CISA adds PHP2Shell vulnerabilities to its list of actively exploited flaws



CISA has added the PHP2Shell vulnerabilities to its catalog of actively exploited flaws and set a patching deadline for U.S. federal agencies. The source text available is limited.

- CISA has added the PHP2Shell vulnerabilities to its catalog of actively exploited security flaws.
- The agency confirms the flaws are now being used in attacks.
- U.S. federal agencies were ordered to patch their systems by Thursday.
- The available source text gives no technical detail, CVE identifiers or affected versions.

What happened

The U.S. Cybersecurity and Infrastructure Security Agency (CISA) has added the PHP2Shell vulnerabilities to its

catalog of actively exploited security flaws. Inclusion in the catalog means the agency has confirmed the flaws are being used in real attacks.

CISA also ordered U.S. federal agencies to patch their systems by Thursday.

What we do not know yet

The portion of the report available to us is brief. It does not name the affected products or versions, does not list CVE identifiers, and does not describe the attacks or who is behind them. We have therefore not included any such details here.

What this means for Saudi institutions

The federal deadline applies only to U.S. agencies, but active exploitation is relevant to any organisation that may run affected software. Given the name, teams that operate PHP-based applications should treat this as a prompt to check their exposure, while waiting for the vendor and CISA details to confirm what is actually affected.

A calm, ordered response is enough at this stage: confirm the facts from the primary advisory, identify what you run, and prioritise patching of internet-facing systems once affected versions are known.

What to do now

1. Read CISA's catalog entry and the vendor advisory to confirm affected products and versions.
2. Inventory PHP-based applications and components, especially internet-facing ones.
3. Apply available patches to confirmed affected systems, starting with exposed assets.
4. Review web server and application logs for unusual activity once indicators are published.
5. Brief your security operations team and record the review in your vulnerability management process.