

MERCHANTS, DOSTAWCY I ISO

PHP2Shell: co nowe luki w PHP oznaczają dla każdego, kto przyjmuje płatności kartą

Opublikowano 22 października 2026

Dwie krytyczne luki w PHP mogą pozwolić nieautoryzowanemu atakującemu uruchomić kod na serwerze WWW. Dla sprzedawcy to ryzyko dla integralności procesu płatności, danych uwierzytelniających do płatności i dostępności. Dla dostawcy to ryzyko portfelowe. Co się stało, co sprawdzić i o co zapytać hosting.

21 października 2026 roku projekt PHP wydał awaryjne poprawki dla dwóch krytycznych luk w samym języku, oznaczonych jako CVE-2026-109618 i CVE-2026-109619 i nazwanych potocznie **PHP2Shell**. Obie mają ocenę 9,2 w skali CVSS 4.0. Każda z nich może pozwolić nieautoryzowanemu atakującemu uruchomić kod na dotkniętym serwerze WWW, a luki tkwią w PHP, a nie w jednej konkretnej aplikacji zbudowanej na nim.

Dla sprzedawców oznacza to ryzyko dla integralności procesu płatności, danych uwierzytelniających do płatności i dostępności. Dla dostawców płatności i ISO oznacza to możliwe narażenie całego portfela sprzedawców. Oto, co się stało, co warto sprawdzić i o co zapytać dostawcę hostingu.

PHP obsługuje bardzo dużą część internetu, w tym większość oprogramowania sklepowego używanego przez małych i średnich sprzedawców. Dlatego jest to historia o płatnościach w takim samym stopniu, jak o bezpieczeństwie.

Co znaleziono

- **CVE-2026-109618** to błąd uszkodzenia pamięci. Jeśli PHP zabraknie pamięci w trakcie przygotowywania żądania, późniejsze porządkowanie może potraktować dane zbudowane tylko częściowo jako prawidłowe.
- **CVE-2026-109619** to błąd nieaktualnego stanu, typu use-after-free. PHP może zachować stan wewnętrzny wskazujący na pamięć żądania, która została już zwolniona, a dalsze przetwarzanie może jej użyć. PHP wskazuje jako skutki zdalne wykonanie kodu i ujawnienie wrażliwej pamięci.
- **Zasięg.** Najstarszy dotknięty kod sięga PHP 5.4. Projekt PHP twierdzi, że zagrożona jest każda instalacja obsługująca żądania WWW, niezależnie od tego, czy działa na PHP-FPM, Apache z

mod_php, CGI, FrankenPHP czy LiteSpeed. PHP ocenia ryzyko jako najwyższe w przypadku mod_php, gdzie pokazany atak może nastąpić, zanim uruchomi się jakikolwiek kod samej witryny. To opisuje atak, który zademonstrowano, a nie twierdzenie, że inne konfiguracje są bezpieczne.

- **Poprawione wersje.** PHP 8.5.12, 8.4.27, 8.3.36 i 8.2.35. Dla wersji od 7.4 do 8.1 nie ma oficjalnego wydania, są tylko minimalne backporty kodu źródłowego. Wersje od 5.4 do 7.3 osiągnęły koniec wsparcia i projekt PHP w ogóle ich nie poprawi. Niektóre dystrybucje Linuksa i dostawcy wsparcia długoterminowego przenoszą poprawki do starszych gałęzi, dlatego przed uznaniem, że dana instalacja jest niezła, należy sprawdzić biuletyn bezpieczeństwa własnego dostawcy.

Jak wygląda sytuacja

Szczegóły wyciekły 20 października, dzień przed planowanym wydaniem, które samo zostało już przyspieszone z 22 października. Zespół analizy zagrożeń Amazon zgłosił próbę ataku na swoje honeypoty 17 października, cztery dni przed publicznym wydaniem, oraz kolejne od tamtej pory, najwyraźniej od wielu podmiotów. Cloudflare i watchTower donoszą o ruchu atakującym, a watchTower twierdzi, że odtworzył obie luki. Publiczny kod exploita istnieje, choć jego niezawodność nie jest zweryfikowana, a projekt PHP oświadczył przy wydaniu, że nie zidentyfikował działającego exploita. Nie ustalono jeszcze, czy dotychczas widziane próby powiodłyby się na prawdziwych systemach produkcyjnych.

To wiarygodne i pilne zagrożenie. Próby wykorzystania i kod proof-of-concept same w sobie nie dowodzą, że dane płatnicze sprzedawców zostały skradzione, a na 22 października nie znamy żadnego potwierdzonego naruszenia danych płatniczych, które dałoby się przypisać PHP2Shell. Należy załatać dotknięte systemy już teraz, zbadać wszystko, co wygląda podejrzanie, i zachować odpowiednie logi. Brak potwierdzonego naruszenia nie jest powodem do zwłoki.

Dlaczego to ważne dla płatności kartą

Sprzedawca nie musi uruchamiać własnego kodu PHP, by zostać dotknięty. Wystarczy oprogramowanie pod sklepem, a skutki docierają do strony płatności na cztery sposoby.

1. **Serwer budujący stronę płatności.** Wykonanie kodu na tym serwerze pozwala atakującemu zmienić to, co robi strona płatności. Wstrzyknięcie skryptu kopiującego dane karty w trakcie ich wpisywania przez klientów (często nazywane skimmingiem lub atakiem w stylu Magecart) to od dawna znana droga, którą skradzione dane kart opuszczają sklepy w PHP.
2. **Dane uwierzytelniające i klucze.** Klucze API bramki płatniczej, sekrety webhooków i loginy administratora zwykle znajdują się w konfiguracji sklepu. Kto może uruchomić kod na serwerze, może je odczytać.
3. **Oszustwa i nadużycia.** Skradzione klucze można wykorzystać do zlecania zwrotów lub wykonywania transakcji testowych. Zhakowana witryna może też hostować drugą, ukrytą stronę płatności, która dla dostawcy wygląda bardzo podobnie do [transaction launderingu](#).

4. **Przestój.** Nawet nieudany atak może powodować awarie procesów roboczych. Sklep, który nie działa, nie przyjmuje zamówień, a odtwarzanie trwające dni może zakłócić cykle [rozliczania subskrypcji](#) i wywołać zwroty oraz spory.

Sprzedawcy, którzy przekierowują na hostowaną stronę płatności dostawcy albo osadzają pola samego dostawcy, są w lepszej sytuacji: numer karty nigdy nie trafia na ich serwer. Nadal jednak narażeni są na opisane wyżej ryzyko związane z panelem administratora, zwrotami i kluczami API.

Co sprawdzić w tym tygodniu

- **Ustalić, jaki PHP działa.** Na własnym serwerze wystarczy uruchomić `php -v`. W hostingu zarządzanym lub współdzielonym trzeba zajrzeć do panelu albo zapytać dostawcę. Wtyczki i oprogramowanie sklepowe tego nie raportują, a sklep może działać na starej wersji PHP przez lata po aktualizacji swojego oprogramowania.
- **Zastosować poprawkę dostawcy.** W większości dystrybucji Linuksa i hostingów oznacza to aktualizację pakietu od dostawcy, a następnie restart PHP, aby zmiana zaczęła działać. Nie należy polegać wyłącznie na numerze wersji PHP, bo dystrybucje często przenoszą poprawkę bez jego zmiany.
- **Przy PHP 7.3 lub starszym zaplanować migrację.** Projekt PHP nie poprawi tych wersji. Warto zapytać dostawcę hostingu lub programistę, czy ich dostawca przenosi poprawkę, i poprosić o datę, a nie o deklarację zamiaru.
- **Środki łagodzące są tymczasowe.** Projekt PHP opisuje ustawienia zawężające każdą z luk, a zaporę aplikacji webowej lub ograniczenia sieciowe mogą zmniejszyć narażenie na czas przygotowania aktualizacji. Żadne z nich nie jest pełną poprawką. Jak najszybciej należy przejść na poprawioną wersję lub zastosować zweryfikowaną łatkę bezpieczeństwa dystrybucji, a jeśli system dostępny z internetu nie może zostać szybko załatany, rozważyć ograniczenie jego ekspozycji.
- **Zmienić sekrety, jeśli nie można wykluczyć naruszenia.** Klucze bramki płatniczej, sekrety webhooków, hasła administratora i dane dostępowe do bazy. Sprawdzić, czy nie pojawili się nowi administratorzy, czy nie zmieniono plików strony płatności i czy na stronie płatności nie ma nieoczekiwanych skryptów.
- **Przyjrzeć się stronie płatności.** Porównać skrypty, które ładuje strona płatności, z tym, czego się oczekuje. Nieznane domeny stron trzecich to sygnał ostrzegawczy.

PCI DSS i dostawca płatności

Jeśli firma podlega zakresowi PCI DSS, mieści się to w istniejących obowiązkach zarządzania podatnościami. Wymaganie 6.3.1 nakłada obowiązek identyfikowania nowych podatności i przypisywania im klasyfikacji ryzyka. Wymaganie 6.3.3 nakłada obowiązek instalowania krytycznych lub istotnych dla bezpieczeństwa łatek, zgodnie z tą klasyfikacją, w ciągu miesiąca od wydania.

To maksimum, a nie zachęta do czekania. Wobec powagi PHP2Shell i doniesień o aktywności exploitów dotknięte systemy dostępne z internetu należy naprawić natychmiast. Gdy strona płatności jest objęta zakresem, istotne mogą być także wymagania dotyczące skryptów na stronach płatności.

Sprzedawca, który nie potrafi wykazać, że załatał szybko, i nie ma zapisu powodów, jest w słabej pozycji, jeśli coś pójdzie źle. Na początek wystarczy krótka pisemna notatka: kiedy lukę zidentyfikowano, co zrobiono i kiedy.

Dostawcę płatności warto poinformować wcześniej, jeśli są powody sądzić, że sklep został naruszony. Dostawcy płatności mogą mieć obowiązki zgłaszania i eskalacji wynikające z obowiązujących zasad organizacji kartowych, a sprzedawca, który zgłasza sprawę pierwszego dnia, jest traktowany zupełnie inaczej niż taki, którego dostawca odkrywa przez alert organizacji kartowej lub serię zgłoszeń oszustw.

Dla dostawców i ISO

Narażenie obejmuje cały portfel i nie jest rozłożone równomiernie. Najpierw warto przyjrzeć się sprzedawcom na PHP po zakończeniu wsparcia, na tanim hostingu współdzielonym i na samodzielnie zarządzanym oprogramowaniu sklepowym. Rozsądne kroki to:

- Skontaktować się ze sprzedawcami na samodzielnie hostowanym oprogramowaniu sklepowym, krótko i wprost, z prośbą o datę zastosowania łatki.
- Obserwować oznaki zhakowanej strony płatności: nagły wzrost testowania kart, nieoczekiwane odrzucenia, zwroty niepasujące do zamówień lub spory zgłaszające oszustwo na kartach używanych wyłącznie u tego sprzedawcy.
- Ponownie uruchomić monitoring stron internetowych dla sprzedawców o największych obrotach i szukać nowych skryptów lub nieznanych miejsc docelowych płatności.
- Pamiętać, że ta sama luka dotyczy własnych aplikacji oraz stron partnerów i afiliantów.

Przy onboardingu przydatne jest pytanie, jakiej platformy i jakiego hostingu używa sprzedawca oraz kto odpowiada za ich aktualność. Odpowiedź dużo mówi o tym, jak konto zachowa się w takim tygodniu jak ten.

Źródła

- [Tenable: PHP2Shell \(CVE-2026-109618, CVE-2026-109619\) frequently asked questions](#)
- [Qualys: PHP announces critical security release for two vulnerabilities](#)
- [Cisco Talos: Two critical RCEs in PHP core](#)
- [CERT-EU: Critical vulnerabilities in PHP \(2026-018\)](#)
- [watchTowr: PHP remote code execution exploited](#)

Czytaj online: <https://payments-marketplace.com/pl-PL/insights/php2shell-payments/>

Payments Marketplace Ltd

Registered in England & Wales No. 17424288

Garden Ctg, Dereham Road, Dereham, Norfolk, NR19 2PW, UK

connect@payments-marketplace.com · +44 1366 727140