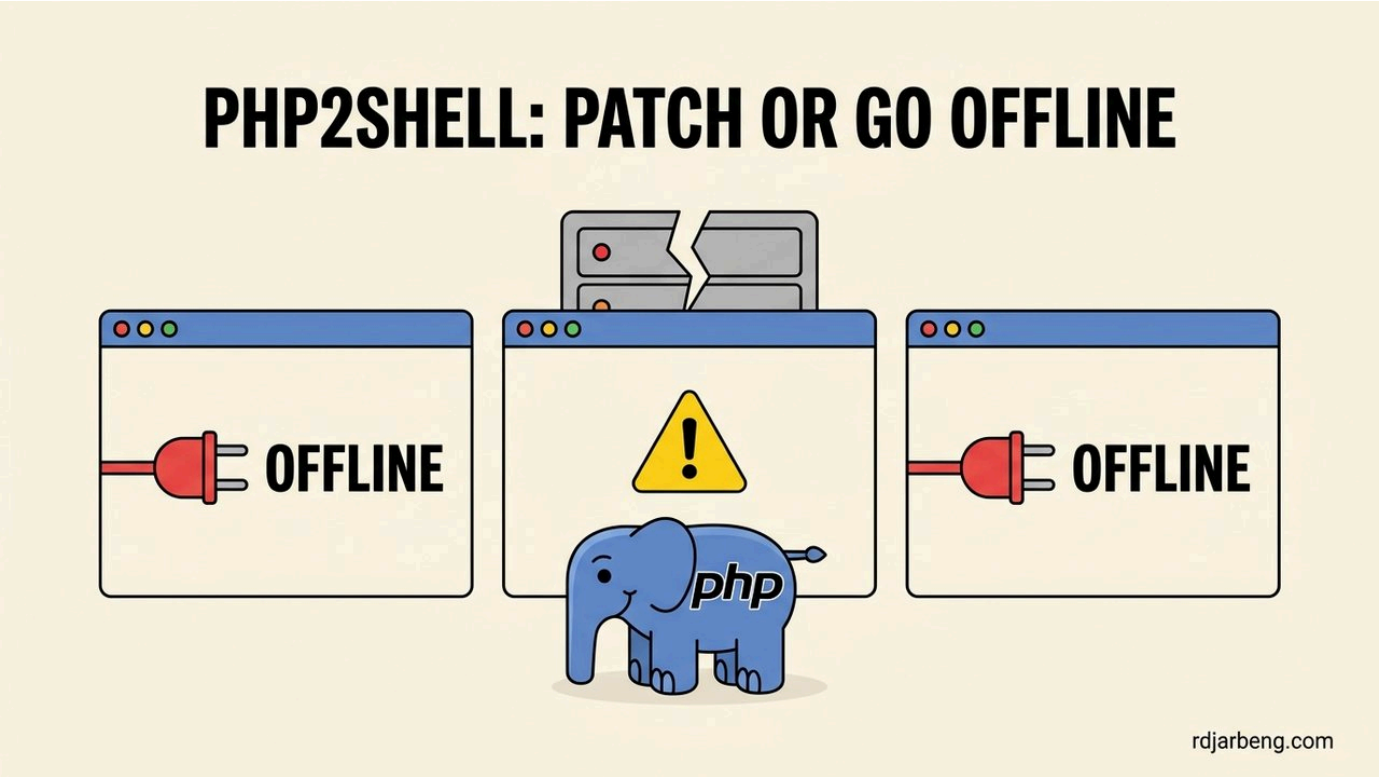
PHP2Shell: The Two PHP Bugs That Took Government Websites Offline, and What to Do

Filed under **Security** Published: October 22, 2026 at 2:55 PM Author: **Richard Djarbeng** 18 min read

KEY POINTS

- Two PHP bugs, CVE-2026-109618 and CVE-2026-109619 (php2shell), let an attacker run code on a web server without logging in. Both are rated 9.2 out of 10.
- The fix is PHP 8.5.12, 8.4.27, 8.3.36 or 8.2.35. Updating your website or its plugins does not fix it; PHP itself has to be updated and the workers restarted.
- Government portals, banks, airlines, a port, hospitals and the BBC took services offline as a precaution while they patched.
- The same day Node.js, GitLab, LiteLLM, Railway and Plex had their own emergencies. Node.js users should update to 26.12.1, 24.22.1 or 22.23.4.

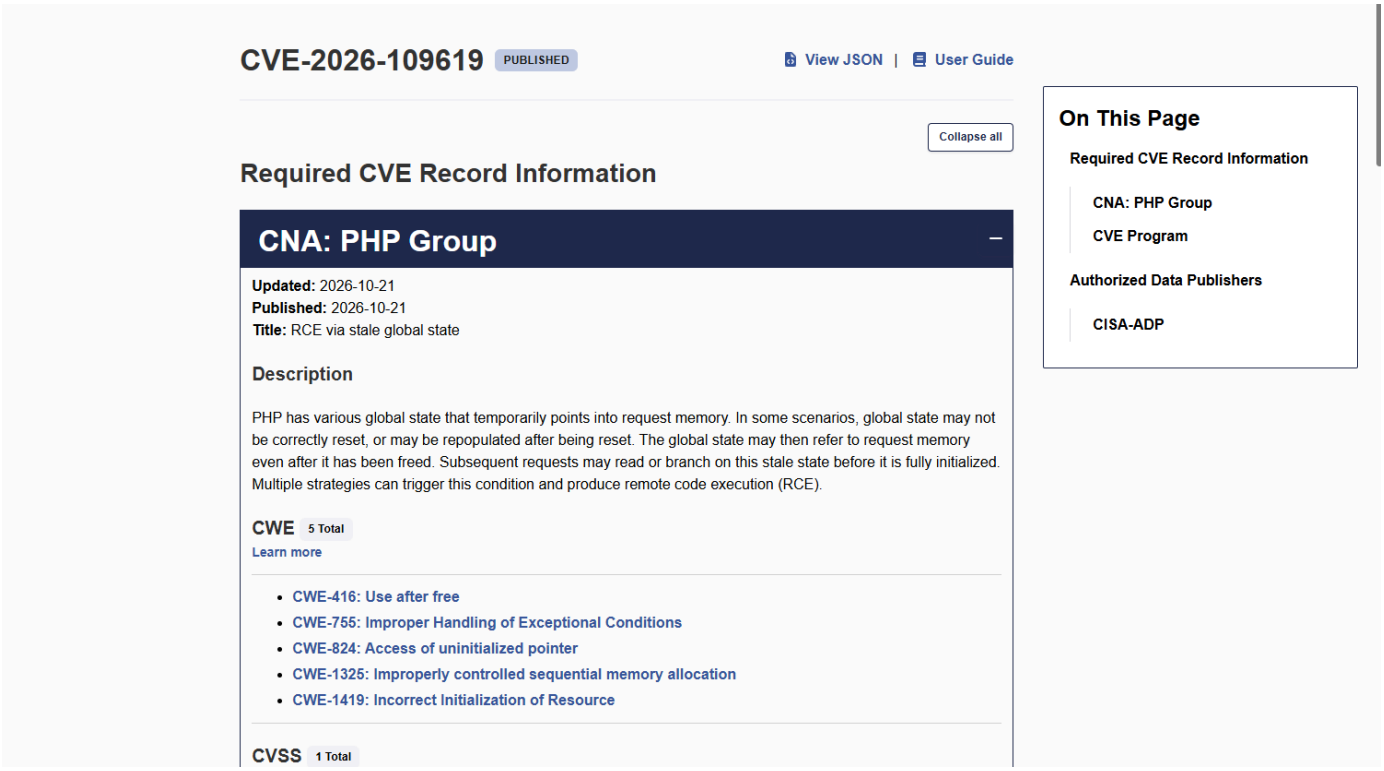
On Wednesday 21 October 2026, PHP released emergency fixes for two bugs that let an attacker take over a web server without logging in. PHP runs a large part of the web, including every WordPress, Drupal, Joomla, Magento and Laravel site, so the response was unusual. The NSA, CISA and CERT-EU told organisations that could not patch straight away to take their sites offline, and governments, banks, airlines and broadcasters did exactly that.



The bugs became known as **php2shell**. Here is what they are, who was affected, and what to do if you run a PHP site or pay someone to host one.

What the two bugs are

The PHP Group published two CVE records on 21 October, CVE-2026-109618 ("RCE via OOM corruption") and CVE-2026-109619 ("RCE via stale global state"). This is the second one:



PHP rates both 9.2 out of 10 (CVSS v4.0). In plain terms, according to [Rapid7's analysis]:

- **CVE-2026-109618: running out of memory mid-request.** When PHP runs out of memory, it can stop processing a request halfway through and then tidy up data that was never fully set up. An attacker can trigger this with specially crafted form or file uploads, even ones that look within the size limits.
- **CVE-2026-109619: leftovers from the previous request.** A PHP worker handles one request after another. Some internal state can survive after its memory has been freed, and the next request on the same worker can use it before it is reset. Rapid7 confirmed two ways in: the session upload progress feature (switched on by default) and, on PHP 8.5, the fatal error backtrace feature.

Either one can end in remote code execution: the attacker runs their own commands with the same permissions as the web server. Every common way of running PHP for the web is affected, including PHP-FPM, Apache mod_php, php-cgi, FrankenPHP and LiteSpeed, and almost every PHP version back to 5.4. [Huntress reports] that researchers at Wiz demonstrated unauthenticated takeovers of WordPress, Drupal and Joomla, and a memory leak path that could expose passwords, API keys and certificates.

Why the CVE page says so little

If you open either CVE record you get one paragraph and a list of weakness types, with no exploit and no fix instructions. That is by design. A CVE record is an identifier, a shared name so that PHP, hosting companies, governments and security tools all know they are talking about the same flaw. The details live in the vendor's advisory and in write-ups from security firms like the ones linked throughout this post.

How the week unfolded

The order of events, from the Huntress timeline and Help Net Security:

- **August 2026:** Ryan Austin of the security firm Calif found one of the bugs with an AI model as part of OpenAI's "Patch the Planet" project. Researchers at Wiz found them independently (Wikipedia summary). OpenAI later called the flaw "one of the most impactful vulnerabilities in recent memory" (The National).
- **17 October:** Someone tried to exploit the bugs against one of Amazon's honeypots, four days before the fix was public. Amazon only connected the attempt to php2shell after the disclosure, which is why it now recommends checking server logs back to 17 October.
- **20 October, morning:** A post on Hacker News said two critical PHP bugs were about to be disclosed. It was taken down, but screenshots spread, and the security firm watchTowr confirmed the bugs were real. PHP brought its release forward by a day.
- **21 October:** PHP released 8.5.12, 8.4.27, 8.3.36 and 8.2.35 and published the advisories. CISA issued an alert the same day.
- **21 and 22 October:** Organisations around the world took sites offline while they patched (the full list is below), and attack attempts climbed into the millions.

Why governments said "take it offline"

Telling people to switch off their own websites is rare. It happened here because the bugs can be reached before any login, affect almost every PHP version in use, and were public before most people could patch.

The NSA's first post put it bluntly:

> *Effective Immediately, all organizations using PHP for their systems are advised to be prepared for an update addressing two critical vulnerabilities. … If an immediate update is not feasible, it is recommend that organizations take the system offline until it can be fully mitigated.*
>
> — NSA Cyber (@NSACyber) **October 21, 2026**

The next day it followed up with detailed guidance:

> *UPDATE: Specific mitigation guidance for the recently identified PHP critical vulnerabilities is now available. Secure your servers today.*
>
> — NSA Cyber (@NSACyber) **October 22, 2026**

CISA gave the same advice for anyone who could not update, or who runs PHP 8.1 or older. CERT-EU advised taking sites offline while patches were pending, and the UK's National Cyber Security Centre warned organisations to look for backdoors.

The most unusual warning came from Japan. Just before 10 pm Tokyo time on 21 October, Prime Minister Sanae Takaichi posted on X asking everyone to act that night: check whether their public websites use PHP, get the fix ready, and treat temporarily stopping a site as a valid way to reduce the risk. She took her own website offline too, as did Kobayashi, the policy chief of her Liberal Democratic Party (Jiji, Nikkei, both in Japanese).

> *皆様に、今夜に対応して頂きたい大至急のお願いです。多くのウェブサイトで使用されているPHPに関し、PHPのセキュリティチームが、重大な２件の脆弱性に関する注意喚起を行っています。…*
>
> — 高市早苗 (@takaichi_sanae) **October 21, 2026**

English translation of the opening: "An urgent request that I would like everyone to act on tonight. PHP's security team has issued a warning about two serious vulnerabilities in PHP, which is used by many websites."

A day later she followed up for the businesses that still had their sites switched off:

> *今回のPHPの脆弱性対応のため、現在もウェブサイトを一時停止されている事業者などの皆様もいらっしゃると思います。…*
>
> — 高市早苗 (@takaichi_sanae) **October 22, 2026**

English translation of the opening: "I imagine some businesses and others still have their websites temporarily suspended while they deal with this PHP vulnerability."

Who went offline

These are the organisations reported to have taken websites or services down as a precaution. Most come from a Reuters factbox; others are linked individually. None of them reported a breach. They switched off to patch safely.

- **Netherlands:** the government portal Overheid.nl, the Port of Rotterdam, the national football association KNVB, the websites of the five largest political parties and many hospital websites, following the National Cyber Security Centre's lead.
- **Finland:** Valtori, the state IT centre, shut down the government's public web services. It said it had seen no attempts to exploit the bugs.
- **France:** the Interior Ministry disabled the websites of the prefectures (local government offices), and the telecoms company Orange suspended some websites, services and apps across several subsidiaries.
- **Germany:** ITZBund, the federal IT provider, shut down several sites as a preventive step.
- **Belgium:** the foreign ministry took its websites offline.
- **Czech Republic:** the government took its website offline at 7 pm on 21 October on the advice of the national cyber watchdog.
- **Denmark:** Aula, the platform Danish schools use to talk to parents, went offline on Wednesday afternoon and came back on Thursday morning (The Copenhagen Post).
- **United Kingdom:** the BBC took some web portals down, as did several councils, including the City of Edinburgh.
- **Japan:** the Ministry of Finance, the Meteorological Agency and the National Tax Agency suspended some online services, and so did Nomura, Mitsubishi UFJ Financial Group, the airline ANA and East Japan Railway.
- **The wider web:** the Wikimedia Foundation took many of its supporting services offline, saying all its PHP systems were affected, and the Internet Archive went down for emergency maintenance and patching before coming back online.

Who this affects

People who run websites. If your site, or any tool it relies on, runs on PHP, it is exposed until PHP itself is updated. That includes WordPress and other content management systems, online shops, forums, booking systems and the admin panels that hosting companies use. A data company, Veridion, matched public version information to

businesses and counted about 7 million companies running an affected version, most with fewer than 50 employees (Wikipedia summary). Small businesses are the ones least likely to have someone watching for this.

People who use websites. For everyone else, the main effect this week was services being unavailable: tax portals, train and airline bookings, banking and school platforms. The bigger worry is a site that stays unpatched. An attacker who takes over a server can read what it stores, including logins, personal details and payment data, and can use it to attack its visitors. There is nothing an ordinary user can patch here, but it is a good week to make sure your passwords are not reused across sites.

How fast attackers moved

Cloudflare sits in front of a large share of the web, so it sees attacks early. Its CEO, Matthew Prince, shared these numbers on 22 October:

> *Some stats on #php2shell. In the 10 hours before it was publicly announced, across the @Cloudflare network we saw thousands of attempts to exploit it. That jumped to tens of thousands in the 45 minutes post announcement, hundreds of thousands within 2 hours, and millions now.*
>
> *— Matthew Prince (@eastdakota) October 22, 2026*

These are attempts, not confirmed break-ins. On 21 October CISA said it had not yet seen successful exploitation, and Germany's BSI and the Dutch authorities reported the same (The Next Web). Amazon's security team, on the other hand, says the bugs are now being exploited broadly (Cybernews). Cloudflare added firewall rules for all its customers, including free accounts, but describes them as a stopgap: they only help traffic that passes through Cloudflare, and attackers have already been changing their payloads to get around them.

What to do

If your site is on shared or managed hosting, the PHP version is your host's responsibility. Updating WordPress, plugins or your own code does not fix this. Check your host's status page or ask them directly whether PHP has been updated to one of the versions below. Most large hosts have published notices, including Plesk, GoDaddy, DigitalOcean and Laravel Forge.

If you run your own server:

1. **Update PHP** to a fixed version and **restart every PHP worker**. The old code keeps running in memory until you do.

PHP branch	Update to
8.5	8.5.12
8.4	8.4.27
8.3	8.3.36
8.2	8.2.35
8.1, 8.0, 7.4	Source patches only

   PHP 8.1 and older are out of support, so there is no official release for them, only source patches. Use your Linux distribution's backported package or move to a supported version. Distributions also use their own version numbers, so check their security notices rather than comparing numbers directly. Debian 12, for example, ships the fix as `8.2.35-1~deb12u1`.

2. **Check the version your web server actually uses.** `php -v` shows the command-line version, which can differ from the one serving your site. Check the FPM service or Apache module, or use a `phpinfo()` page that you delete straight afterwards.

3. **If you cannot update today**, these settings from Rapid7 reduce the risk but don't remove it. Set them at server level (php.ini, the FPM pool or Apache config), not with `ini_set()` or `.user.ini`, which apply too late, and then restart the workers:

   - `session.upload_progress.enabled = 0` blocks the confirmed session path for CVE-2026-109619.
   - `fatal_error_backtraces = 0` blocks the backtrace path on PHP 8.5.
   - Keep `post_max_size` small (8M if you don't need uploads) and block multipart uploads at your proxy if your site never accepts them.

   PHP's own advice is that updating is the only complete fix. If none of this is possible, do what CISA and the NSA said: take the site offline or put it behind a VPN until it is patched.

4. **Look for signs of compromise from 17 October onwards.** Watch for PHP worker processes starting shells or downloading tools, new files in the web root, new cron jobs and new user accounts. If you find any, isolate the server and keep the logs before you

clean up.

The rest of the week: other software patched or breached

PHP was not the only problem. On the same day, Node.js, GitLab and several hosting and AI tools had their own emergencies. This viral post summed up the mood, half joking, but X cuts it off after a few lines:

> *So basically, today PHP got two critical bugs that let anyone run code on your server without even logging in, which is bad, and then the details leaked before the patch was even out, which is worse... Anyway, it's Wednesday.*
>
> *— ⅝ (@fiveoutofnine) October 21, 2026*

Here is everything else it lists, checked against official sources.

Node.js: ten fixes, four rated High

Node.js runs JavaScript on servers and in build tools. It released fixes for all supported versions on 21 October: **26.12.1, 24.22.1 and 22.23.4** (Node.js advisory). The four High-rated issues are:

- A bug in the Brotli decompressor (CVE-2026-111497) that allows remote code execution when Node unpacks Brotli-compressed data from an attacker, for example a response fetched from a malicious server.
- Two ways to get around TLS hostname checks (CVE-2026-105483 and CVE-2026-106069), which means an HTTPS connection could trust a server it should have rejected.
- A DNS lookup bug (CVE-2026-105482) where names containing a hidden NUL character are cut short and resolve to the wrong place.

The same releases also remove the compromised eMudhra certificates (see below) from Node's built-in list of trusted certificates.

Who should care: anyone running a Node.js server, API or app, and anyone whose build tools run on Node (most JavaScript projects, including static sites like this one). A static site is far less exposed than a live server, because Node only runs while the site is being built, but update anyway. Node 20 and older are out of support and did not get these fixes, so they will stay vulnerable.

GitLab: a 10.0 bug in self-hosted servers

GitLab fixed CVE-2026-110264, rated the maximum 10.0. Sending a specially crafted regular expression lets an attacker run code on a self-managed GitLab server without logging in. It affects every version since GitLab 7.0 in 2014. The fix is in 19.5.1, 19.4.3 and 19.3.6, with backports to older branches (GitLab's release notes). Users of gitlab.com don't need to do anything; companies that host their own GitLab do.

LiteLLM: a poisoned release

LiteLLM is a popular Python library for calling AI models. Version 1.107.0 was published to PyPI and Docker with a credential stealer inside, and was live for about two hours on 21 October. Installing it was enough to run the malware, which went after cloud keys, SSH keys, browser passwords and AI API tokens. LiteLLM says 1.106.2 is safe and that any machine that installed 1.107.0 should be treated as compromised, with every secret on it rotated (LiteLLM incident report).

Railway and Zeabur: hosting platforms leak their customers' secrets

Railway, a platform for deploying apps, disclosed on 21 October that attackers may have reached customers' environment variables, the place where apps keep passwords and API keys. It paused deployments across the platform while it responded and invalidated every Railway API token. The advice to customers is to treat every secret stored there as exposed and rotate it (GitGuardian's summary).

Zeabur, a smaller deployment platform, is shutting down after a breach that exposed data for about 321,000 accounts and 3.2 million environment variable records. The company says the attackers most likely used credentials taken in an earlier breach in August. Services end on 31 December 2026 (SC Media).

Plex: home media servers being taken over

CISA added four Plex Media Server flaws, nicknamed Plex2Shell, to its list of vulnerabilities that are being actively exploited. Attackers chain them to take over Plex servers that missed the September update, without a Plex login, and then mine cryptocurrency or route other people's traffic through them. The fix is Plex Media Server 1.43.3.10861 or later, but updating doesn't remove an infection that is already there (Guardian MSSP). If you run Plex at home, update it and check for unfamiliar processes.

eMudhra: a private key that was the number 4

eMudhra, an Indian certificate authority whose job is to vouch that websites are who they say they are, had eight intermediate certificates revoked after their private keys turned out to be tiny numbers. One was literally 4, which means anyone could have recreated it. Mozilla, Google, Apple and Microsoft have blocked the certificates, and Node.js removed them too (BleepingComputer). Browsers update this automatically; older devices and software with their own certificate lists may need an update.

Conclusion

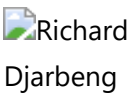
Three things stand out from this week.

The gap between a fix and an attack is now hours. The PHP details leaked a day early, and Cloudflare saw exploit attempts climb from thousands to millions within a day of the announcement. An update that waits for the weekend may already be too late, which is why the advice from the NSA, CISA and Japan's Prime Minister was to patch that night or take the site offline.

Most people don't control the software they depend on. If your site is on shared hosting, PHP is your host's job, and updating WordPress or its plugins does nothing for these bugs. The same goes for secrets stored on platforms like Railway, or the certificates your browser trusts. So the useful question is who runs each layer, and whether they have patched it.

AI is now finding these bugs. Calif found php2shell with an OpenAI model, and Wiz found it independently with its own tools. That should mean more serious flaws are found and fixed before attackers find them, but it also means more of them arriving at once, as this week showed.

Aside on this site: This site is built with Eleventy and served by GitHub Pages, so it doesn't run PHP at all. Node.js only runs while it is being built, and the build has been moved to a patched version. If you run a PHP site, check your PHP version today, or ask your host to confirm theirs.



Written by Richard Djarbeng

Software and Machine Learning Engineer

Richard is a software and machine learning engineer with an MSc in Electrical and Computer Engineering from Carnegie Mellon University. He writes about machine learning, computer vision, IoT, satellite and climate data, and life in Kigali.

About · GitHub · LinkedIn · YouTube · Editorial policy · Report an error

CITATION (IEEE)

[1] R. Djarbeng, "PHP2Shell: The Two PHP Bugs That Took Government Websites Offline, and What to Do," *rdjarbeng.com*, Oct. 22, 2026. [Online]. Available: https://rdjarbeng.com/php2shell-php-vulnerabilities-week-of-october-21/

▶ BibTeX format

RELATED CONCEPTS & STANDARDS

CVE

`Security`

Common Vulnerabilities and Exposures: a public catalogue that gives each known security flaw a unique ID, so everyone refers to the same bug by the same name.

About CVE →

Remote code execution (RCE)

`Security`

A flaw that lets an attacker run their own commands on someone else's server over the network. Unauthenticated RCE means they don't need an account or password first.

CVSS

`Security`

The Common Vulnerability Scoring System rates how severe a flaw is from 0 to 10. 9.0 and above is critical.

CVSS specification →

RECOMMENDED READING

Why My AI Agent Keeps Forgetting Things: Tokens, Context Windows and Compaction

Nov. 25, 2026

Why Flight Tickets in Africa Are So Expensive, From First Principles

Nov. 8, 2026

What can you do? Testing AI Agents, Gemini Argon and Claude

Oct. 14, 2026