

PHP2Shell Checklist for PHP Development Teams

The bug is in PHP, not in PrestaShop, Magento or Laravel. Your host's PHP update is the fix, and a framework update changes nothing. Tick these off once the host has patched. Replace SITE_ROOT with the site's document root.

1. Confirm the fix is really running

- ☐ The web server runs PHP 8.2.35, 8.3.36, 8.4.27, 8.5.12 or later.
- ☐ On PHP 8.1 or older, the cPanel package changelog shows the fix (change php81 to your version).
`rpm -q --changelog ea-php81-php-cli | grep -m1 CVE-2026-109618`
- ☐ The version and SAPI (the way PHP connects to the web server) were checked through the browser, not with `php -v`. A result of `apache2handler` means `mod_php`, which is the highest-risk setup.
`<?php echo PHP_VERSION . ' ' . PHP_SAPI; (delete this file right after)`
- ☐ Every Docker image was rebuilt on a PHP base released after October 21, 2026.
`docker build --pull --no-cache -t IMAGE_NAME .`

2. Clean up what only your team can fix

- ☐ Stale or deprecated `php_value` and `php_flag` lines in `.htaccess` files were removed or moved to `php.ini`.
`grep -rn --include=.htaccess -E "php_(value|flag|admin_value|admin_flag)" SITE_ROOT`
- ☐ The nginx configuration passes no `PHP_VALUE` or `PHP_ADMIN_VALUE` that raises a warning.
`grep -rn -E "PHP_VALUE|PHP_ADMIN_VALUE" /etc/nginx/`
- ☐ Session upload progress is off, and every PHP worker was restarted.
`session.upload_progress.enabled = 0`
- ☐ The most frequent fatal errors in upload and import paths were fixed. A timeout counts as a fatal error.
`grep -oE "PHP Fatal error: .{0,120}" SITE_ROOT/error_log | sort | uniq -c | sort -rn | head -30`
- ☐ The settings `post_max_size`, `max_input_vars`, `max_input_nesting_level` and `max_file_uploads` are conservative.
- ☐ Routes that never accept uploads (JSON APIs, for example) refuse multipart/form-data at the web server.

3. Check for signs of compromise

- ☐ PHP files changed since October 20 match your deployment history.
`find SITE_ROOT -name "*.php" -type f -newermt "2026-10-20"`
- ☐ The access logs were searched for the public scanner's User-Agent. A hit means a scan, not a break-in.
`grep -i "php2shell-" /var/log/apache2/access.log`
- ☐ The admin user list in the store or app holds no accounts that nobody can explain.

Host not patched yet? These are temporary stopgaps, not fixes. Set `session.upload_progress.enabled = 0` and, on PHP 8.5 only, `fatal_error_backtraces = 0`. PHP also lists `memory_limit = -1`, but it lets one request use all of the server's memory, so use it for a day or two at most. Restart every PHP worker after any change.

A clean result on every line is reassuring, but it is not proof. PHP's advisories say updating PHP is the only complete fix.