

THREAT ADVISORY | 21. OKTOBER 2026 · VERSION 1.1

PHP2Shell: Was die kritischen PHP-Lücken für Ihr Unternehmen bedeuten

Am 21. Oktober 2026 hat das PHP-Projekt Patches für zwei kritische Schwachstellen veröffentlicht, über die Angreifer ohne Anmeldung Code auf Webservern ausführen können. Betroffen ist jede PHP-Installation, die Webanfragen bedient, vom WordPress-Blog bis zur Firewall-Oberfläche. Dieses Advisory sagt, was in welcher Reihenfolge zu tun ist und worauf Sie in Ihren Logs achten sollten.

Zielgruppe

Betreiber von Webanwendungen und Appliances (DACH), CISO, IT-Leitung, SOC, Hosting-Verantwortliche

Regulatorik

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

Klassifizierung

TLP:CLEAR, Weitergabe erlaubt

Stand

21. Oktober 2026, 22:30 Uhr, Version 1.1

Quellen

PHP-Projekt, BSI, CERT-EU, CIRCL, CERT-FR, NCSC UK, NCSC-NL, BACS, CERT.at, Unit 42, Wiz, Cloudflare, Imperva, CVE.org, ENISA, Debian, Ubuntu, Red Hat, CISA

Änderungen

Version 1.1 vom 21. Oktober 2026, 22:30 Uhr. Ergänzt: BSI-Warnung Version 1.1 (Proof-of-Concept öffentlich), der Stand der Ausnutzung am Abend (öffentliche Exploits, Angriffsversuche bei Cloudflare, Patchstack und watchTowr, keine bestätigte Übernahme), die Speicheroffenlegung über das PHPSESSID-Cookie als Log-Signal, die WAF-Regeln der großen Anbieter und das Produktinventar von NCSC-NL. Erstfassung 18:00 Uhr.

SECHS SOFORTMASSNAHMEN

- 1**
PHP finden, auch wo es nicht draufsteht
Webserver, Container-Images, Hosting-Pakete, Admin-Oberflächen, Appliances wie pfSense oder OPNsense. Aus dem Internet erreichbare Systeme zuerst.
[CIS Control 1, 2](#)
- 2**
Patchen, jeden Worker neu starten
PHP 8.5.12, 8.4.27, 8.3.36 oder 8.2.35, bei Distributionen das Backport-Paket. Container neu bauen. Prüfen, was läuft, nicht nur php -v.
[CIS Control 7](#)
- 3**
Nicht patchbar? Vom Netz
Exponierte PHP-Systeme vorübergehend vom Internet trennen oder hinter VPN stellen, bis der Patch läuft. Eine WAF allein genügt nicht.
[CIS Control 12](#)
- 4**
Übergangsmaßnahmen des PHP-Projekts
session.upload_progress.enabled=0, memory_limit=-1 administrativ, bei PHP 8.5 fatal_error_backtraces=0, Multipart am Proxy begrenzen. Jede schließt nur einzelne Wege.
[CIS Control 4](#)
- 5**
Logs vor und nach dem Patch durchsehen
Multipart-Anomalien, PHP_SESSION_UPLOAD_PROGRESS, abstürzende Worker, Speichererschöpfung, Kindprozesse des Web-Workers, neue Dateien im Webroot.
[CIS Control 8](#)
- 6**
Alte PHP-Zweige ablösen
PHP 8.1 und älter sind ohne Support, 8.2 endet am 31. Dezember 2026. Die Quell-Backports für 7.4, 8.0 und 8.1 sind eine Notlösung.
[CIS Control 2, 7](#)

Management Summary

Was ist passiert?

Am 20. Oktober kündigte das PHP-Sicherheitsteam ein Notfall-Release an, am 21. Oktober um 15:34 Uhr folgten die Advisories mit den Fix-Versionen. Zwei Fehler im PHP-Kern (CVE-2026-109618 und CVE-2026-109619, CVSS 4.0 jeweils 9,2) erlauben Codeausführung mit den Rechten des Web-Workers, ohne Anmeldung; gegen Apache mit mod_php ist sie demonstriert, ein Pfad auch gegen PHP-FPM. Das BSI stuft die Lage als Orange ein, seit dem Abend mit dem Zusatz „Proof-of-Concept öffentlich“. Exploits liegen auf GitHub, Cloudflare, Patchstack und watchTowr sehen Angriffsversuche; eine bestätigte Übernahme eines Systems meldet bis 22 Uhr niemand, der KEV-Katalog der CISA führt die Lücken nicht.

Was bedeutet das?

Die Fehler liegen in der Laufzeitumgebung, nicht in einer Anwendung: Ein Update von WordPress, TYPO3 oder Nextcloud hilft nicht, das PHP darunter muss gepatcht werden. Betroffen ist jede PHP-Installation, die Webanfragen bedient, auch Firewalls wie pfSense und OPNsense und Admin-Oberflächen von Geräten, bei denen der Betreiber oft nicht weiß, dass PHP im Hintergrund läuft. Laut Wiz liefert die Lücke auch ohne Codeausführung Speicher des Workers zurück, darin Passwörter, Tokens und Schlüssel. Das BSI erwartet eine breite Ausnutzung innerhalb weniger Stunden.

Was müssen wir tun?

Sofort die sechs Maßnahmen oben. Heute: Patch-Fenster für alle aus dem Internet erreichbaren PHP-Systeme, Bestätigung von Hostern und Herstellern. Diese Woche: interne Systeme und Appliances, Patch-Nachweis je System, Ablösung der Zweige 8.1 und älter. Ein Update beseitigt keine bereits erfolgte Kompromittierung, deshalb gehört die Durchsicht der Logs dazu.

Warum dieses Advisory regulatorisch relevant ist

NIS2 (Art. 21 Abs. 2 lit. e) und BSIG (§ 30 Abs. 2 Nr. 5) verlangen Schwachstellenmanagement, § 30 Abs. 1 Satz 3 BSIG die Dokumentation der Einhaltung; DORA (Art. 9 Abs. 4 lit. f) fordert dokumentierte Richtlinien für Patches und Updates, ISO 27001:2022 die Handhabung technischer Schwachstellen (A.8.8). Ein Patch-Nachweis je System mit Datum, Paketversion und Neustart ist der Beleg, den Aufsicht und Auditor sehen wollen.

Zwei Fehler im PHP-Kern, nicht in einer Anwendung

Die beiden Schwachstellen liegen nicht in WordPress, TYPO3 oder pfSense, sondern in der Laufzeitumgebung darunter, die jede Webanfrage entgegennimmt, bevor die Anwendung läuft. Bei Apache mit mod_php läuft der bestätigte Angriff deshalb vollständig ab, bevor eine Zeile Anwendungscode ausgeführt wird.

CVE-2026-109618 betrifft die Behandlung von Speichermangel: Stößt PHP beim Einlesen einer Anfrage an seine Speichergrenze, bricht es ab, während interne Datenstrukturen erst halb aufgebaut sind, und die Aufräumroutine behandelt diesen Zustand als gültig. Angreifer erzeugen den Speichermangel gezielt, etwa mit Multipart-Anfragen ohne brauchbare Längenangabe, sehr vielen Kopfzeilen je Formularteil oder verschachtelten leeren Dateifeldern. Derselbe Fehler gibt laut Wiz auch ohne Codeausführung Speicher des Workers preis, rund 67 Kilobyte je Versuch, als überlanges PHPSESSID-Cookie in der Antwort. CVE-2026-109619 betrifft Zustand, der eine Anfrage überlebt: Ein Worker bearbeitet viele Anfragen nacheinander; bricht eine durch einen fatalen Fehler oder eine Zeitüberschreitung ab, bleiben Verweise auf freigegebenen Speicher stehen, und die nächste Anfrage kann sie nutzen, bevor die Initialisierung sie zurücksetzt. Bestätigte Wege führen über den standardmäßig aktiven Sitzungs-Upload-Fortschritt und bei PHP 8.5 über die Backtraces fataler Fehler; Codeausführung ist gegen mod_php und, über den Sitzungspfad, gegen PHP-FPM belegt.

Zum Stand der Ausnutzung am Abend: Die Entdecker Calif, Wiz und OpenAI haben ihre Analysen veröffentlicht, auf GitHub liegen mehrere Exploits, deren Funktion niemand unabhängig bestätigt hat. Cloudflare beobachtet seit dem frühen Morgen Angriffsverkehr gegen Kunden und Angreifer, die ihre Payloads gegen die WAF-Regeln verändern; Patchstack meldet Vorfälle bei Hosting-Partnern, watchTowr Versuche im Honeypot-Netz. Imperva und Huntress sahen in ihren Kundenbasen bis zum Nachmittag nichts. Eine bestätigte Übernahme eines konkreten Systems nennt niemand, das BSI stuft die Berichte als nicht verifizierbar ein, der KEV-Katalog der CISA führt die Lücken

nicht. Trotzdem geht das BSI von kurzfristiger Ausnutzung aus, und Wiz sagt, warum: Frontier-KI hat den Bau des Exploits von Monaten auf Stunden verkürzt. Die Ausnutzung ist aufwendig (CVSS: AC:H, AT:P) und braucht etwa, dass zwei Anfragen beim selben Worker landen; mehr Worker verzögern den Erfolg, sie verhindern ihn nicht.

Chronologie

20.10.	Details sickern durch, PHP-Sicherheitsteam kündigt für den 21.10., 14:00 UTC, Patches für zwei kritische Lücken an; CERT-EU und CERT.at warnen am Abend vor
21.10., 05:02	Cloudflare schaltet WAF-Regeln für alle Kunden scharf und beobachtet Angriffsverkehr; um 09:34 Uhr melden Hosting-Partner erste Vorfälle an Patchstack
21.10., vormittags	BSI-Warnung BITS-H 2026-297981-1032, Orange. NCSC-NL (11:03 Uhr) und CERT.at melden Exploit-Code im Umlauf, Funktion unbestätigt
21.10., 15:34	PHP-Projekt veröffentlicht beide Advisories mit den Fix-Versionen; NCSC-NL meldet die Patches um 15:50 Uhr als verfügbar
21.10., bis 17:10	Debian DSA-6574-1 und Ubuntu USN-8948-1 abrufbar, Red Hat stuft beide Lücken als Important ein. KEV-Katalog ohne Eintrag (16:04 Uhr). Unit 42 (16:35 Uhr): mindestens eine europäische Regierung nimmt ihre Website vorsorglich offline
21.10., abends	BSI Version 1.1: „Proof-of-Concept öffentlich“, Betreiber in Deutschland trennen Systeme vom Netz. CERT-EU v1.1. Wiz und Calif veröffentlichen ihre Analysen, mehrere Exploits liegen auf GitHub. KEV weiter ohne Eintrag

Jede PHP-Installation, die Webanfragen bedient

Betroffen sind laut PHP-Projekt alle Installationen, die Webanfragen bedienen. Das Risiko ist nicht überall gleich, die Reihenfolge der Patches sollte dem folgen.

9,2	4	22,6	3/4
CVSS 4.0, beide Schwachstellen, kritisch	gepatchte Versionszweige: 8.2, 8.3, 8.4, 8.5	Millionen exponierte Instanzen laut Unit 42 (Cortex Xpanse)	BSI-Kritikalität Orange, Stufe 3 von 4: unverzüglich handeln

Laufzeitumgebung	Risiko laut PHP-Advisory	Warum
Apache mod_php	Am höchsten, Codeausführung bestätigt	Die Aufräumroutine läuft auch nach fehlgeschlagenem Anfragestart; der Angriff findet vor jedem Anwendungscode statt.

PHP-FPM	Betroffen, Sitzungspfad bestätigt	Ein Kindprozess stirbt bei fehlgeschlagenem Start; nach erfolgreichem Start bleiben beide Lücken erreichbar.
FrankenPHP	Wahrscheinlich betroffen	Ruft die Aufräumroutine wie <code>mod_php</code> bedingungslos auf; kein Exploit demonstriert.
LiteSpeed / LSAPI	Bedingt betroffen	Verhalten wie FPM; anwendungsabhängige Wege wahrscheinlich, nicht validiert.
php-cgi	Betroffen	Vom PHP-Projekt als gefährdet genannt, ohne eigene Analyse.

PHP-Zweig	Gepatchte Version	Hinweis
8.5	8.5.12	Zusätzlich <code>fatal_error_backtraces=0</code> prüfen
8.4	8.4.27	Debian trixie: 8.4.27-1~deb13u1 (DSA-6574-1)
8.3	8.3.36	Ubuntu 24.04: 8.3.6-0ubuntu0.24.04.13 (USN-8948-1)
8.2	8.2.35	Support endet am 31. Dezember 2026. Debian bookworm um 17 Uhr noch ohne Fix
8.1 und älter	kein regulärer Fix	Ohne Support, laut BSI und CERT-FR bis PHP 5 verwundbar. Minimale Quell-Backports für 7.4, 8.0 und 8.1; Ubuntu 22.04 liefert 8.1.2-1ubuntu2.28

Bei Distributionen zählt die Paketversion, nicht die Hauptversion: Debian und Ubuntu tragen den Fix zurück, `php -v` zeigt weiter 8.3.6. Red Hat stuft beide Lücken als Important ein (CVSS 3.1: 8,1), weil RHEL standardmäßig PHP-FPM nutzt, wo die Ausnutzung in der Praxis schwer sei; auch Red Hat kennt nur das Update als vollständige Abhilfe. remirepo bekommt die Updates laut PHP-Projekt nicht am Tag der Veröffentlichung. Docker-Images neu bauen, Appliance-Hersteller müssen Firmware liefern. Die Entdecker haben die Codeausführung unter anderem gegen WordPress, Drupal, Joomla, Magento, Nextcloud, Roundcube und phpMyAdmin demonstriert; NCSC-NL sammelt betroffene Produkte, auch Zabbix und eingebettete Web-Oberflächen von Netzwerkgeräten, in einem öffentlichen Repository.

Patches zuerst, alles andere überbrückt nur

Schritt	Umsetzung	Quelle
Inventar	Alle PHP-Installationen: Version, Laufzeitumgebung, Erreichbarkeit, Betreiber; betroffen sind alle Versionen bis zurück zu PHP 5. Container-Images, Hosting-Pakete, Appliances und Fachanwendungen einschließen; Lieferanten fragen, welche Produkte PHP enthalten.	NCSC UK, BACS, BSI

Patch einspielen	Fix-Version des Zweigs oder Backport-Paket der Distribution. Danach jeden PHP-Dienst und jeden dauerhaft laufenden PHP-Prozess neu starten; Container neu bauen und ausrollen.	PHP-Projekt, BACS, CIRCL
Behebung prüfen	Version des laufenden Workers und Paketversion gegen das Distributions-Advisory, nicht php -v. Bestätigung von Hostern und Herstellern. Patch-Stand, Advisory, Build und Neustart je System festhalten.	BSI, BACS, CIRCL
Exposition senken	Wer nicht kurzfristig patchen kann: exponierte Systeme vom Internet trennen, interne Anwendungen auf VPN beschränken. Das kostet Verfügbarkeit, verhindert aber die Übernahme. Nicht den PHP-Handler abschalten, sonst liefert der Webserver Quellcode und Zugangsdaten aus; Wartungsseite oder Netzsperrung.	BSI, CERT-EU, NCSC UK, CERT-FR, CIRCL
Übergang CVE-2026-109618	memory_limit=-1 administrativ (php_admin_value im FPM-Pool oder Apache), kein Pool, Verzeichnis oder Anwendung darf ein Limit zurücksetzen; CIRCL warnt vor Ressourcenerschöpfung. post_max_size 8M ohne, 32M mit Uploads, upload_max_filesize darunter, max_input_nesting_level und Zahl der Uploads begrenzen. Am Proxy Body-Größe, Feldzahl, Multipart-Teile, Kopfzeilen und Verschachtelung begrenzen, auch ohne Content-Length; Multipart blocken, wo nicht nötig (unterbindet Uploads und Formulare). Kein RLIMIT_AS oder RLIMIT_DATA.	PHP-Projekt, CIRCL, CERT-FR, BACS
Übergang CVE-2026-109619	session.upload_progress.enabled=0 in php.ini, FPM-Pool oder Serverkonfiguration (nicht ini_set(), nicht .user.ini), alle Worker neu starten. PHP 8.5: fatal_error_backtraces=0. Ungültige oder veraltete INI-Werte entfernen, die vor der Initialisierung Warnungen erzeugen; error_reporting, display_errors oder @ helfen nicht. Fatale Fehler und Zeitüberschreitungen auf erreichbaren Pfaden beheben. Wenn Leistung keine Rolle spielt: pm.max_requests=1.	PHP-Projekt, CIRCL, CERT-FR
WAF	Cloudflare (auch im Free-Tarif), Imperva, Akamai und Check Point haben Regeln; Cloudflare sieht bereits Payload-Varianten gegen seine Regeln. Sie helfen nur, wenn der Server ausschließlich über die WAF erreichbar ist. BSI, BACS und CIRCL: Eine WAF allein genügt nicht und ersetzt den Patch nicht.	BSI, BACS, CIRCL

Vom Netz nehmen ist keine Übertreibung: Laut Unit 42 hat mindestens eine europäische Regierung ihre Website vorsorglich abgeschaltet, NCSC UK empfiehlt dasselbe, wo weder Patch noch Übergangsmaßnahmen rechtzeitig möglich sind. Wer beides nicht kann, muss das System vor dem späteren Patch auf Kompromittierung und Hintertüren prüfen.

Keine bestätigten Indikatoren, aber klare Spuren

Das BSI nennt auch in Version 1.1 keine Angriffsindikatoren und empfiehlt die kursierenden IOC-Listen nicht; CIRCL nennt seine Beobachtungspunkte Hunting-Ansätze, keine validierten IOCs; kein Hersteller hat bis 22 Uhr IP-Adressen, Hashes oder Domänen veröffentlicht. Was sich sagen lässt, folgt aus dem Angriffsweg: Jeder bekannte Auslöser ist ein Multipart-POST, der vor der Anwendung verarbeitet wird; die Ausnutzung erzeugt Fehler auf dem Server und endet, wenn sie gelingt, in Codeausführung mit den Rechten des Web-Workers. Ein Signal ist hart: Die Speicheroffenlegung kommt als überlanges PHPSESSID-Cookie zurück. Die übrigen Spuren nennen CIRCL und Wiz in allgemeiner Form, die Beispiele in Klammern sind unsere Ableitung.

Signal	Log-Quelle	Bedeutung
Antwort mit Set-Cookie: PHPSESSID von Kilobytes Länge statt 32 Zeichen; gehäufte HTTP 500 nach Multipart-Anfragen	Reverse-Proxy oder WAF mit Access-Log für die Statuscodes	Speicheroffenlegung ist gelungen, der Worker-Speicher geht als Cookie an den Angreifer (Wiz, Imperva: rund 67 KB)
Multipart-POSTs ohne brauchbaren Content-Length, mit sehr vielen Kopfzeilen je Teil, verschachtelten oder leeren Dateifeldern, auffällig großen Bodies	WAF, Reverse-Proxy mit Body-Inspektion; das Access-Log zeigt das nicht	Die Wege zum Speichermangel (CVE-2026-109618)
Feld PHP_SESSION_UPLOAD_PROGRESS; Uploads an Endpunkte, die keine annehmen	WAF, Proxy, Anwendungs-Log	Der bestätigte Sitzungspfad (CVE-2026-109619); laut PHP-Projekt kaum genutzt, deshalb prüfenswert
Serien gleichartiger POSTs von einer Quelle in kurzer Folge	Access-Log, WAF	Zwei Anfragen müssen denselben Worker treffen; der Angreifer wiederholt
Abstürzende Worker, OOM-Killer, unerwartete Neustarts (etwa Segmentation fault bei Apache-Kindprozessen, FPM-Kinder mit Signal 11)	Apache error_log, fpm.log, Kernel-Log	Fehlgeschlagene Versuche; mit verdächtigem HTTP-Verkehr korrelieren
Speichererschöpfung vor dem Start eines Skripts, Zeitüberschreitungen, Fatal-Errors, INI-Warnungen beim Anfragestart oder beim Aufräumen (etwa „Allowed memory size exhausted“)	PHP error_log	Die Erzeuger veralteter Zustände; INI-Warnungen sind selbst ein Angriffsweg
Unerwartete Kindprozesse des Web-Workers, neue geplante Aufgaben des Nutzers (etwa www-data startet sh, curl, wget, python oder nc)	auditd, EDR, Sysmon für Linux	Codeausführung ist gelungen; das robusteste Signal

Unautorisierte Änderungen an Web-Dateiintegritätsprüfung, EDR Webshell oder Persistenz Dateien, Zugriffe auf Zugangsdaten (etwa neue .php-Dateien im Webroot, geänderte .htaccess, ausführbare Dateien in /tmp)

Ungewöhnliche ausgehende Verbindungen des Web-Workers (etwa zu unbekannten Zielen, DNS-Anfragen von Servern, die sonst keine stellen) Proxy, NetFlow, DNS-Log Nachladen von Werkzeugen oder Command-and-Control

Die Signale 2 bis 4 brauchen eine Komponente vor PHP, die Anfrage-Bodies inspiziert; für die Prüfung empfiehlt das PHP-Projekt `log_errors=On`, ein funktionierendes `error_log` und `error_reporting=E_ALL`, wobei das Fehlen eines Eintrags kein Beleg für Sicherheit ist. Wer ein überlanges PHPSESSID-Cookie in seinen Antworten findet, sollte davon ausgehen, dass Zugangsdaten und Tokens aus dem Worker-Speicher abgeflossen sind, und sie nach dem Patch wechseln. Bei jedem Fund: Web-, PHP-, Anwendungs- und Host-Logs sichern, bevor jemand neu startet, erst untersuchen, dann bereinigen.

Worauf dieses Advisory beruht

Alle Angaben stammen aus den verlinkten Quellen; Uhrzeiten in MESZ, Stand 21. Oktober 2026, 22:30 Uhr (Version 1.1). Beobachtungen zu Angriffsversuchen sind Herstellerangaben (Cloudflare, Patchstack, watchTowr, Imperva, Huntress) und als solche gekennzeichnet; die Zuordnung zu ISO 27001:2022 A.8.8 und zu den CIS Controls ist unsere Einordnung. Das Advisory wird aktualisiert, sobald eine Ausnutzung bestätigt oder Indikatoren veröffentlicht werden.

Primärquellen online

- PHP: Security advisory, update now (21.10.2026)
<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>
- BSI: BITS-H 2026-297981-1032, Version 1.1 (21.10.2026)
https://www.bsi.bund.de/SharedDocs/Cybersicherheitswarnungen/DE/2026/2026-297981-1032.pdf?__blob=publicationFile&v=4
- Wiz Research: PHP2Shell, technische Analyse der Entdecker (21.10.2026)
<https://www.wiz.io/blog/php2shell-php-rce>
- Cloudflare: Protections against php2shell, WAF-Regeln und beobachteter Angriffsverkehr (21.10.2026)
<https://blog.cloudflare.com/php-vulnerabilities-waf-protection/>

- Imperva: PHP2Shell, Exploitation Techniques (21.10.2026)
<https://www.imperva.com/blog/php2shell-vulnerability-updates/>
- NCSC-NL: php2shell-Repository mit Produktinventar
<https://github.com/NCSC-NL/php2shell>
- CERT-EU: Security Advisory 2026-018
<https://cert.europa.eu/publications/security-advisories/2026-018/>
- 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>
- NCSC-NL: Ernstige kwetsbaarheden in PHP
<https://community.ncsc.nl/p/php>
- BACS: Kritische Schwachstellen in PHP
<https://www.bacs.admin.ch/de/warnung-php-de>
- CERT.at: Bevorstehendes kritisches PHP-Sicherheitsupdate, Update #1
<https://www.cert.at/de/aktuelles/2026/10/bevorstehendes-kritisches-php-sicherheitsupdate>
- Unit 42: Threat Brief, Critical Vulnerabilities in PHP
<https://unit42.paloaltonetworks.com/cve-2026-109618-cve-2026-109619/>
- Patchstack: Major security alert, Live-Zeitstrahl der Hosting-Partner
<https://patchstack.com/articles/php-major-vulnerability-alert-10-2026/>
- watchTowr: Rapid Reaction PHP2Shell, Honeypot-Beobachtungen
<https://watchtowr.com/intelligence/php-remote-code-execution-php2shell/>
- CVE.org: CVE-2026-109618
<https://www.cve.org/CVERecord?id=CVE-2026-109618>
- Debian Security Tracker: CVE-2026-109618 (DSA-6574-1)
<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>

Wie wir Sie bei der Umsetzung unterstützen können

Dieses Advisory sagt, was heute zu tun ist. Die Umsetzung hängt an Fragen, die nur in Ihrer Umgebung zu beantworten sind: Wo läuft PHP, ohne dass es jemand weiß? Zeigen Ihre Logs die beschriebenen Spuren überhaupt? Wer prüft sie, während das Team im Patch-Fenster steckt?

Wir unterstützen bei der Bestandsaufnahme, prüfen, ob Ihre Log-Quellen die Signale abdecken, übernehmen die Durchsicht, wenn Ihr Team nicht kann, und im Verdachtsfall Lagebild und Eindämmung.

Kontaktieren Sie uns für ein erstes Gespräch.

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

Dieses Advisory ist Teil unserer Threat Briefings für Unternehmen unter NIS2, DORA und KRITIS in der DACH-Region. TLP:CLEAR, Weitergabe erlaubt. Stand: 21. Oktober 2026, 22:30 Uhr, Version 1.1.

Glossar

Begriff	Bedeutung
SAPI	Anbindung von PHP an den Webserver: mod_php (Modul im Apache), PHP-FPM (eigener Prozess-Pool), php-cgi, FrankenPHP, LSAPI
Worker	PHP-Prozess, der nacheinander viele Anfragen bearbeitet; sein Zustand ist der Kern von CVE-2026-109619
Multipart	HTTP-Format für Formulare mit Datei-Uploads; der Einstieg für CVE-2026-109618
OOM	Out of Memory: PHP erreicht sein Speicherlimit und bricht die Anfrage ab
Backport	Fix in einer älteren Paketversion; die Versionsnummer bleibt, der Patch ist drin
WAF	Web Application Firewall: filtert HTTP-Anfragen vor der Anwendung
KEV	Known Exploited Vulnerabilities: CISA-Katalog der nachweislich ausgenutzten Schwachstellen