Joomla 5.4.6 and 6.1.1 Patch TEN Security Issues

Joomla 5.4.6 and 6.1.1 close ten security issues including an MFA bypass and a com_users privilege escalation. Here is the patch order for an agency running 30+ sites.

Phil E. Taylor | 28 May 2026

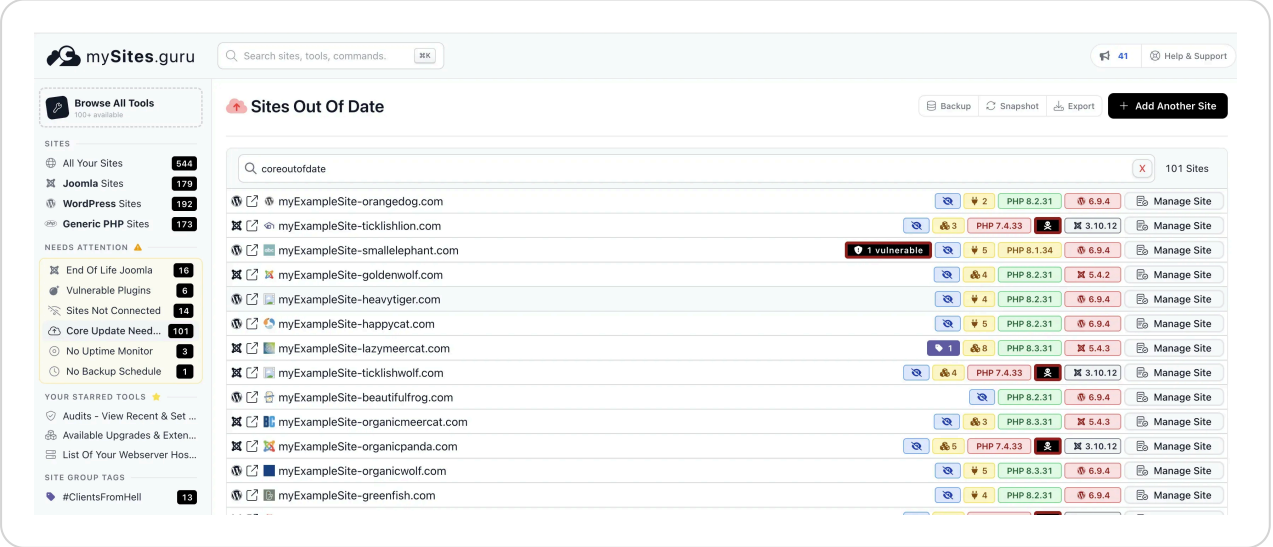
Joomla shipped 5.4.6 and 6.1.1 on 26 May 2026, and the release closes ten CVEs in one go. Two of them are MFA Authentication Bypass issues. One is a High-severity privilege escalation through the com_users batch task. Two more are XSS bugs in the joomla/filter Framework package that have been present since Joomla 3.0.0.

If you only run one Joomla site, you click Update once and move on. If you run thirty, you have a patch night ahead of you, and patch order matters more than you think.

How Does mySites.guru Help Patch Joomla 5.4.6 and 6.1.1 Across Many Sites?

The first thing to do on a release like this is take inventory. Which of your sites are on 5.4.5 right now? Which are still on 4.4.x and need to clear 5.4 before they can reach 6.1.1? Which are on 6.1.0 and ready for a single-step bump?

mySites.guru's "Joomla outdated" check covers exactly this. Every connected Joomla site gets a row showing current version, target version, and a colour code for how far behind it is. After today's release, every 5.4.5 and 6.1.0 site is flagged red. You can manage multiple Joomla sites from one dashboard and patch them in a single bulk operation rather than logging in to each admin.



The Joomla 3.10.12 and 5.4.2 rows above are flagged red. The WordPress 6.9.4 rows are flagged red too, because the 7.0 line shipped. One row carries a "1 vulnerable" badge. You see all of it without logging in to a single admin.

The two parts that matter most across a fleet:

- Bulk Joomla core upgrade runs the update on every selected site in parallel. Backups are taken first if you've enabled that in your site settings. The same tool handles WordPress, so a mixed fleet gets patched in one pass instead of two patch nights.
- **Post-upgrade verification** confirms the new version actually landed. Sites occasionally hang at "downloading" or fail silently at the database fix step, and a quick hash or status check catches those before a client does.

This is precisely the release the mass-upgrade workflow was built for. Ten CVEs in one go, no cherry-picking, every site on 5.4.5 or 6.1.0 needs the patch tonight. Walking through how to upgrade hundreds of Joomla and WordPress sites from one dashboard explains what the click path looks like, and how to update Joomla, Joomla extensions, WordPress, and WordPress plugins from mySites.guru covers the wider day-to-day workflow once core patching is done.

If you want to see what this looks like before signing up, the free audit runs the same outdated-version check on your sites without asking for a credit card.

> **Important**
>
> You cannot upgrade directly from Joomla 4.x to 6.1.1. The 6.1.1 release notes are explicit: update to 5.4 first, then move to 6.x. The 5.4.6 release itself needs 4.4 as the starting point. Plan two-step paths for any 4.x sites you still operate.

What the Hell Is a "CVE"?

CVE stands for Common Vulnerabilities and Exposures. It is a public catalogue of known security holes in software, maintained by MITRE and mirrored by the US National

Vulnerability Database (NVD). Every entry gets a unique ID in the form `CVE-YYYY-NNNNN`, where the year is when the ID was assigned (not necessarily when the bug was found or fixed).

A few things worth knowing if you have not bumped into them before:

- A CVE ID is an identifier, not a severity. "CVE-2026-48898" tells you which bug, not how bad. Severity is a separate rating (Low, Moderate, High, Critical) usually scored against the CVSS framework.
- The same CVE can mean different things on different versions of the software. A flaw might be exploitable on one Joomla minor version and not another, which is why the "Affected" column matters.
- A CVE is not a patch. It is the ticket on the wall. The patch is the new version (here, 5.4.6 and 6.1.1) that closes the bug.
- NVD entries lag. When a vendor like Joomla publishes an advisory, the CVE ID is reserved but the NVD page often shows "RESERVED" or "AWAITING ANALYSIS" for several days while NVD analysts write up severity and impact. The vendor advisory is the authoritative source until they catch up.

If you manage Joomla sites, you do not need to read every CVE. You need to know which ones affect your version and how urgent they are. The table below has both.

What Are the Ten CVEs in Joomla 5.4.6 and 6.1.1?

Joomla's Security Centre published all ten advisories on 26 May 2026. The full list, ordered by severity:

CVE	Severity	Affected	Fix	What it is
CVE-2026-48898	High	4.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	Privilege escalation through the com_users batch task. Improper access check lets a low-privilege backend user elevate beyond their assigned group.

CVE	Severity	Affected	Fix	What it is
CVE-2026-48896	Moderate	4.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	MFA Authentication Bypass. Insufficient state checks in the MFA flow.
CVE-2026-48897	Moderate	4.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	MFA Authentication Bypass. Incorrectly reset session states.
CVE-2026-48899	Moderate	4.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	Incorrect Access Control in sample data plugins. Unauthorised users can trigger sample data install actions.
CVE-2026-48903	Moderate	3.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	XSS in the joomla/filter Framework's `checkAttribute` method. Present since 3.0.0.
CVE-2026-48904	Moderate	4.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	Privilege escalation through com_users webservice endpoints. Improper access check on the group editing API.
CVE-2026-48905	Moderate	3.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	XSS in the joomla/filter Framework's `cleanAttributes` code. Present since 3.0.0.
CVE-2026-48900	Low	4.1.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	ACL bypass in com_scheduler. Low-priv users can edit task types of existing scheduler tasks.
CVE-2026-48901	Low	4.0.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	Incorrect cache key construction for InputFilter objects.
CVE-2026-48902	Low	3.9.0-5.4.5, 6.0.0-6.1.0	5.4.6, 6.1.1	Transport encryption downgrade. Password and username reset links generated as HTTP when "Force SSL" is off. Reaches back to 3.9.0.

NVD entries may show "RESERVED" status for the first few days after disclosure. The authoritative writeups live in the Joomla Security Centre and were live the day the

patches dropped.

Eight of the ten affect every install from 4.0.0 onwards. CVE-2026-48900 is scoped to 4.1.0 and later (com_scheduler shipped in 4.1). CVE-2026-48902, 48903 and 48905 reach further back, into the Joomla 3 lineage.

What About the CVEs Coming in the Next Joomla Release?

Two open pull requests filed on 2026-05-27 line up three more CVEs for the next 5.4 and 6.1 point releases. PRs #47847 (5.4) and #47848 (6.1) bump bundled `symfony/yaml` from 6.4.25 (5.4 branch) and 6.4.34 (6.1 branch) to 6.4.41 to close three Symfony-side advisories:

- CVE-2026-45304 - YAML Parser exponential memory allocation via recursive collection-alias expansion. The classic “Billion Laughs” attack pattern, transplanted from XML into YAML.
- CVE-2026-45305 - ReDoS via catastrophic backtracking in `Parser::cleanup()` regex.
- CVE-2026-45133 - Stack exhaustion via unbounded recursion in nested blocks, sequences, and mappings.

All three are denial-of-service, not remote code execution. None of them are “your site is being hacked tomorrow.” But they affect every version of `symfony/yaml` from 2.0.0 right through to 8.0.11 - which means the buggy code shipped in 2011 and nobody noticed until 2026-05-20.

That gap is the interesting bit.

How Is AI Accelerating the Frequency of Joomla and PHP Library CVEs?

Three Symfony YAML parser CVEs disclosed on the same day, fixing code that has been sitting in production since 2011, is not a coincidence. Vulnerability discovery has shifted gear in the last twelve months, and this release is what the new shape looks like.

For most of open source history, finding a security bug required a human with niche expertise sitting down and reading code. The reason a "Billion Laughs" variant in `symfony/yaml` survived for fifteen years is not that the code is impossibly complex. It is that very few humans had the time or motivation to look at it. The library worked. It parsed YAML. Nobody was going to craft a recursive collection-alias payload at it and watch the memory graph go vertical, because that is a fairly specific thing to be doing on a Tuesday.

Large language models have changed the economics. An LLM can read the entire `symfony/yaml` codebase in seconds, suggest classes of input that might trigger pathological behaviour, and write the fuzzing harness to confirm. A researcher who would have taken a week to manually audit one parser can now sweep a hundred packages in an afternoon. That is what is happening across the ecosystem right now.

The same pattern is visible in the Joomla 5.4.6 / 6.1.1 release itself. The credits list on the Joomla Security Centre names "Doyensec (with Claude/Anthropic Research)" against CVE-2026-48896. That is not unusual any more. It will become the norm.

A few things follow from that:

- The disclosure rate is going up, and will keep going up. Expect more CVEs per Joomla point release, not fewer. The same applies to PHP frameworks, WordPress plugins, and NPM packages. The ground was never clean, it is just being properly inspected for the first time.
- More disclosures means more theoretical issues. Not every CVE will be exploitable in your environment. The three symfony/yaml issues above only matter if attacker-controlled YAML reaches a parser on your site, and for most Joomla installs that path does not exist. The CVE still gets a number, still appears in `composer audit`, and still needs a response from agencies who have promised clients clean scans.

- The patch cadence problem gets worse. When core releases shipped one or two CVEs every few months, an agency could afford to patch on a rolling schedule. When releases routinely close ten core CVEs and reference three more queued in the next minor, patch cadence becomes operational. Ten Joomla sites done by hand was annoying. Fifty during a high-disclosure year is not sustainable without tooling.
- Defence in depth stops being a nice-to-have. Most of the new disclosures are denial-of-service or theoretical-impact issues that would have been low priority five years ago. They are still worth patching, because "we update everything within a week of a CVE" is the only honest answer to "are you on top of security?" when a client asks.

If you run an agency the practical version of this is short. You are not going to read every CVE on every release. You need a system that tells you which of your sites are exposed to which fix, and lets you patch them in bulk before the next batch arrives. That is the free audit pitch, and it is more true this year than it was last year.

Which CVE Should You Patch First in Joomla?

The real answer for most agencies is "all of them, at once, because they all need 5.4.6 or 6.1.1 anyway." There is no cherry-picking individual patches out of this release. But if you have a backlog of upgrades and have to phase the rollout, the priority order is:

1. CVE-2026-48898 (com_users batch task privilege escalation)

This is the only High-severity item. The com_users batch task lets administrators apply changes to many users at once: changing groups, resetting passwords, deleting accounts. An improper access check means a backend user with limited rights can use the batch operation to elevate beyond their assigned permissions.

Patch first on any site where you do not personally control every backend account. That includes multi-author publications, sites with freelance editors, anywhere a junior staff

member has Manager or Publisher access, and any client site where the user list has grown over years without audit.

2. CVE-2026-48896 and 48897 (MFA Authentication Bypass)

Two distinct issues, both Moderate. CVE-2026-48896 is “insufficient state checks” in the MFA flow. CVE-2026-48897 is “incorrectly reset session states.” Both have been present since Joomla 4.0.0.

These only matter if you rely on Joomla’s MFA. If you have super-user accounts protected by 2FA on the assumption that a stolen password alone is not enough, that assumption was wrong from 4.0.0 through 5.4.5 and 6.0.0 through 6.1.0. Patch every site running MFA before you patch the ones without it.

3. CVE-2026-48904 (com_users webservice priv-esc)

A privilege escalation parallel to CVE-2026-48898 but reached via the webservice API rather than the HTML admin. Only matters if you have webservices enabled, which most production sites do.

4. The XSS pair (CVE-2026-48903 and 48905)

Both live in the joomla/filter Framework package: `checkAttribute` and `cleanAttributes`. Both are described as inadequate content filtering allowing XSS payloads to survive attribute sanitisation. The advisories do not publish concrete bypass payloads, so the practical risk depends on what extensions you have installed that pass user-supplied HTML through the framework filter. Patch in normal rollout order.

5. The Low and remaining Moderate items

CVE-2026-48899 (sample data plugin access control), CVE-2026-48900 (com_scheduler task type editing), CVE-2026-48901 (InputFilter cache key), CVE-2026-48902 (HTTP password reset links). Worth closing, but not the items that will be exploited Tuesday morning.

How Do You Verify a Joomla Bulk Update Actually Worked?

The most common patch-night failure I see is not patches that break sites. It is patches that silently don't apply. A site reports the new version in the dashboard but the underlying files are still on the old build. Or the database fix step fails, and the site loads but logs errors.

There are three checks worth running after a bulk Joomla update:

1. **Version string check.** Hit every site's frontend (or your admin's version display) and confirm it reports 5.4.6 or 6.1.1. mySites.guru's site overview does this in one screen.
2. **Database fix.** Joomla's Extensions → Manage → Database tool flags any schema mismatches. Worth running on every site after a Moderate or High patch night. Across 30+ sites, doing this manually is impractical, which is why automating it as part of the update flow saves hours.
3. **MFA round-trip.** Log in to one admin account per site with MFA. The 48896/48897 fixes touched session-state handling, so a quick check on each site catches anything weird.

Two companion pieces to keep near the workflow: detecting locked Joomla scheduled tasks, which sometimes appear in the days after a Joomla version bump if a `cli/joomla.php` task left a row stuck in `task_lock`, and automatic updates for any Joomla extension, which closes the gap between core patches like this one and the long tail of extension CVEs that never make a Joomla.org headline.

How Do You Spot Joomla Sites Still on Old Versions Without Logging Into Each Admin?

This is the harder problem if you have not connected your sites to a fleet manager. Manually, the options are:

- **Reading `administrator/manifests/files/joomla.xml`** on every site, which requires either SSH or FTP access. Doable for a handful of sites, painful for thirty.
- **Watching for the `/administrator/manifests/files/joomla.xml` URL** on each site and parsing the `<version>` element. Some sites have this blocked.
- **Checking the frontend HTML for a generator tag** if it has not been stripped, which it usually has on hardened sites.

The connector-based approach used by mySites.guru reports the running version directly from the site, which is faster, accurate, and works even when the version is intentionally hidden from public scraping. That is the approach I would take across more than five Joomla sites: the multi-site management dashboard lists every connected site by version in one view, and the extension auto-update guidance keeps the long tail of plugins current without manual sweeps.

What Else Shipped in Joomla 5.4.6 and 6.1.1?

It is not all security. The 6.1.1 build also bundles:

- **phpseclib 3.0.52**, picking up fixes from two consecutive bumps in the release cycle (3.0.51 in PR 47620, then 3.0.52 in PR 47738).
- **joomla/oauth2 4.0.2** via PR 47722, fixing OAuth2Client authentication.
- **Three NPM dev-dependency updates** via PR 47622.

These library bumps don't change behaviour for end users but they do close transitive vulnerabilities that would otherwise show up in any composer security audit run against the Joomla codebase.

Further Reading

- Joomla 6.1.1 and 5.4.6 Security and Bugfix Release announcement - the official joomla.org write-up.
- Joomla Security Centre - full list of advisories with affected versions and credits.

- GitHub release notes for 5.4.6 and 6.1.1 - changelog, diff links, and the upgrade-path notes.
- 5.4.5 to 5.4.6 diff - every changed file in the security release.
- 6.1.0 to 6.1.1 diff - same for the 6.x branch.

If you want to see how many of your Joomla sites are still on 5.4.5 or 6.1.0, start with a free audit - no credit card needed.

Frequently Asked Questions

When were Joomla 5.4.6 and 6.1.1 released?

Both versions shipped on 26 May 2026. 6.1.1 was published at 00:00 UTC and 5.4.6 followed at 15:10 UTC. Joomla.org announcement number 5954 covers the joint release.

How many CVEs are fixed in Joomla 5.4.6 and 6.1.1?

Ten advisories, CVE-2026-48896 through CVE-2026-48905. They include two MFA Authentication Bypass issues, a High-severity privilege escalation in com_users batch tasks, an ACL bypass in com_scheduler, a webservice privilege escalation, two XSS fixes in the joomla/filter Framework package, sample data plugin access control, and password reset link transport downgrade.

Which CVE in this release is the most urgent?

CVE-2026-48898 is the only High-severity item. It is a privilege escalation through the com_users batch task, present since Joomla 4.0.0. Anyone with a backend account and access to user batch operations can elevate beyond their assigned group.

Can I upgrade directly to Joomla 6.1.1 from a 4.x install?

No. The 6.1.1 release notes state you must update to 5.4 first, then move to 6.x. If you are on Joomla 4.x and want the patched code, upgrade to 5.4.6 first and then to 6.1.1. The 5.4.6 release itself requires 4.4 or later as the starting point.

Does the MFA bypass affect Joomla sites that don't have MFA enabled?

No. CVE-2026-48896 and CVE-2026-48897 only matter if you rely on Joomla's Multi-Factor Authentication to protect backend accounts. Sites without MFA configured for super-users are not exposed to these two specific issues, but they are still exposed to the other eight CVEs.

Will mySites.guru flag my Joomla 5.4.5 sites as needing this patch?

Yes. The 'Joomla outdated' check picks up any version behind the latest stable, so every site still on 5.4.5 or 6.1.0 surfaces in the dashboard. Use the bulk update workflow to patch all of them in one pass instead of logging in to each admin.

Does this release ship any other library updates?

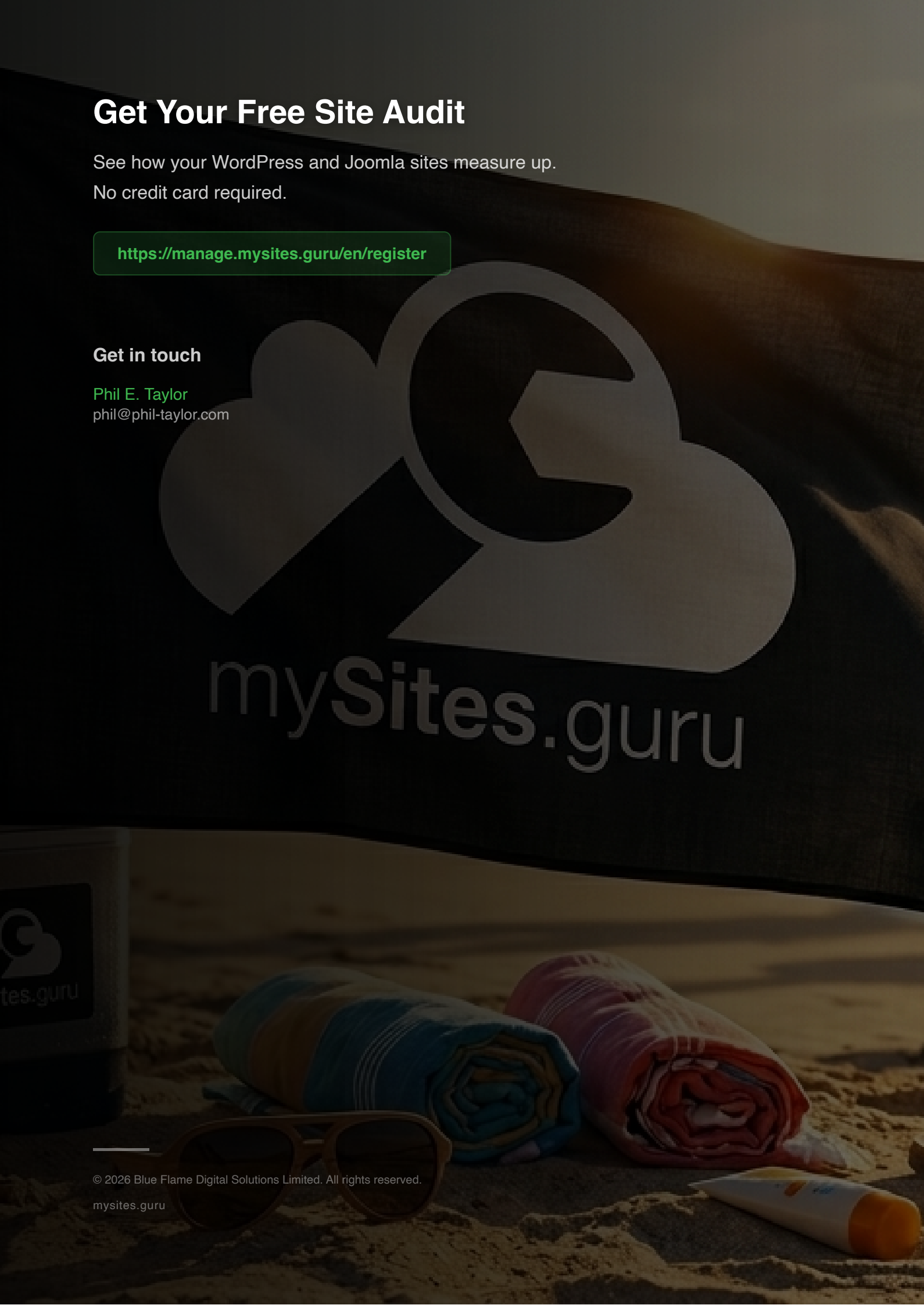
Joomla 6.1.1 ships phpseclib 3.0.52, picking up fixes from two consecutive bumps in the release cycle (3.0.51 in PR 47620, then 3.0.52 in PR 47738). It also includes joomla/oauth2 4.0.2 via PR 47722 and three NPM dev-dependency updates via PR 47622.

Get Your Free Site Audit

See how your WordPress and Joomla sites measure up.

No credit card required.

https://manage.mysites.guru/en/register



Get in touch

Phil E. Taylor
phil@phil-taylor.com