs, PHP-FPM, application, host/syslog, proxy and WAF, retroactively from October 17 (earliest known attempt, Amazon) and before workers are restarted.
2. Search access logs from October 17 for multipart POSTs with HTTP 500, for burst patterns per source IP, for POSTs with PHP_SESSION_UPLOAD_PROGRESS on endpoints without uploads and for the php2shell user agents.
3. Check the length of Set-Cookie: PHPSESSID in the response logs, if it is logged at all. If it is not logged, note it as a gap: it cannot be reconstructed after the fact.
4. Review error logs for “Allowed memory size exhausted” before script start, for segfaults and for OOM kills of the workers.
5. Host side: new PHP files in the web root, modified .htaccess, executable files in /tmp, child processes of the web user, outbound connections to unknown destinations.
6. Document a positive or negative finding for each system. The negative finding is what you will later need for customer communication and as audit evidence.

Four core signals:

Signal	Log source	Meaning
Individual memory-limit errors in the error log after a multipart request, often without any other anomaly (such as “Allowed memory size exhausted”)	Web server and PHP error log	According to CrowdStrike, the only reliable trace of a successful attack; easily overlooked in operations

Response with Set-Cookie: PHPSESSID Reverse proxy or WAF Memory disclosure has succeeded, and worker kilobytes long instead of 32 characters, with response headers, memory goes to the adversary as a cookie clusters of HTTP 500 after multipart access log for the sta- (Wiz, Imperva: around 67 KB) requests tus codes

Crashing workers, OOM killer, unexpected restarts (segfault in Apache child php-fpm.log, kernel log Failed attempts; correlate with suspicious HTTP processes, FPM children with signal 11) traffic. After updating to 8.4.27 or 8.5.12, think of the SimpleXML regression first ("zend_m-m_heap corrupted", GH-24488)

Child processes of the web worker, new auditd, EDR, file in- Code execution has succeeded; the FBI and files in the web root, outbound connec- tegrity monitoring, CrowdStrike name it as a core signal tions to unknown destinations proxy

For the error log to show anything at all: log_errors=On, a working error_log and error_reporting=E_ALL, according to the PHP project. An overlong PHPSESSID cookie means leaked credentials and tokens, which must be rotated after the patch.

On the "exploits" in circulation: CERT.at and Tenable warn of malicious code in them. The real attack does not write a web shell to the web root; it bends memory in the running process. Any short script that drops a file and calls it is a fake.

What this advisory is based on

All information comes from the linked sources; times in CEST, as of October 23, 2026, 9:00 a.m. (version 1.4). Observations of attack attempts are vendor statements (Amazon, VulnCheck, ELLIO and others) and are marked as such; the mapping to ISO 27001:2022 A.8.8 and to the CIS Controls is our own classification. The advisory will be updated as soon as exploitation is confirmed or reliable indicators are published.

Primary sources online

- PHP: Security advisory, update now
<https://www.php.net/archive/2026.php#2026-10-21-1>
- PHP: GHSA-w8hw-w9fx-j79w (CVE-2026-109618)
<https://github.com/php/php-src/security/advisories/GHSA-w8hw-w9fx-j79w>
- PHP: GHSA-6wmf-4rv5-p3rp (CVE-2026-109619)
<https://github.com/php/php-src/security/advisories/GHSA-6wmf-4rv5-p3rp>
- php-src: GH-24488, SimpleXML regression
<https://github.com/php/php-src/issues/24488>
- BSI: BITS-H 2026-297981-1232, version 1.2 (October 22, 2026, German)
https://www.bsi.bund.de/SharedDocs/Cybersicherheitswarnungen/DE/2026/2026-297981-1032.pdf?__blob=publicationFile&v=8

- ENISA et al.: Joint statement on critical PHP vulnerabilities
<https://www.enisa.europa.eu/news/joint-statement-on-critical-php-vulnerabilities>
- Amazon Threat Intelligence: PHP2Shell exploitation prior to public disclosure
<https://aws.amazon.com/blogs/security/amazon-threat-intelligence-identifies-php2shell-exploitation-prior-to-public-disclosure-in-madpot/>
- VulnCheck: PHP2Shell (update October 23, 2026, 1:00 a.m.)
<https://www.vulncheck.com/blog/php2shell-vulnerabilities>
- ELLIO: PHP2Shell, attacks before the patch
<https://ellio.tech/en/blog/php2shell-what-we-observed-and-what-to-do/>
- BleepingComputer: Amazon and Dataminr on PHP2Shell
<https://www.bleepingcomputer.com/news/security/amazon-says-php2shell-exploitation-began-4-days-before-patches/>
- Sansec: PHP2Shell (update October 22, 2026)
<https://sansec.io/research/php2shell>
- Wiz Research: PHP2Shell
<https://www.wiz.io/blog/php2shell-php-rce>
- Cloudflare: Protections against php2shell
<https://blog.cloudflare.com/php-vulnerabilities-waf-protection/>
- Imperva: PHP2Shell Exploitation Techniques
<https://www.imperva.com/blog/php2shell-vulnerability-updates/>
- CrowdStrike: Threats Delivered via PHP2Shell
<https://www.crowdstrike.com/en-us/blog/crowdstrike-protects-customers-from-php-rce-zero-days>
- The Hacker News: FBI notes on forensic traces
<https://thehackernews.com/2026/10/someone-tried-to-exploit-php2shell-4.html>
- Datadog Security Labs: PHP2Shell
<https://securitylabs.datadoghq.com/articles/php2shell-critical-vulnerabilities-in-php/>
- NCSC-NL: Serious vulnerabilities in PHP (Dutch, as of October 22, 2026, 4:00 p.m.)
<https://www.ncsc.nl/alerts/kwetsbaarheden-php-patches-beschikbaar-update-nu>
- NCSC-NL: php2shell repository with product inventory
<https://github.com/NCSC-NL/php2shell>
- CERT-EU: Security Advisory 2026-018
<https://cert.europa.eu/publications/security-advisories/2026-018/>
- CISA: Critical Vulnerabilities in PHP Require Immediate Patching
<https://www.cisa.gov/news-events/alerts/2026/10/21/critical-vulnerabilities-php-require-immediate-patching>
- CIRCL: TR-102
<https://www.circl.lu/pub/tr-102/>
- CERT-FR: CERTFR-2026-ALE-014
<https://www.cert.ssi.gouv.fr/alerte/CERTFR-2026-ALE-014/>
- NCSC UK: Critical security issue affecting PHP
<https://www.ncsc.gov.uk/news/critical-security-issue-affecting-php>

- BACS: Critical vulnerabilities in PHP (German)
<https://www.bacs.admin.ch/de/warnung-php-de>
- CERT.at: Upcoming critical PHP security update (German)
<https://www.cert.at/de/aktuelles/2026/10/bevorstehendes-kritisches-php-sicherheitsupdate>
- Unit 42: Threat Brief
<https://unit42.paloaltonetworks.com/cve-2026-109618-cve-2026-109619/>
- Patchstack: Major security alert, timeline
<https://patchstack.com/articles/php-major-vulnerability-alert-10-2026/>
- heise: php2shell, many websites offline (German)
<https://www.heise.de/news/php2shell-Updates-trudeln-ein-viele-Webseiten-offline-Angriffe-laufen-11496366.html>
- Palo Alto Networks: CVE-2026-109618 (updated October 23, 2026)
<https://security.paloaltonetworks.com/CVE-2026-109618>
- Fortinet: FG-IR-26-186
<https://www.fortiguard.com/psirt/FG-IR-26-186>
- Cisco: Vulnerabilities in PHP Affecting Cisco Products
<https://sec.cloudapps.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-php-october26-Jj384Daq>
- Sophos: Advisory PHP2Shell
<https://www.sophos.com/de-de/security-advisories/sophos-sa-20261021-php2shell>
- Netgate: Critical PHP Security Update
<https://www.netgate.com/blog/critical-php-security-update>
- OPNsense: 26.7.7 and 26.4.2p3
<https://forum.opnsense.org/index.php?topic=53191.0>
- QNAP: QSA-26-18
<https://www.qnap.com/en/security-advisory/qa-26-18>
- Synology: SA-26:14
https://www.synology.com/de-de/security/advisory/Synology_SA_26_14
- Assetnote: php2shell-scanner
<https://github.com/assetnote/php2shell-scanner>
- severint.net: overview of hosting provider status pages (German, as of October 23, 2026)
<https://www.severint.net/2026/10/22/wie-deutsche-webhoster-auf-php2shell-reagiert-haben/>
- Debian Security Tracker: CVE-2026-109618 (DSA-6574-1, DLA-4848-1)
<https://security-tracker.debian.org/tracker/CVE-2026-109618>
- Ubuntu: CVE-2026-109618 (USN-8948-1)
<https://ubuntu.com/security/CVE-2026-109618>
- Red Hat: CVE-2026-109618
<https://access.redhat.com/security/cve/CVE-2026-109618>
- CISA: Known Exploited Vulnerabilities Catalog
<https://www.cisa.gov/known-exploited-vulnerabilities-catalog>

How we can support you with implementation

This advisory says what needs to be done today. Implementation depends on questions that can only be answered in your environment: where is PHP running without anyone knowing? Do your logs show the traces described at all? Who checks them while the team is busy with the patch window?

We support you with the inventory, check whether your log sources cover the signals, take over the review if your team cannot, and, in case of suspicion, establish the threat picture and contain the incident.

Contact us for an initial conversation.

kontakt@threat-informed.de | www.threat-informed.de

This advisory is part of our threat briefings for companies under NIS2, DORA and critical infrastructure rules in Germany, Austria and Switzerland. TLP:CLEAR, sharing permitted. As of: October 23, 2026, 9:00 a.m., version 1.4.

Glossary

Term	Meaning
Worker	PHP process that handles many requests one after another; its state is at the core of CVE-2026-109619
Multipart	HTTP format for forms with file uploads; the entry point for CVE-2026-109618
Backport	Fix in an older package version; the version number stays the same, the patch is included
KEV	Known Exploited Vulnerabilities: CISA catalog of vulnerabilities proven to be exploited