# WordPress Website Maintenance *Checklist*

To keep a WordPress site secure, stable and fast in late 2026, three things have to happen every time.

01

Security patches go on within hours, not weeks.

02

PHP and the database stay on supported versions (PHP 8.2 reaches end of life on 31 December 2026).

03

Backups are stored off-site and you have tested restoring them.

The rest of this checklist supports those three jobs.

| 8 | 10 | 16 | 8 | 7 |
|---|---|---|---|---|
| **Daily** | **Weekly** | **Monthly** | **Quarterly** | **Yearly** |
| ≈10–15 min | ≈45–90 min | ≈2–4 hours | ≈half day | ≈1–2 days |

PREPARED BY
Syed Muhammad Hussain

FACTS LAST CHECKED
7 November 2026

# About this checklist *(read first)*

| | |
|---|---|
| HOW IT WAS MADE | Built from AI-assisted research, then reviewed and adapted by Syed Muhammad Hussain for a law firm's WordPress site running on Windows Server and IIS. |
| FACTS LAST CHECKED | 7 November 2026. Versions, dates and limits change; re-check php.net (supported versions), WordPress's Hosting Handbook (server compatibility) and make.wordpress.org before relying on them. |
| COMMANDS ASSUME LINUX | The tools column uses Linux commands (`tail -f`, `df -h`, `/var/log/nginx/`, `find -perm`, `htop`, chmod-style permissions). On Windows/IIS, use IIS logs (default `%SystemDrive%\inetpub\logs\LogFiles`), NTFS permissions for the application pool identity, `web.config` instead of `.htaccess`, Windows Task Scheduler instead of cron, and `Get-PSDrive` for disk space. The accompanying post has the full IIS table. |
| CORRECTIONS MADE ON 7 NOVEMBER 2026 | • Patchstack's 5-hour figure is the weighted median time to the *first exploit attempt* for the most heavily exploited vulnerabilities, not the start of mass attacks.<br>• WordPress.org's plugin release delay was announced on 5 June 2026 at up to 24 hours for auto-updates; by September 2026 it was 6 hours and also covered dashboard updates.<br>• Removed two figures without a primary source (a malware-share percentage and a mobile-traffic share). |
| PRIMARY SOURCES FOR KEY DATES | PHP supported versions (php.net); CISA KEV bulletin, 26 Oct 2026 (PHP2Shell); Oracle MySQL blog (MySQL 8.0 end of life, April 2026); CA/Browser Forum Ballot SC-081v3 (200-day limit from 15 March 2026, 47 days by March 2029); Let's Encrypt "Upcoming features"; make.wordpress.org/core/7-2 (WordPress 7.2 schedule). |
| NOT ADVICE FOR YOUR SETUP | Frequencies suit a typical business site. Test changes on staging first. |

# Contents


01 About this checklist 2
02 TL;DR 3
03 Key findings: 2026 reference baselines 4
04 How to use this checklist 6
05 Stop-everything: critical issues 7
06 Daily checklist 8
07 Weekly checklist 10
08 Monthly checklist 12
09 Quarterly checklist 15
10 Yearly checklist 17
11 Section coverage map 19
12 WP-CLI quick reference 19
13 Maintenance and issues log templates 21
14 Recommendations 22
15 Caveats 22

OVERVIEW

# TL;DR

### How fast you patch now matters more than anything else.

Patchstack's *State of WordPress Security in 2026* counted 11,334 new WordPress-ecosystem vulnerabilities in 2025, up 42%. Of those, 91% were in plugins. For the most heavily exploited bugs, the weighted median time to the first exploit attempt was 5 hours after disclosure, and about half of high-impact vulnerabilities were exploited within 24 hours. That puts vulnerability alerts and security updates in the daily schedule, not the monthly one.

| 11,334 | 91% | 5 h |
|---|---|---|
| new vulnerabilities in 2025 (up 42%) | in plugins | weighted median to first exploit attempt |

### Version requirements and web standards changed in 2026:

- WordPress 7.0 and later need at least PHP 7.4. WordPress.org recommends PHP 8.3+ with MariaDB 10.11+ or MySQL 8.0+.
- WordPress's Hosting Team recommends PHP 8.4/8.5 and MySQL 8.4 for new 7.1 installs.
- MySQL 8.0 reached end of life on 30 April 2026.
- Public SSL/TLS certificates have lasted at most 200 days since 15 March 2026.
- Google's Core Web Vitals "good" thresholds are unchanged: LCP ≤ 2.5 s, INP ≤ 200 ms, CLS ≤ 0.1, measured at the 75th percentile of real visits.

### The schedule below is ready to use.

It has 8 daily, 10 weekly, 16 monthly, 8 quarterly and 7 yearly tasks. Each task has a checkbox, priority, reason, suggested tools, status and notes. A "Stop-Everything" table lists critical issues, and log templates are at the end.

REFERENCE

# Key findings: 2026 reference baselines

Check against these values when you work through the compatibility, security and performance tasks.

| AREA | CURRENT VALUE (AS OF 6 NOV 2026) | WHAT IT MEANS FOR MAINTENANCE |
|---|---|---|
| **Current WordPress** | 7.1.4 (security release, 28 Oct 2026); 7.1.3 (6 Oct 2026) shipped 7 security fixes | Anything below 7.1.4 on the 7.1 branch is missing security fixes; WordPress.org says to update immediately. Fixes were backported to branches back to 4.7. |
| **Next major release** | WordPress 7.2 scheduled for 8 Dec 2026 (RC1 due 17 Nov) | Test on staging during RC; e-commerce sites should consider holding the major upgrade until after peak holiday trading. |
| **WordPress minimum PHP** | PHP 7.4 (since WordPress 7.0, released 20 May 2026); PHP 7.2/7.3 dropped | Sites on PHP 7.3 or lower are stuck on the 6.9 branch and cannot receive 7.x. |
| **WordPress.org recommendation** | PHP 8.3+; MariaDB 10.11+ or MySQL 8.0+; HTTPS | Public baseline. Note the conflict: MySQL 8.0 itself is end-of-life (below). |
| **Hosting Team recommendation (WP 7.1, new installs)** | PHP 8.4.x/8.5.x; MySQL 8.4.x; MariaDB 10.11.x/11.4.x/11.8.x | The stricter, more current target. “A version that is end-of-life upstream is never recommended.” |
| **PHP support** | 8.2 EOL 31 Dec 2026; 8.3 EOL 31 Dec 2027; 8.4 EOL 31 Dec 2028; 8.5 EOL 31 Dec 2029; 8.1 and older already EOL | Plan any 8.2 → 8.4/8.5 migration now. |
| **PHP2Shell (CVE-2026-109618/-109619)** | Critical flaw in PHP itself that lets attackers run code without logging in; fixed 21 Oct 2026 in 8.5.12, 8.4.27, 8.3.36, 8.2.35; added to CISA’s list of known exploited vulnerabilities on 26 Oct 2026 | No WordPress plugin or core update can fix this — verify the PHP point release with your host. |
| **Databases** | MySQL 8.0 EOL 30 Apr 2026 (8.4 LTS supported to Apr 2032); MariaDB 10.6 EOL 6 Jul 2026 | Treat MySQL 8.0 and MariaDB 10.6 as upgrade items. |
| **Core Web Vitals** | Good: LCP ≤ 2.5 s, INP ≤ 200 ms, CLS ≤ 0.1; Poor: > 4 s, > 500 ms, > 0.25; measured at p75 of real users. INP replaced FID on 12 Mar 2024 | Judge on field data (CrUX/Search Console), not just Lighthouse lab scores. |
| **TLS certificate lifetime (CA/Browser Forum rule SC-081v3)** | Max 200 days since 15 Mar 2026; 100 days from 15 Mar 2027; 47 days from 15 Mar 2029 (domain validation can be reused for only 10 days) | Renewing by hand once a year no longer works. Automate renewal with ACME and monitor expiry with a separate tool. |
| **Let’s Encrypt** | Expiry reminder emails ended 4 Jun 2025; default lifetime moves to 64 days (10 Feb 2027) and 45 days (16 Feb 2028) | You must run your own expiry monitoring; don’t hardcode “renew every 60 days.” |

| AREA | CURRENT VALUE (AS OF 6 NOV 2026) | WHAT IT MEANS FOR MAINTENANCE |
|---|---|---|
| **File permissions (WordPress Hardening guide)** | Directories 755, files 644, wp-config.php 400 or 440 (host-dependent; 600/640 also common) | Never 777. Add `DISALLOW_FILE_EDIT`. |
| **Backups** | CISA: 3-2-1 (3 copies, 2 media types, 1 off-site); #StopRansomware guide: offline, encrypted backups with regularly tested integrity | Modern target is 3-2-1-1-0: one immutable/offline copy, zero restore errors. |
| **Vulnerability alert sources** | Patchstack database; Wordfence Intelligence (API free for personal and commercial use); WPScan (free API: 25 requests/day) | Point at least one alert feed at each site's exact list of plugins and themes. |
| **WordPress.org plugin cooldown** | Announced 5 Jun 2026 (up to 24 hours, for auto-updates). By Sep 2026 the wait was 6 hours and also covers dashboard updates. Since 9 Sep 2026, releases that an automated review (AI models plus Jetpack Scan) rates high-risk are blocked. | This helps against attackers slipping malicious code into plugin updates, but it does not replace your own checks. It also does not cover premium plugins sold outside WordPress.org. |
| **Email deliverability** | Gmail, Yahoo, Microsoft: senders of 5,000+ emails a day need SPF, DKIM and DMARC (at least p=none), with the From domain matching; spam complaint rate under 0.3% | Form notifications and WooCommerce emails fail without warning if they are not authenticated. |

> **What this means:** WordPress core itself is rarely the weak point. Patchstack recorded only 6 core vulnerabilities in 2025, all low priority. Even so, core shipped several security releases in autumn 2026. The October 2026 PHP2Shell flaw also showed that a fully updated WordPress site can be taken over through the PHP software it runs on. Maintenance therefore has to cover three layers: **code (core, plugins, themes), the server software (PHP, database, web server), and recovery (backups)**.

DETAILS

# How to use this checklist

01 **Copy the schedule tables** into your project tool, spreadsheet, or client report. Each task has an ID (D = Daily, W = Weekly, M = Monthly, Q = Quarterly, Y = Yearly).

02 **Tick the checkbox** (☐ → ☑) once the task is done for the current cycle.

03 **Update Status** to one of: **Pending**, **In Progress**, **Completed**. Reset to Pending at the start of each cycle.

04 **Record findings** in the Notes column and log anything that needs follow-up in the Issues Log at the end.

05 **Priority key:**

| | |
|---|---|
| ● CRITICAL | Failure means a hacked site, lost data, an outage or lost revenue. Act the same day. |
| ● HIGH | Significant risk or degradation if missed. Act within the cycle. |
| ○ MEDIUM | Performance, SEO or housekeeping. Schedule within 1–2 cycles. |
| ○ LOW | Nice-to-have improvements. |

06 **Always:** take a fresh backup before updates or database cleanup, and test major changes on staging first.

ESCALATE

# Stop-everything: critical issues requiring immediate attention

If any of these are found during any check, pause routine work and escalate.

| | TRIGGER | IMMEDIATE ACTION | TARGET RESPONSE |
|---|---|---|---|
| ☐ | A vulnerability alert (Patchstack/Wordfence/WPScan) rates an installed plugin or theme high/critical, or marks it as actively exploited | Update immediately. If no fix exists yet, deactivate or remove the plugin, or block the attack with a firewall (WAF) rule | Within hours (heavily exploited bugs saw a first exploit attempt a weighted median of about 5 hours after disclosure) |
| ☐ | WordPress core security release (e.g., 7.1.4) not applied | Update core now; confirm auto-updates for minor releases are working | Same day |
| ☐ | PHP point release below 8.5.12 / 8.4.27 / 8.3.36 / 8.2.35, or PHP ≤ 8.1 | Get written confirmation from your host that the PHP2Shell fix is installed and PHP has been restarted; schedule an upgrade to a newer PHP version | Same day |
| ☐ | Site down, white screen, or HTTP 500s on key pages | Check PHP fatal error log, roll back last change, restore from backup if needed | Immediately |
| ☐ | Unknown administrator account, unexpected role change, or unexplained new Application Password | Remove/disable account, rotate all admin passwords and salts, run full malware scan, review logs | Immediately |
| ☐ | Malware found, changed core files (`wp core verify-checksums` fails), PHP files in `/uploads/`, or spam content/redirects | Take the site offline or block traffic with the firewall. Clean it or restore a known-good backup, change all passwords, and find how the attacker got in | Immediately |
| ☐ | Last successful backup older than 24 hours (or older than the most data you can afford to lose) | Run a manual backup now, fix the scheduler, and check the off-site copy | Same day |
| ☐ | SSL certificate expiring in < 14 days or renewal failed | Fix automatic renewal (ACME), renew by hand, and confirm the server sends the full certificate chain | Same day |
| ☐ | Domain expiring in < 30 days or auto-renew disabled/failed payment | Renew and enable auto-renew and registrar lock | Same day |
| ☐ | Disk usage > 90% or database partition full | Clear logs/old backups/cache, expand storage | Immediately |
| ☐ | Checkout/payment failure or gateway webhook errors | Test in sandbox, check gateway logs/API keys, notify stakeholders | Immediately |
| ☐ | Search Console “Security issues” or “Manual actions” notice | Clean site, then request review | Same day |

D1–D8 · 8 TASKS

# Daily checklist

≈10–15 minutes; automate where possible

## ☐ D1 Confirm the site is up and review any overnight downtime alerts

● CRITICAL

AREA
Infrastructure

WHY IT MATTERS
Downtime loses revenue and leads and hurts search crawling; alerts often reveal attacks or an overloaded server

SUGGESTED TOOLS
UptimeRobot, Better Stack, Pingdom, Jetpack Monitor

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

## ☐ D2 Verify the last scheduled backup (files + database) completed and reached off-site storage

● CRITICAL

AREA
Backup

WHY IT MATTERS
An unnoticed failed backup is discovered only during a disaster

SUGGESTED TOOLS
Backup plugin dashboard (UpdraftPlus, BlogVault, Jetpack VaultPress Backup), host backups, `wp db export` logs

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

## ☐ D3 Check vulnerability alerts for every installed plugin, theme and core version

● CRITICAL

AREA
Plugins & Themes / Security

WHY IT MATTERS
46% of 2025 vulnerabilities had no fix when they were made public. The most heavily exploited were first attacked a weighted median of 5 hours after disclosure, and about half of high-impact ones within 24 hours

SUGGESTED TOOLS
Patchstack, Wordfence Intelligence (free API/webhooks), WPScan (`wpscan --url … --api-token …`), managed dashboards (MainWP, ManageWP, WP Umbrella)

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

## ☐ D4 Apply security releases (core minor releases, plugin/theme security fixes) — backup first, verify site after

● CRITICAL

AREA
Core / Plugins

WHY IT MATTERS
A security release makes the flaw public, so every hour of delay leaves the site exposed

SUGGESTED TOOLS
Dashboard → Updates, `wp core update --minor`, `wp plugin update <slug>`

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

## ☐ D5 Scan the PHP error log for new fatal errors

● HIGH

AREA
Logs

WHY IT MATTERS
Fatal errors break pages, checkout and forms silently

SUGGESTED TOOLS
`wp-content/debug.log` (with `WP_DEBUG_LOG`, `WP_DEBUG_DISPLAY` false), host error log, `tail -f`

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ D6 **Check the security plugin or firewall (WAF) summary for blocked attacks, failed logins and new admin users** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security | A surge in password-guessing attempts or a new admin account are early signs of a hack | Wordfence, Solid Security, Sucuri, Patchstack, Cloudflare WAF events, `wp user list --role=administrator` |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ D7 **(E-commerce) Check orders, failed payments and gateway webhook errors** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Functionality | Payment failures cost revenue immediately | WooCommerce → Status → Logs, Stripe/PayPal dashboards |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ D8 **Check disk space and alert thresholds (warn at 80%)** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Infrastructure | Full disks crash MySQL, block backups and corrupt sessions | Host panel, `df -h`, monitoring agent |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

W1–W10 · 10 TASKS

# Weekly checklist

≈45–90 minutes

☐ W1 **Apply remaining (non-security) plugin and theme updates — staging first, then production; read changelogs** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Plugins & Themes | Updating in weekly batches limits breakage and stops the site falling far behind | `wp plugin list --update=available`, `wp theme update --all`, staging site |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W2 **Run a malware scan and check that core and plugin files have not been changed** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security | The Patchstack/Monarx 2026 report found that attackers increasingly inject code into legitimate core, plugin and theme files, and the Lock360 family rewrites cleaned files from server memory | `wp core verify-checksums`, `wp plugin verify-checksums --all`, Wordfence/Sucuri/MalCare scanners |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W3 **Review Apache/Nginx access and error logs: 404/403/5xx counts, recurring warnings, suspicious requests (`xmlrpc.php`, floods of `wp-login.php` requests, probes for `.env` / `.git` files, `?author=` requests that list usernames)** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Logs | Error spikes show broken features; probe patterns show what attackers are targeting | `/var/log/nginx/`, `/var/log/apache2/`, GoAccess, `grep " 500 "`, Cloudflare analytics |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W4 **Submit each contact/lead form and confirm delivery to the inbox (not spam)** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Functionality | Silent form or mail failures lose leads for weeks before anyone notices | Form plugin entries, WP Mail SMTP / FluentSMTP logs, mail-tester.com |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W5 **(E-commerce) Run a full test checkout in sandbox mode: cart → payment → confirmation email → order status** — CRITICAL (if applicable)

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Functionality | Checkout is the revenue path; plugin updates often break it | WooCommerce sandbox, Stripe test mode |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W6 **Verify SSL certificate expiry date and that the last auto-renewal succeeded** ● CRITICAL

AREA: Security

WHY IT MATTERS: Certificates now last ≤ 200 days and Let's Encrypt no longer emails expiry warnings

SUGGESTED TOOLS: `openssl s_client -connect example.com:443 | openssl x509 -noout -dates`, SSL Labs, external cert monitor

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W7 **Delete spam/trash comments and expired transients (temporary cached data)** ○ MEDIUM

AREA: Database

WHY IT MATTERS: Expired transients stay in the `wp_options` table until something reads them, and spam bloats tables

SUGGESTED TOOLS: `wp comment delete $(wp comment list --status=spam --format=ids)`, `wp transient delete --expired`

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W8 **Check Search Console for new critical issues: indexing drops, security issues, manual actions** ● HIGH

AREA: SEO

WHY IT MATTERS: Sudden drops out of Google's index or security warnings hit traffic quickly

SUGGESTED TOOLS: Google Search Console (Pages, Security & Manual Actions)

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W9 **Review server resource trends: CPU, RAM, PHP workers, slow responses and time to first byte (TTFB)** ○ MEDIUM

AREA: Infrastructure

WHY IT MATTERS: A server that slowly fills up is heading for an outage, and slow responses hurt load times (LCP)

SUGGESTED TOOLS: Host metrics, New Relic, `htop`, Query Monitor

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ W10 **Spot-check homepage, top landing pages, menus and navigation on desktop and mobile** ○ MEDIUM

AREA: Functionality

WHY IT MATTERS: Catches visual breakage from updates or cache issues

SUGGESTED TOOLS: Manual check, visual regression (BackstopJS, Percy)

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

M1-M16 · 16 TASKS

# Monthly checklist

≈2-4 hours

☐ M1 **Audit plugins and themes. Remove anything inactive, unused, duplicated, abandoned (no update in about 12 months, or closed on WordPress.org) or pirated ("nulled"). Delete unused default themes but keep one as a fallback** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Plugins & Themes | Deactivated plugins can still be exploited if their files can be reached, and every plugin is one more way in | `wp plugin list --status=inactive`, `wp plugin delete <slug>`, WordPress.org plugin pages |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M2 **Check compatibility: "Tested up to" vs current WP, PHP compatibility, known conflicts; test upcoming WP 7.2 on staging** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Plugins & Themes | Prevents fatal errors on the next core/PHP upgrade | Health Check & Troubleshooting plugin, Site Health, PHP Compatibility Checker, staging |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M3 **Review user roles and permissions. Remove old accounts, give each user only the access they need, require two-factor login (2FA) for admins and editors, and review Application Passwords** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Core / Security | Accounts with too much access, or left behind by people who have gone, are a common way in for attackers | Users screen, `wp user list`, `wp user application-password list <user>`, 2FA plugin |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M4 **Review login activity history and suspicious patterns (unusual IPs, countries, times)** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security | Catches stolen-password attacks or misuse that the daily summaries miss | Activity log plugin (WP Activity Log, Simple History), WAF logs |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M5 **Check file permissions and look for unexpected files (PHP in `/uploads/`, unknown files in root)** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security | Loose permissions and runnable files in the uploads folder let attackers keep a foothold | `find . -type d ! -perm 755`, `find . -type f ! -perm 644`, `find wp-content/uploads -name "*.php"` |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M6 **Confirm the exact PHP version and patch release, the database version, and whether each is still supported** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Core / Infrastructure | PHP 8.2 reaches end of life on 31 Dec 2026, and PHP2Shell showed that PHP patches come from your host, not from WordPress | Tools → Site Health → Info → Server, `wp --info`, `php -v`, `mysql --version` |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M7 **Measure Core Web Vitals on key templates (home, post, product, checkout) using field data** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Performance | Good = LCP ≤ 2.5 s, INP ≤ 200 ms, CLS ≤ 0.1 at p75. Per the HTTP Archive Web Almanac 2025, only 62% of mobile sites have good LCP and 77% have good INP (vs 97% on desktop), and just 48% pass all three | PageSpeed Insights, Search Console Core Web Vitals report, CrUX, Chrome DevTools Performance panel |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M8 **Review caching and CDN: how often pages are served from cache, object caching (Redis/Memcached), cache exclusions for cart/checkout/account pages, and cache headers** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Performance | A badly configured cache slows pages or, worse, shows one customer's private pages to another | WP Rocket/LiteSpeed Cache/W3TC, Cloudflare, `curl -I` headers, `wp cache flush` |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M9 **Optimise new images: convert to WebP/AVIF, use the right dimensions, lazy-load images below the fold, and add `fetchpriority="high"` to the main above-the-fold image** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Performance | The largest image is usually what LCP measures | ShortPixel, Imagify, EWWW, `wp media regenerate` |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M10 **Find slow database queries, slow plugin code and unnecessary scripts/styles; stop loading scripts on pages that don't need them** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Performance | Slow queries delay the server's first response, and extra JavaScript makes pages slow to react (INP) | Query Monitor, MySQL slow query log, Perfmatters/Asset CleanUp, Chrome Coverage tab |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M11 **Database cleanup: limit/clean post revisions, auto-drafts, trashed posts, orphaned meta; then optimise tables (backup first)** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Database | Bloat slows queries and backups | `wp db export`, `wp post delete $(wp post list --post_type=revision --format=ids)`, `WP_POST_REVISIONS` constant, `wp db optimize`, WP-Optimize |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M12 **Check internal and external links and redirects (chains, loops, 404 targets)** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| SEO / Functionality | Broken links waste search engines' crawl time and frustrate users | Screaming Frog, Ahrefs/Semrush site audit, Redirection plugin 404 log |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M13 **Review XML sitemap and robots.txt (no accidental `Disallow: /`, "Discourage search engines" unchecked, sitemap submitted)** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| SEO | One wrong setting can remove the whole site from Google | `/wp-sitemap.xml` or SEO plugin sitemap, `/robots.txt`, Search Console Sitemaps |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M14 **Verify Google Analytics 4 and Search Console tracking, key events/conversions and consent banner behaviour** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Analytics | Broken tags corrupt reporting and decisions | GA4 Realtime/DebugView, Tag Assistant, Site Kit |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M15 **Restore a recent backup to staging (partial test: database + one directory)** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Backup | CISA advises testing backups regularly; you can't rely on a backup you have never restored | Staging environment, backup plugin restore, `wp db import` |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ M16 **Test mobile responsiveness and current browsers (Chrome, Safari/iOS, Firefox, Edge) on key pages and forms** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Functionality | Many visitors use phones, and Safari quirks often break forms and checkout | BrowserStack, LambdaTest, real devices, DevTools device mode |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

Q1–Q8 · 8 TASKS

# Quarterly checklist

≈half day

☐ Q1 **Full disaster-recovery drill: restore the entire site from the off-site copy onto a clean server. Record how long the restore took and how much recent data was lost** — CRITICAL

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Backup | Proves you can recover from ransomware, a host failure or losing access to your account | Clean server/staging host, backup tool, runbook |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q2 **Review the backup setup: 3-2-1 rule, one copy that can't be changed or is kept offline, encryption, and separate login credentials. Keep at least 7 days of backups, and longer in case a hack goes unnoticed for a while** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Backup | Attackers delete any backups they can reach; CISA recommends offline, encrypted copies | S3 Object Lock, Backblaze B2 Object Lock, Wasabi, separate storage account |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q3 **Security hardening review: change admin, SFTP, database and API passwords and the WordPress security keys (salts) if needed. Confirm `DISALLOW_FILE_EDIT` is set, wp-config.php is 400/440 (or your host's equivalent), XML-RPC is off if unused, and 2FA covers all admins** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security | Limits the damage from leaked passwords and from attackers who manage to log in | `wp config shuffle-salts`, `wp config get`, security plugin hardening panel |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q4 **Database security and health: a dedicated database user with only the access it needs, a non-default table prefix where practical, and remote access turned off. Look for tables left by removed plugins and for oversized settings that load on every page (autoloaded options)** — MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Database / Security | Leftover tables and oversized autoloaded settings slow every page, and a database user with too much access makes a breach worse | `wp db size --tables`, `wp db check`, Site Health autoload warning, phpMyAdmin/Adminer |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q5 **DNS review: correct A/AAAA/CNAME and MX records; SPF, DKIM and DMARC set up and matching your sending domain; CAA records; sensible TTLs; DNSSEC where supported; remove old records** ● HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Infrastructure | Old records pointing at services you no longer use let attackers take over subdomains, and unauthenticated site email goes to spam | `dig`, MXToolbox, DMARC report analyser, registrar/Cloudflare DNS |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q6 **Capacity review: PHP `memory_limit` and `WP_MEMORY_LIMIT`, PHP workers, `max_execution_time`, upload limits, DB size growth, bandwidth trends** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Infrastructure | Prevents 502/503 errors during traffic peaks or campaigns | Site Health → Info, host metrics, load test (k6, Loader.io) |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q7 **Full SEO crawl: indexing coverage, canonical tags, duplicate titles, pages with no internal links, Core Web Vitals trend** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| SEO | Catches slow-building technical SEO issues | Screaming Frog, Search Console Pages report, SEO plugin audit |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Q8 **Full regression test of critical user journeys (forms, search, login/registration, membership, checkout, downloads) plus basic accessibility checks** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Functionality | Cumulative updates can break rarely-used flows | Manual test script, Playwright/Cypress, WAVE, axe |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

Y1–Y7 · 7 TASKS

# Yearly checklist

≈1–2 days

☐ Y1 **Verify domain and hosting expiry dates, auto-renew, payment method, registrar lock, WHOIS/contact emails** — CRITICAL

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Infrastructure | An expired domain takes the site and email offline and can be lost permanently | Registrar dashboard, host billing, calendar reminders 60/30/7 days |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Y2 **Plan the PHP and database upgrades: move off versions that are out of support or close to it (PHP 8.2 ends 31 Dec 2026; MySQL 8.0 ended 30 Apr 2026; MariaDB 10.6 ended 6 Jul 2026). Target PHP 8.4/8.5 and MySQL 8.4 or MariaDB 10.11+** — CRITICAL (this year)

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Core / Infrastructure | Unsupported versions get no security fixes | Staging with target PHP, PHPCompatibility (PHPCS), host support |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Y3 **Major WordPress release plan: test 7.2 (8 Dec 2026) on staging; decide update window (avoid peak trading); confirm auto-update policy for core, plugins and themes** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Core | Major releases bring editor/API changes that can break themes and plugins | Staging, WordPress Beta Tester plugin, changelogs/dev notes |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Y4 **Review the plugin list and premium licences: renew licences (expired licences stop security updates) and replace abandoned or overlapping plugins** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Plugins & Themes | Patchstack's 2026 report counted 1,983 valid vulnerability reports for premium or freemium components (29% of all reports), 76% of them exploitable, and expired licences stop updates | Licence dashboards, plugin inventory spreadsheet |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Y5 **Get an independent security review or penetration test, and review your firewall (WAF) and hosting security** — HIGH

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security | Patchstack's 2026 whitepaper found traditional defences blocked only 12% of WordPress-specific vulnerability attacks (and only 26% of all vulnerability attacks in its pentest of hosting companies) | External auditor, WPScan, Patchstack/Wordfence premium, host security review |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

☐ Y6 **Review hosting: performance vs cost, support quality, how fast the host patches PHP (e.g., its PHP2Shell response), backups, staging, uptime guarantee (SLA)** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Infrastructure | Your host now controls a critical part of your security posture | Host comparison, uptime/SLA reports |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

---

☐ Y7 **Documentation and access audit: update runbooks, credentials vault, list of who has admin/SFTP/DNS/registrar/host access; review privacy policy, cookie consent and legal pages** ○ MEDIUM

| AREA | WHY IT MATTERS | SUGGESTED TOOLS |
|---|---|---|
| Security / Compliance | Without up-to-date documentation, recovering the site and removing departing staff's access both fail | Password manager (1Password, Bitwarden), runbook doc |

STATUS ○ Pending ○ In Progress ○ Completed NOTES / ISSUES FOUND _____

---

APPENDIX

# Section coverage map

Every required area maps to specific tasks:

| # | SECTION | REQUIRED ITEMS → TASK IDS |
|---|---|---|
| 1 | **Plugin & Theme Maintenance** | Updates → `D4`, `W1` · Outdated/inactive/unnecessary → `M1`, `Y4` · Compatibility/conflicts → `M2` · Vulnerabilities & removal → `D3`, `M1` |
| 2 | **WordPress Core Maintenance** | Core updates → `D4`, `Y3` · PHP/DB compatibility → `M6`, `Y2` · User roles → `M3` · Suspicious admins → `D6`, `M3` |
| 3 | **Server & Error Logs** | PHP error logs → `D5` · Apache/Nginx logs → `W3` · 404/403/500 → `W3`, `M12` · Recurring warnings/fatals/suspicious requests → `D5`, `W3` · Resource usage/disk → `D8`, `W9`, `Q6` |
| 4 | **Security Monitoring** | Malware/suspicious files → `W2`, `M5` · Firewall/security plugin status → `D6`, `Y5` · Login attempts/activity → `D6`, `M4` · SSL validity → `W6` · File permissions & DB security → `M5`, `Q3`, `Q4` |
| 5 | **Performance Optimization** | Speed & Core Web Vitals → `M7` · Caching/CDN → `M8` · Images & DB tables → `M9`, `M11` · Slow queries/scripts → `M10` |
| 6 | **Backup & Recovery** | Schedules → `D2` · DB + file backups → `D2`, `Q2` · Restore testing → `M15`, `Q1` · Secure off-site storage → `Q2` |
| 7 | **Database Maintenance** | Revisions/spam/transients → `W7`, `M11` · Health & optimisation → `M11`, `Q4` · Unnecessary records → `Q4` |
| 8 | **Website Functionality** | Forms & email → `W4` · Links → `M12` · Navigation/menus/pages → `W10` · Mobile/browser → `M16` · Checkout/payments → `D7`, `W5` · Full journeys → `Q8` |
| 9 | **SEO & Analytics** | Indexing/crawl errors → `W8`, `Q7` · Sitemap/robots.txt → `M13` · Broken links/redirects → `M12` · GA/GSC tracking → `M14` |
| 10 | **Domain, Hosting & Infrastructure** | Domain/hosting expiry → `Y1` · DNS → `Q5` · Uptime → `D1` · PHP version, memory, capacity → `M6`, `Q6`, `Y2`, `Y6` |

# WP-CLI quick reference for routine maintenance

| PURPOSE | COMMAND |
|---|---|
| **Environment info** | `wp --info` `wp core version` `wp core check-update` |
| **Updates** | `wp core update --minor` `wp plugin update --all` `wp theme update --all` |
| **List installed plugins/themes** | `wp plugin list --fields=name,status,version,update` `wp theme list` |
| **Check files are unchanged** | `wp core verify-checksums` `wp plugin verify-checksums --all` |
| **Users** | `wp user list --role=administrator` `wp user delete <id> --reassign=<id>` |
| **Database** | `wp db export backup.sql` `wp db check` `wp db optimize` `wp db size --tables` |

| PURPOSE | COMMAND |
| --- | --- |
| **Cleanup** | `wp transient delete --expired` `wp cache flush` `wp comment delete $(wp comment list --status=spam --format=ids)` |
| **Security config** | `wp config shuffle-salts` `wp config set DISALLOW_FILE_EDIT true --raw` |
| **Cron** | `wp cron event list` `wp cron event run --due-now` |

### Useful maintenance settings for `wp-config.php`:

`define('DISALLOW_FILE_EDIT', true);`

`define('WP_POST_REVISIONS', 10);` (or a similar limit)

`define('WP_DEBUG', false);` on the live site. Turn on `WP_DEBUG_LOG` with `WP_DEBUG_DISPLAY` set to false only while debugging.

# Maintenance log template

| DATE | CYCLE (D/W/M/Q/Y) | PERFORMED BY | TASKS COMPLETED (IDS) | BACKUP TAKEN BEFORE CHANGES? (Y/N) | VERSIONS AFTER (WP / PHP / DB) | SUMMARY |
|---|---|---|---|---|---|---|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

# Issues log template

| ID | DATE FOUND | TASK ID | ISSUE DESCRIPTION | PRIORITY | OWNER | ACTION TAKEN / PLANNED | STATUS (PENDING / IN PROGRESS / COMPLETED) | DATE RESOLVED |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |

# Recommendations

01 **Handle security alerts every day, not every week.** Patchstack found a 5-hour weighted median before the first attack and about half of high-impact vulnerabilities exploited within 24 hours, so weekly updating is too slow for security fixes. Set up Patchstack, Wordfence Intelligence or WPScan alerts for each site's plugin list. Turn on auto-updates for minor core releases and low-risk plugins. Where you can't update right away, add firewall rules that block known attacks on unpatched plugins ("virtual patching").

02 **Move off PHP 8.2 before 31 December 2026, and off MySQL 8.0 now.** Aim for PHP 8.4 or 8.5, which is what WordPress's Hosting Team recommends for WP 7.1, after testing on staging. PHP 8.3 is acceptable but is only supported until the end of 2027. Ask your host in writing which exact PHP release you are on, to confirm the PHP2Shell fix is installed.

03 **Monitor certificate expiry with a separate tool.** Certificates now last 200 days and will last 100 days from March 2027. Use automatic renewal (ACME) plus an independent expiry monitor, because Let's Encrypt no longer sends reminders.

04 **Upgrade backups from 3-2-1 to 3-2-1-1-0.** Add one copy that can't be changed or is kept offline, with separate login credentials. Never skip the quarterly full restore drill.

05 **Plan the WordPress 7.2 upgrade carefully.** Test it during the release-candidate phase, which starts 17 November. For WooCommerce and lead-generation sites, consider holding the live upgrade until after the holiday sales peak and the first 7.2.x update. Keep applying every security release immediately.

06 **Cut down on plugins every month.** Deleting plugins you don't need is the cheapest security win. Check premium plugins as carefully as free ones, because WordPress.org's new release delay and automated review don't cover them.

07 **Authenticate site email.** Send site email through an SMTP provider that requires login, and publish SPF, DKIM and DMARC records so form notifications and order emails reach inboxes.

# Caveats

- **Version facts change fast.** The WordPress, PHP and database versions above are correct as of 6 November 2026. Re-check WordPress.org's requirements page, the Hosting Handbook compatibility pages and php.net's supported-versions page every quarter. The 8 December date for WordPress 7.2 is a plan, not a guarantee.
- **Official recommendations conflict.** WordPress.org's public requirements page still says "MySQL 8.0 or greater", but MySQL 8.0 reached end of life on 30 April 2026 and the Hosting Team recommends MySQL 8.4. Follow the stricter Hosting Team advice.
- **Some statistics come from a vendor.** Patchstack's figures on attack speed and hosting defences come from its own monitoring and tests, and Patchstack sells protection products. They show the direction of the problem but are not independent measurements. Details of the WordPress 7.1.4 fixes beyond the official announcement come from secondary security coverage, including the "upgrade-window RCE" description and the number of fixes.

- **The details of WordPress.org's plugin release delay have changed.** The June 2026 announcement described a temporary 24-hour delay that applied to auto-updates. The September 2026 post says the delay is now 6 hours and also covers one-click updates from the dashboard. I couldn't find an official post announcing the cut from 24 hours to 6.

- **File permissions depend on the host.** WordPress's hardening guide recommends 755 for folders, 644 for files and 400/440 for wp-config.php. Some managed hosts require 600, 640 or 644 instead, so check with your host before changing permissions or file ownership.

- **The frequencies suit a typical business site.** High-traffic shops and membership sites should do some tasks more often, such as reviewing logs daily and checking restores weekly. Small brochure sites can relax the low-priority items, but never the Critical ones.