

Critical PHP flaws allow unauthenticated remote code execution: update now



The UK NCSC urges immediate action on two critical PHP flaws, CVE-2026-109618 and CVE-2026-109619, that could allow unauthenticated remote code execution. PHP has released a fix.

- Two critical PHP flaws, CVE-2026-109618 and CVE-2026-109619, may allow attackers to run commands on a web server running PHP.
- An out-of-memory condition can cause memory corruption, which under certain conditions can be exploited for unauthenticated remote code execution.
- PHP version 5 onwards is affected; older versions should move to a fully supported release.
- Products and appliances that bundle PHP may also be exposed.
- Temporary mitigations exist; if they cannot be applied, the NCSC advises considering taking the

system offline.

The UK National Cyber Security Centre (NCSC) has encouraged organisations to act immediately on a critical security issue in PHP, the widely used server-side scripting language that generates dynamic web content, talks to databases and runs backend application logic. PHP is also embedded in a range of other products.

What the flaws are

An attacker could trigger an out-of-memory condition in PHP that leads to memory corruption. Under certain conditions this can be exploited to achieve unauthenticated remote code execution. PHP has fixed the issue.

The two identifiers are:

- **CVE-2026-109618**: remote code execution through an out-of-memory condition in RFC 1867 input parsing.
- **CVE-2026-109619**: remote code execution through global state corruption after a fatal error.

Organisations running PHP version 5 onwards are affected. Those on older versions are urged to move to a fully supported version. The NCSC says it will keep monitoring for impact on UK organisations.

Recommended response

The NCSC advises identifying all PHP installations, starting with internet-facing systems, and not limiting the search to outdated versions. Organisations should update to the latest fixed version, apply the hardening steps from the PHP Team, and watch for vendor updates for appliances and services that bundle PHP. NCSC-NL has published an inventory of software products with a PHP dependency that may be affected.

Where updating is not possible, the PHP Team has provided temporary mitigations. If these cannot be applied quickly, the NCSC suggests considering taking the system offline. Systems that cannot be updated or disabled immediately should be checked for signs of compromise and for backdoors before any upgrade or mitigation.

What to do now

1. Inventory all PHP installations, starting with internet-facing systems, regardless of version.
2. Update PHP to the latest fixed version and apply the PHP Team's hardening steps.
3. Ask vendors of appliances and applications that bundle PHP for fixed releases, and check the NCSC-NL inventory of affected products.
4. Where patching is delayed, apply the PHP Team's temporary mitigations, or take the system offline if that is not feasible.
5. Review unpatched systems for signs of compromise and backdoors, with incident response support if needed.
6. Move systems on unsupported PHP versions to a supported release and record the work under your

vulnerability management process.