THREAT ADVISORY | 21. OKTOBER 2026 · VERSION 1.4

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 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

23. Oktober 2026, 09:00 Uhr, Version 1.4

Quellen

PHP-Projekt, 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

Änderungen

Version 1.4 vom 23. Oktober 2026, 09:00 Uhr. Neu: NCSC-NL prüft erste Signale erfolgreicher Ausnutzung, Exploitstand von VulnCheck, Herstellerübersicht (PAN-OS, Fortinet, Cisco, pfSense, OPNsense, QNAP, Synology), Debian bookworm und Red Hat korrigiert, Regression in 8.4.27 und 8.5.12. Frühere Fassungen: 1.3 (22.10., 17:30 Uhr), 1.2 (22.10., 10:00 Uhr), 1.1 (21.10., 22:30 Uhr), Erstfassung 21.10., 18:00 Uhr.

SECHS SOFORTMASSNAHMEN

1

PHP finden, auch wo es nicht draufsteht

Webserver, Container-Images, Hosting-Pakete, Admin-Oberflächen, Appliances wie pfSense, OPNsense, PAN-OS oder NAS von QNAP und Synology. 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 21. Oktober hat das PHP-Projekt zwei Fehler im PHP-Kern gepatcht (CVE-2026-109618 und CVE-2026-109619, CVSS 4.0 jeweils 9,2), die Codeausführung mit den Rechten des Web-Workers ohne Anmeldung erlauben. Das BSI stuft die Lage als Orange ein, „Proof-of-Concept öffentlich“. Amazon sah einen Versuch schon am 17. Oktober, seit den Patches greifen mehrere Akteure automatisiert an. Eine bestätigte Übernahme meldet bis zum 23. Oktober niemand: NCSC-NL prüft seit dem 22. Oktober erste Signale erfolgreicher Ausnutzung, das BSI kennt keine Kompromittierung aus regulierter Wirtschaft oder Bundesverwaltung, der KEV-Katalog der CISA führt die Lücken weiter 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 jeder erreichbare PHP-Endpunkt, auch Login-Routen und Health-Checks, dazu Firewalls und Appliances (pfSense, OPNsense, PAN-OS) und NAS-Systeme von QNAP und Synology, bei denen niemand daran denkt, dass PHP darin läuft. Seit dem 22. Oktober zeichnet sich ab: Versuche sind breit und automatisiert, erfolgreiche Angriffe schwer und bisher nicht belegt.

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, Hersteller-Firmware einplanen (PAN-OS-Hotfixes ab dem 26. Oktober), Patch-Nachweis je System, Ablösung der Zweige 8.1 und älter. Ein Update beseitigt keine erfolgte Kompromittierung, deshalb gehört die Log-Durchsicht 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 für Aufsicht und Auditor.

Zwei Fehler im PHP-Kern, nicht in einer Anwendung

Die Schwachstellen liegen nicht in WordPress, TYPO3 oder pfSense, sondern in der Laufzeitumgebung darunter, die jede Anfrage entgegennimmt, bevor die Anwendung läuft. Bestätigte Wege: der standardmäßig aktive Sitzungs-Upload-Fortschritt und bei PHP 8.5 die Backtraces fataler Fehler. So läuft der Angriff ab:

Angriffsablauf, vereinfacht

Phase	Vorgehen	Beleg
01 Unauthentifizierte HTTP-Anfrage	Der Angreifer schickt an einen beliebigen PHP-Endpunkt eine präparierte Anfrage, meist multipart/form-data. Kein Konto, keine Nutzerinteraktion; die Anwendung darüber kann fehlerfrei sein.	PHP-Advisories, Wiz
02 Zwei Fehlerklassen im Kern	A (CVE-2026-109618): Eine Speicherzuteilung scheitert beim Einlesen, PHP bricht mit halb aufgebauten Anfragedaten ab, der Shutdown verarbeitet ungültigen Zustand. B (CVE-2026-109619): Ein fataler Abbruch hinterlässt Verweise auf freigegebenen Speicher, die die nächste Anfrage an denselben Worker benutzt.	GHSA-w8hw-w9fx-j79w, GHSA-6wmf-4rv5-p3rp
03 Speicherkorruption oder Wiederverwendung	Ungültiger oder wiederverwendeter Speicher steuert Lese- und Ausführungspfade. mod_php: vor jedem Anwendungscode demonstriert. PHP-FPM: braucht nach dem Start eine passende Anwendungsoperation.	PHP-Advisories, Datadog

04 Codeausführung oder Speicheroffenlegung	Code läuft mit den Rechten des Web-Workers in dessen OS- und Container-Grenzen. Oder, ohne Codeausführung: rund 67 KB Worker-Speicher mit Passwörtern, Tokens und Schlüsseln als überlanges PHPSESSID-Cookie. Beides ist ein Schaden.	Wiz, Imperva, Sansec
05 Optional: weitere Kompromittierung	Mit Codeausführung oder erbeuteten Zugangsdaten: Webshell, Dateiänderungen, Datenbankzugriff. Persistenz, C2 und Seitwärtsbewegung brauchen passende Rechte und Egress.	CrowdStrike, FBI, CISA
06 Geschäftliche Auswirkung	Schon die Speicheroffenlegung verletzt die Vertraulichkeit. Weitere Schritte: Skimming, Betrug, Erpressung, Manipulation, Ausfall; Ransomware braucht zusätzliche Fähigkeiten und Zugang.	Unit 42, Sansec

Zum Stand der Ausnutzung am 23. Oktober: Amazon bestätigt einen Honeypot-Versuch schon am 17. Oktober unter Embargo und sieht seit der Veröffentlichung mehrere Akteure mit anonymisierter Infrastruktur, ohne zu wissen, ob einer gegen reale Systeme Erfolg gehabt hätte. ELLIO sah als Nutzlast Mirai-Loader, DDoS-Booter und einen Go-Fernzugriffsagenten: Botnetz-Betreiber, keine gezielten Angreifer. VulnCheck hat einen eigenen Exploit für PHP 8.4.10 bis 8.4.26 und kennt sonst keinen, der auf mehreren Versionen Code ausführt; von den Hunderten PoCs auf GitHub seien wenige echt, einer umgehe ASLR auf x86-64, aber nur auf 8.4.26 mit Standard-Heap. Sansec hält Massenausnutzung deshalb für unwahrscheinlicher als am ersten Tag gedacht, Dataminr sieht nur Scans und untaugliche Exploits. Gegenläufig: NCSC-NL prüft seit dem 22. Oktober „erste Signale erfolgreicher Ausnutzung“, ohne Details. Das BSI kennt keine bestätigte Kompromittierung aus regulierter Wirtschaft oder Bundesverwaltung und hält Berichte über umgehbare Patches für widerlegt; der KEV-Katalog führt die Lücken nicht. Ruhe ist trotzdem kein Beleg: Ein erfolgreicher Angriff hinterlässt kaum Spuren (siehe unten), die Zahl verwundbarer Systeme ist laut BSI weltweit weiter sehr hoch.

Chronologie

17.10.	Amazon (MadPot) beobachtet einen Ausnutzungsversuch von einer anonymisierten IP, vier Tage vor der Veröffentlichung
20.10.	Details sickern durch, PHP kündigt Patches für den 21.10., 14:00 UTC an; CERT-EU und CERT.at warnen am Abend vor
21.10., 05:02	Cloudflare schaltet WAF-Regeln scharf und beobachtet Angriffsverkehr
21.10., vormittags	BSI-Warnung BITS-H 2026-297981-1232, Orange. NCSC-NL und CERT.at melden Exploit-Code im Umlauf
21.10., 15:34	PHP-Projekt veröffentlicht beide Advisories mit den Fix-Versionen
21.10., nachm.	Debian, Ubuntu, Red Hat, Netgate und OPNsense liefern Updates; niederländische Behörden, die Charité und Teile von berlin.de gehen vorsorglich offline. Abends BSI 1.1: „Proof-of-Concept öffentlich“; Wiz und Calif veröffentlichen

22.10., vorm.	Amazon bestätigt den Versuch vom 17.10. und mehrere Akteure; CrowdStrike und Datadog: ein erfolgreicher Angriff hinterlässt fast keine Spur
22.10., nachm.	BSI 1.2: keine bestätigten Kompromittierungen aus regulierter Wirtschaft und Bundesverwaltung, Patch-Umgehung widerlegt. Sansec: öffentliche Exploits scheitern an Standardinstallationen. NCSC-NL prüft ab 16 Uhr erste Signale erfolgreicher Ausnutzung
23.10., 01:00	VulnCheck: eigener Exploit für PHP 8.4.10 bis 8.4.26, ein öffentlicher PoC umgeht ASLR nur auf 8.4.26; Dataminr sieht nur Scans. Palo Alto kündigt PAN-OS-Hotfixes ab 26.10. an. 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
CVSS 4.0, beide Schwachstellen, kritisch

42 %
der PHP-Hosts im Internet laufen auf EOL-Zweigen ohne offiziellen Fix (VulnCheck, 2,6 Mio. Hosts)

22,6
Millionen exponierte Instanzen laut Unit 42 (Cortex Xpanse)

3/4
BSI-Kritikalität Orange, Stufe 3 von 4: unverzüglich handeln

Laufzeitumgebung	Risiko laut PHP-Advisory	Warum
Apache mod_php	Am höchsten, Code-ausführung 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, LiteSpeed/LSAPI, php-cgi	Betroffen, nicht validiert	Vom PHP-Projekt als gefährdet genannt; FrankenPHP verhält sich wie mod_php, LSAPI wie FPM, kein Exploit demonstriert.

PHP-Zweig	Gepatchte Version	Hinweis
8.5	8.5.12	Zusätzlich fatal_error_backtraces=0 prüfen; SimpleXML-Regression wie 8.4.27

8.4	8.4.27	Debian trixie: 8.4.27-1~deb13u1 (DSA-6574-1). Regression: SimpleXML-Absturz (GH-24488), Fix für 8.4.28 vorbereitet
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 (LTS): 8.2.35-1~deb12u1 (DLA-4848-1)
8.1 und älter	kein regulärer Fix	Ohne Support, bis PHP 5 verwundbar. Quell-Backports für 7.4, 8.0 und 8.1; Ubuntu 22.04: 8.1.2-1ubuntu2.28, 20.04 und älter nur mit Ubuntu Pro

Bei Distributionen zählt die Paketversion, nicht die Hauptversion: Debian und Ubuntu tragen den Fix zurück, php -v zeigt weiter 8.3.6. Red Hat (Important, CVSS 3.1: 8,1) hat am 21. Oktober Errata für RHEL 8, 9 und 10 veröffentlicht; php:8.4 auf RHEL 9 und RHEL 7 stehen aus. Docker-Images neu bauen. Bei Appliances liefert der Hersteller, der Stand ist uneinheitlich (Palo Alto, Fortinet, Cisco und Sophos kennen keine Ausnutzung gegen ihre Produkte; weitere Hersteller führen NCSC-NL und BSI):

Hersteller	Stand 23. Oktober
Palo Alto PAN-OS 10.2 bis 12.2	Betroffen, Hotfixes ab 26. Oktober angekündigt; Panorama und WildFire-Appliances ohne Termin. Prisma Access, Cortex XDR, GlobalProtect nicht betroffen
Fortinet	FortiOS, FortiManager, FortiAnalyzer, FortiWeb nicht betroffen; FortiDDoS, FortiSOAR, FortiClient Cloud in Prüfung
Cisco	Expressway und URWB betroffen, Fix offen; ISE Portal Builder, Catalyst Center in Prüfung; Secure Firewall, IOS, Webex nicht betroffen
pfSense, OPNsense	Updates da: pfSense nur per Shell (Plus 26.07, 26.03.1, CE 2.9.0 mit PHP 8.5.12, CE 2.8.1 mit 8.3.36), OPNsense 26.7.7 und 26.4.2p3; Captive Portal neu starten
QNAP	Alle QTS- und QuTS-hero-Zweige betroffen, kein Fix; bis dahin vom Internet trennen, VPN, Dienste stoppen
Synology	PHP-Pakete 7.4 bis 8.4 gefixt, Mail Station manuell, PHP 7.3 ohne Fix; DSM-internes PHP laut Synology geringes Risiko
F5, Citrix NetScaler, cPanel	Laut BSI mindestens teilweise betroffen, Hersteller-Advisory prüfen
Sophos	Nicht betroffen

Patchen zuerst, alles andere überbrückt nur

Schritt	Umsetzung	Quelle
Inventar	Alle PHP-Installationen: Version, Laufzeitumgebung, Erreichbarkeit, Betreiber. 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. Hoster und Hersteller bestätigen lassen: IONOS, Hetzner, Mittwald, Host Europe, domainfactory und all-inkl melden Vollzug für selbst betriebene Plattformen, VPS und Dedicated Server bleiben Kundensache. Aktive Prüf-Scanner lösen den Speicherfehler absichtlich aus und schlagen bei kleinem memory_limit auch auf gepatchten Systemen an. Patch-Stand, Build und Neustart je System festhalten.	BSI, BACS, CIRCL, Statusseiten der Hoster, Assetnote
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 aus; Wartungsseite oder Netzsperre.	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. post_max_size 8M ohne, 32M mit Uploads, upload_max_filesize darunter, max_input_nesting_level begrenzen. Am Proxy Body-Größe, Feldzahl, Multipart-Teile, Kopfzeilen und Verschachtelung begrenzen, auch ohne Content-Length; Multipart blocken, wo nicht nötig. 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 INI-Werte entfernen, die vor der Initialisierung Warnungen erzeugen; Unterdrücken hilft nicht. Fatale Fehler und Zeitüberschreitungen auf erreichbaren Pfaden beheben. Notfalls pm.max_requests=1.	PHP-Projekt, CIRCL, CERT-FR
WAF	Cloudflare (auch Free), Imperva, Akamai und Check Point haben Regeln, Cloudflare sieht bereits Varianten dagegen. Sie helfen nur, wenn der Server ausschließlich über die WAF erreichbar ist, und ersetzen den Patch nicht (BSI, BACS, CIRCL).	BSI, BACS, CIRCL, Cloudflare

Vom Netz nehmen bleibt die Rückfalloption für Systeme, die weder Patch noch Übergangsmaßnahmen bekommen; BSI, NCSC UK, NCSC-NL, FBI und CISA empfehlen es so, QNAP für seine NAS ausdrücklich. Sansec hält es für Shops seit dem 22. Oktober nicht mehr für nötig. Wer patchen kann, patcht und bleibt online.

Wenig Spur, deshalb früh und breit suchen

CrowdStrike beschreibt, dass ein erfolgreicher Angriff fast keine Spur hinterlässt. Der Worker bleibt stabil, die Antwort ist HTTP 200, die Anwendungslogs bleiben leer, weil kein Anwendungscode läuft. Die einzige verlässliche serverseitige Spur sei ein einzelner memory-limit-Fehler im Error-Log, der im Betrieb als Rauschen abgetan wird. Wer diese Zeilen nicht auswertet, sieht den Angriff nicht.

Bestätigte Indikatoren gibt es trotzdem nicht: Das BSI empfiehlt die kursierenden IOC-Listen nicht, CIRCL nennt seine Beobachtungspunkte Hunting-Ansätze. Amazon, VulnCheck und ELLIO haben Quell-IPs genannt, bitten aber, sie zu untersuchen statt zu blocken, weil Anonymisierungs- und Hosting-Adressen rotieren. Tragfähig ist das Verhalten, nicht die Adresse. Eine schnelle Suche lohnt trotzdem: VulnCheck sah Scanner mit selbstbenannten User-Agents (php2shell-scan, php2shell-verify, php2shell-poc), trivial zu ändern, im Access-Log aber in Sekunden zu finden.

Auftrag an die IT, in dieser Reihenfolge

1. Logs sichern: Webserver-Access und -Error, PHP-FPM, Application, Host/Syslog, Proxy und WAF, rückwirkend ab dem 17. Oktober (frühester bekannter Versuch, Amazon) und bevor Worker neu gestartet werden.
2. Access-Logs ab dem 17. Oktober nach Multipart-POSTs mit HTTP 500 durchsuchen, nach Burst-Mustern je Quell-IP, nach POSTs mit PHP_SESSION_UPLOAD_PROGRESS auf Endpunkten ohne Upload und nach den php2shell-User-Agents.
3. Set-Cookie: PHPSESSID in den Response-Logs auf Länge prüfen, sofern es überhaupt geloggt wird. Wird es nicht geloggt, als Lücke notieren: es ist nachträglich nicht rekonstruierbar.
4. Error-Logs auf „Allowed memory size exhausted“ vor dem Skriptstart, auf Segfaults und auf OOM-Kills der Worker durchsehen.
5. Host-Seite: neue PHP-Dateien im Webroot, geänderte .htaccess, ausführbare Dateien in /tmp, Kindprozesse des Web-Users, ausgehende Verbindungen zu unbekannten Zielen.
6. Je System einen Positiv- oder Negativbefund dokumentieren. Der Negativbefund ist das, was Sie später für die Kundenkommunikation und als Auditnachweis brauchen.

Vier Kernsignale:

Signal	Log-Quelle	Bedeutung
Einzelne memory-limit-Fehler im Error-Log nach einer Multipart-Anfrage, oft ohne weitere Auffälligkeit (etwa „Allowed memory size exhausted“)	Webserver- und PHP-Error-Log	Laut CrowdStrike die einzige verlässliche Spur eines erfolgreichen Angriffs; wird im Betrieb leicht übersehen
Antwort mit Set-Cookie: PHPSESSID von Kilobytes Länge statt 32 Zeichen, gehäufte HTTP 500 nach Multipart-Anfragen	Reverse-Proxy oder WAF mit Antwort-Headern, Access-Log für die Statuscodes	Speicheroffenlegung ist gelungen, Worker-Speicher geht als Cookie an den Angreifer (Wiz, Imperva: rund 67 KB)
Abstürzende Worker, OOM-Killer, unerwartete Neustarts (Segfault bei Apache-Kindprozessen, FPM-Kinder mit Signal 11)	Apache error_log, php-fpm.log, Kernel-Log	Fehlgeschlagene Versuche; mit verdächtigem HTTP-Verkehr korrelieren. Nach dem Update auf 8.4.27 oder 8.5.12 zuerst an die SimpleXML-Regression denken („zend_mm_heap corrupted“, GH-24488)
Kindprozesse des Web-Workers, neue Dateien im Webroot, ausgehende Verbindungen zu unbekannten Zielen	auditd, EDR, Dateiintegritätsprüfung, Proxy	Codeausführung ist gelungen; FBI und CrowdStrike nennen es als Kernsignal

Damit das Error-Log überhaupt etwas zeigt: log_errors=On, ein funktionierendes error_log und error_reporting=E_ALL, so das PHP-Projekt. Ein überlanges PHPSESSID-Cookie bedeutet abgeflossene Zugangsdaten und Tokens, die nach dem Patch zu wechseln sind.

Zu den kursierenden „Exploits“: CERT.at und Tenable warnen vor Schadcode darin. Der echte Angriff schreibt keine Webshell ins Webroot, er verbiegt Speicher im laufenden Prozess; jedes kurze Skript, das eine Datei ablegt und aufruft, ist eine Fälschung.

Worauf dieses Advisory beruht

Alle Angaben stammen aus den verlinkten Quellen; Uhrzeiten in MESZ, Stand 23. Oktober 2026, 09:00 Uhr (Version 1.4). Beobachtungen zu Angriffsversuchen sind Herstellerangaben (Amazon, VulnCheck, ELLIO und andere) 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 belastbare Indikatoren veröffentlicht werden.

Primärquellen 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 (22.10.2026)
  https://www.bsi.bund.de/SharedDocs/Cybersicherheitswarnungen/DE/2026/2026-297981-1032.pdf?__blob=publicationFile&v=8
- ENISA u. a.: 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 23.10.2026, 01:00 Uhr)
  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 und Dataminr zu PHP2Shell
  https://www.bleepingcomputer.com/news/security/amazon-says-php2shell-exploitation-began-4-days-before-patches/
- Sansec: PHP2Shell (Update 22.10.2026)
  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-Hinweise zur Spurenlage
  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: Ernstige kwetsbaarheden in PHP (Stand 22.10.2026, 16:00 Uhr)
  https://www.ncsc.nl/alerts/kwetsbaarheden-php-patches-beschikbaar-update-nu
- 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/
- 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: Kritische Schwachstellen in PHP
  https://www.bacs.admin.ch/de/warnung-php-de
- CERT.at: Bevorstehendes kritisches PHP-Sicherheitsupdate
  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, Zeitstrahl
  https://patchstack.com/articles/php-major-vulnerability-alert-10-2026/
- heise: php2shell, viele Webseiten offline
  https://www.heise.de/news/php2shell-Updates-trudeln-ein-viele-Webseiten-offline-Angriffe-laufen-11496366.html
- Palo Alto Networks: CVE-2026-109618 (aktualisiert 23.10.2026)
  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 und 26.4.2p3
  https://forum.opnsense.org/index.php?topic=53191.0
- QNAP: QSA-26-18
  https://www.qnap.com/en/security-advisory/qsa-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: Hoster-Statusseiten im Überblick (Stand 23.10.2026)
  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

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: 23. Oktober 2026, 09:00 Uhr, Version 1.4.

Glossar

Begriff	Bedeutung
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
Backport	Fix in einer älteren Paketversion; die Versionsnummer bleibt, der Patch ist drin
KEV	Known Exploited Vulnerabilities: CISA-Katalog der nachweislich ausgenutzten Schwachstellen