



La falla PHP trovata dall'AI prima degli altri

Di **Fabio Angelici** • Pubblicato il 22 ottobre 2026 • 7 min di lettura



RISPOSTA RAPIDA

Due vulnerabilità critiche nel motore di PHP, nascoste da oltre dieci anni e trovate con l'intelligenza artificiale. Riguardano circa il 70% dei siti mondiali e gli attacchi sono già iniziati. Perché aggiornare WordPress non serve e quali versioni di PHP installare subito.

Mercoledì diversi siti istituzionali europei sono andati offline senza troppe spiegazioni. Non un attacco riuscito: una ritirata precauzionale.

Il motivo è una coppia di vulnerabilità nel motore di PHP, battezzate PHP2Shell. Stavano nel codice da oltre dieci anni e nessuno se n'era accorto.

Le ha trovate un'intelligenza artificiale.

Partiamo dalle basi: cosa è PHP

PHP è un linguaggio di programmazione che gira sul server. Quando apri una pagina, il tuo browser non vede mai il codice PHP: vede solo l'HTML che quel codice ha prodotto un istante prima, assemblando contenuti presi dal database.

Nasce nei primi anni Novanta, inizialmente come acronimo di Personal Home Page.

Un nome che oggi fa quasi tenerezza, considerando dove è arrivato.

Perché PHP sta sotto a una fetta enorme del web pubblico: le stime parlano di circa il 70% dei siti mondiali.

WordPress è scritto in PHP. Drupal, Joomla, Magento, PrestaShop e Nextcloud pure.

Anche Laravel, il framework che uso per i progetti su misura, gira su PHP.

Questo è il punto che rende la faccenda seria. Un bug in un plugin colpisce chi usa quel plugin. Un bug nel motore di PHP colpisce, in linea di principio, chiunque abbia un sito.

Cosa è successo

Il 21 ottobre 2026, alle 16 ora italiana, il PHP Group ha rilasciato quattro versioni di emergenza.

Correggono due vulnerabilità: CVE-2026-109618 e CVE-2026-109619, entrambe valutate 9.2 su 10 nella scala CVSS 4.0.

Si tratta in tutti e due i casi di esecuzione di codice da remoto senza autenticazione.

Tradotto: un attaccante manda una richiesta HTTP costruita in un certo modo e riesce a far eseguire al server codice suo.

Non serve una password, non serve un account, non serve che il sito abbia plugin vulnerabili. Il codice parte prima che l'applicazione entri in gioco.

Le due falle riguardano la gestione della memoria. La prima sfrutta il modo in cui PHP reagisce quando esaurisce la memoria a disposizione:

in quel momento l'interruzione può avvenire mentre le strutture interne della richiesta sono costruite solo a metà, e la pulizia successiva le tratta come se fossero valide.

La seconda nasce da stato interno legato a una richiesta che sopravvive alla memoria che lo conteneva:

una richiesta successiva, gestita dallo stesso processo, può ritrovarselo tra i piedi prima che venga azzerato.

Roba di basso livello, invisibile a chi scrive codice applicativo. È esattamente il tipo di difetto che sta sotto il radar per anni.

Come sono state scoperte

Qui sta la parte che secondo me conta più della falla stessa.

Le vulnerabilità sono emerse nell'ambito di un programma di ricerca chiamato Patch the Planet,

in cui la società di sicurezza Calif e Wiz hanno lavorato insieme alla PHP Foundation e a OpenAI.

A individuarle è stato il modello GPT-5.6 Sol di OpenAI, che ha analizzato il codice sorgente del motore.

Secondo quanto riportato da Wiz, quelle falle erano nel codice da più di un decennio e per un essere umano sarebbero state estremamente complicate da sfruttare. L'AI non ha scoperto qualcosa di nuovo: ha reso accessibile qualcosa che era già lì e che richiedeva uno sforzo fuori portata.

Il resto è cronaca poco esaltante. Il rilascio era programmato per il 22 ottobre, l'embargo è saltato in anticipo, i dettagli sono circolati sui social e le patch sono state pubblicate di corsa il giorno prima.

Amazon Threat Intelligence riferisce di aver individuato un tentativo di sfruttamento già il 17 ottobre, quattro giorni prima della divulgazione pubblica.

Cloudflare ha poi rilevato milioni di tentativi.

Una precisazione che mi sembra onesta: l'AI qui è servita a chi difende, non solo a chi attacca.

La falla esisteva comunque, e qualcuno l'avrebbe trovata prima o poi.

Il fatto che l'abbia trovata un gruppo che poi ha coordinato la patch con la PHP Foundation è la parte buona della storia.

La parte scomoda è che lo stesso strumento funziona benissimo anche dall'altra parte del tavolo.

Come si risolve (e perché aggiornare il CMS non basta)

Qui va detta la cosa più importante dell'articolo, perché la vedo fraintesa ovunque.

Aggiornare WordPress non chiude questa vulnerabilità.

Nemmeno aggiornare plugin e temi. Nemmeno mettere il sito in manutenzione, perché PHP elabora la richiesta prima che WordPress decida cosa mostrare.

La falla sta un piano più sotto, nel linguaggio su cui il CMS gira. L'unica correzione reale è aggiornare PHP a una di queste versioni:

- PHP 8.5.12
- PHP 8.4.27
- PHP 8.3.36
- PHP 8.2.35

Dopo l'aggiornamento vanno riavviati i processi PHP, altrimenti i worker già in esecuzione continuano a girare con il codice vecchio.

Se sei su un hosting condiviso, la versione di PHP si cambia dal pannello di controllo, e molti provider hanno già aggiornato in autonomia.

Su WordPress la versione attiva la vedi in Strumenti, Salute del sito, Informazioni, sezione Server.

E qui arriva la nota dolente. Le patch ufficiali coprono solo i rami dall'8.2 in su.

Chi gira su PHP 8.1, 8.0, 7.x o 5.x dal progetto PHP non riceve niente:

secondo le statistiche di WordPress.org, al 22 ottobre il 35,8% dei siti WordPress si trova in questa condizione. Più di uno su tre.

Per quei casi la strada è una sola, ed è quella che andava fatta comunque: migrare a un ramo supportato.

Esistono backport di comunità e patch non ufficiali in giro, ma applicare una patch presa da un repository sconosciuto al motore che serve il tuo sito

è un rimedio che rischia di costarti più della malattia.

Stesso discorso per i plugin miracolosi spuntati in queste ore: non esiste un plugin WordPress capace di correggere un bug che sta dentro PHP.

A questo punto il CMS va aggiornato comunque, perché le versioni recenti di WordPress, Joomla, Drupal, Magento e PrestaShop sono quelle compatibili con i rami PHP ancora supportati.

L'ordine corretto è: prima verifichi che temi e plugin reggano la versione nuova di PHP, poi sposti il sito.

Al contrario rischi una schermata bianca.

Un'ultima cosa, e non è un dettaglio: la patch chiude la porta, non ripulisce la casa.

Se il tuo server era esposto nei giorni scorsi, dopo l'aggiornamento vale la pena controllare i file modificati di recente, gli utenti amministratori, le attività pianificate e i log di accesso. E cambiare le credenziali, incluse chiavi API e token.

Domande frequenti

Come faccio a sapere quale versione di PHP usa il mio sito?

Su WordPress la trovi in Strumenti, Salute del sito, Informazioni, sezione Server. Su altri CMS di solito è nel pannello di amministrazione o in quello dell'hosting.

Su un VPS basta un `php -v` lanciato dalla console

Il mio hosting dice di aver già aggiornato. Devo fare altro?

Verifica che il tuo sito stia davvero usando la versione nuova, perché su molti hosting la versione PHP si imposta per singolo dominio.

Un aggiornamento a livello di server non sposta automaticamente i siti configurati su un ramo vecchio.

Il sito è piccolo e non interessa a nessuno. Sono al sicuro?

No, e questo è il fraintendimento più comune. Gli attacchi di questo tipo sono automatizzati: scansionano l'intero spazio degli indirizzi senza guardare chi sei.

Un server compromesso serve a mandare spam o a colpire altri sistemi, e il tuo traffico non c'entra nulla.

Ho un sito fatto su misura, senza CMS. Sono escluso?

No. La falla è in PHP, non nel software che ci costruisci sopra. Vale per Laravel, per Symfony e per qualsiasi codice scritto a mano.

Cosa fare adesso

Se hai un sito, l'unica cosa da fare oggi è verificare su quale versione di PHP gira. Sono tre minuti.

Se il numero che leggi non è 8.5.12, 8.4.27, 8.3.36 o 8.2.35, hai una cosa da sistemare prima di fine giornata.

FONTI

1. [Help Net Security: PHP2Shell: Critical PHP vulnerabilities patched, update now!](#)
2. [Reuters via Internazionale: Government websites go dark after AI-discovered flaw](#)
3. [Fortune: AI tools discovered the PHP vulnerability that could take down 70% of the internet](#)
4. [SOCRadar: PHP2Shell \(CVE-2026-109618 & CVE-2026-109619\): Affected Versions & Fix](#)
5. [Punto Informatico: PHP rilascia patch per le vulnerabilità scoperte da GPT-5.6 Sol](#)

Letto originariamente su <https://fabioangelici.com/blog/la-falla-php-trovata-dallai-prima-degli-altri>

PDF generato il 22 ottobre 2026