mySites.guru

Phoca Cart 5.2.4 and 6.1.7 fix a front-end SQL injection

Phoca Cart 5.2.4 and 6.1.7 patch an unauthenticated SQL injection in the Joomla extension's product filter. Joomla 5 sites running Phoca Cart 6.x are not offered it.

Phil E. Taylor | 16 August 2026

Active Joomla Extension security alerts: SP Page Builder RCE • JCE 2.9.99.10 • Fabrik: unauth RCE • Phoca Cart: unauth SQLi

On the morning of 16 August 2026, Phoca shipped two releases of Phoca Cart, the e-commerce extension for Joomla: 6.1.7 at 10:23 UTC and 5.2.4 sixteen minutes later. Here is everything Phoca published about what changed.

RELEASE NOTE FOR PHOCA CART 5.2.4, IN FULL

```
This is a security release.
```

Five words. No affected versions, no description of the flaw, no severity, no CVE, no workaround, and nothing to tell a site owner whether this is a quiet hardening tweak or an anonymous visitor reading their customer table.

In fairness to Phoca, that looks like a timing gap rather than a policy: they routinely publish a proper news item for security releases, as they did for Phoca Commander 6.1.4 on 6 August 2026, complete with a changelog. At the time of writing, a few hours after these two packages went out, no such post had appeared for Phoca Cart, and there was no forum thread and no GitHub security advisory either. One may well follow.

This is the same trap we keep flagging, because extension developers are consistently better at writing code than at telling anyone what the code does. JoomShaper shipped a critical Helix3 patch under a changelog that said only “Security Update”. Balbooa’s remote code execution fix arrived under a plain “Fixed” heading, with nothing to say it closed a hole being exploited at the time. Cotton Cloud pushed a security release to every site through the update system while its changelog file still held exactly one entry, for a version long past. The Joomla project’s own guidance covers this as rule 17 of the twenty rules for developers handling a security report: do not hide security fixes in vague language.

Phoca at least uses the word “security”, which puts this ahead of a fix buried under “Fixed”. You are still left with the category and nothing else: no way to tell whether this one goes to the top of the queue or waits until Thursday. The words a vendor puts in a

release note are a poor guide to how urgently you need to update. The version number is the part you can act on.

So we pulled both packages apart to work out what changed. The answer is an unauthenticated SQL injection in the product filter, fixed in six places in one file. The more useful answer, if you manage Joomla sites for other people, is that a large share of the installs affected by this will never be offered the fix by Joomla's own updater, and will keep reporting themselves as up to date.

TL;DR

- Phoca Cart **5.2.4** (Joomla 5) and **6.1.7** (Joomla 6) fix an **unauthenticated SQL injection** in the product filter
- The `a` (attribute) and `s` (specification) request parameters were concatenated into the `WHERE` clause of the product listing query inside hand-written quotes, on a **public page, with no login and no token**
- Six injection points in `admin/libraries/phocacart/search/search.php`, now passed through `$db->quote()`
- **Joomla 4 and Joomla 3 are still unpatched.** We confirmed the identical vulnerable code in 4.0.12 and 3.5.8, the newest releases offered to those platforms
- **Joomla's updater will not offer the fix to a Joomla 5 site running Phoca Cart 6.x.** The update feed offers it 5.2.4, which is lower than what it has, so Joomla shows nothing
- **CVE-2026-74251**, published by the Joomla CNA the same afternoon at CVSS 4.0 9.3, Critical. Its stated version range is wrong at both ends and understates who is affected
- **This is not our find.** Phoca found and fixed this one; we analysed the published diff afterwards

What Phoca Cart 5.2.4 and 6.1.7 actually changed

A warning for anyone trying to check this themselves, because the obvious method gives the wrong answer. GitHub's compare view between the two tags is useless here. The `5.2.4` tag points at commit `9549020f`, which is the same commit as `6.1.7`. Comparing `5.2.3...5.2.4` therefore reports 19 commits and 223 changed files, eight months of unrelated 6.x development, none of which is in the 5.2.4 package that people actually install.

The real 5.2.4 code exists only in the release ZIP. Download both packages, unzip, and diff the trees:

```
curl -sLO https://github.com/PhocaCz/PhocaCart/releases/download/5.2.3/com
curl -sLO https://github.com/PhocaCz/PhocaCart/releases/download/5.2.4/com
unzip -q com_phocacart_v5.2.3.zip -d v523 && unzip -q com_phocacart_v5.2.4
diff -rq v523 v524
```

Two files differ. `phocacart.xml` carries the version bump from 5.2.3 to 5.2.4 and a new creation date, and the rest of its diff is line-ending noise. Everything else is `admin/libraries/phocacart/search/search.php`.

That is a clean, tightly scoped security release. There is no feature work smuggled in alongside the fix, which makes it a low-risk update to apply. The bundled plugins in the package are untouched and stay at 5.1.1.

How does the Phoca Cart SQL injection work?

The vulnerable code is in `PhocacartSearch::getSqlPartsArray()`, which builds the SQL for filtering products by attribute or specification. Filter values were pasted directly into the query string inside quotes the code wrote itself:

```
// Before, in 5.2.3 and 6.1.6: single-quote break-out
$inA[] = '(at2.alias = ' . $db->quote($k) . ' AND v2.alias IN (' . '\'' .
```

```
// Before: double-quote break-out
$inAS[$iA] = 'at2.alias = ' . $db->quote($k) . ' AND v2x' . $iA . '.alias
```

Note that the array *key* was already going through `$db->quote()`. Only the values were left raw. The fix pre-quotes the whole array and quotes the scalar:

```
$aQuoted = [];
foreach ($a as $vQ) {
    $aQuoted[] = $db->quote($vQ);
}

$inA[] = '(at2.alias = ' . $db->quote($k) . ' AND v2.alias IN (' . implode
$inAS[$iA] = 'at2.alias = ' . $db->quote($k) . ' AND v2x' . $iA . '.alias
```

The same change is applied six times, covering the attribute and specification paths in both the “match any” and “match all” filter modes, each duplicated for the multilingual and single-language code paths.

Nothing stands between that concatenation and an anonymous visitor. The front-end product listing model reads both parameters straight from the request:

```
$this->setState('a', $app->getInput()->get('a', '', 'array')); // Attribu
$this->setState('s', $app->getInput()->get('s', '', 'array')); // Specifi
```

Joomla’s `array` filter casts the input to an array. It does not sanitise the individual values inside it, so whatever the visitor sends arrives intact at the string concatenation above. The resulting fragment goes into the `WHERE` clause and the `LEFT JOIN` of the product listing query. That is a public category or product listing page: no account, no token, no user interaction.

Note on what we did and did not test

This analysis is a read of the published diff and the surrounding source, not a live exploitation attempt. We have not run an injection against any site, and we are not publishing a payload. The vulnerable pattern and its reachability are both plain in code that anyone can download,

which is why we are comfortable describing the mechanism while leaving the working request to the imagination.

Which Phoca Cart versions are affected?

Everything before the two fixed releases, on every branch. We extracted the newest package offered to each Joomla generation and checked the same function in each:

Joomla version	Newest Phoca Cart offered	Contains the fix?
Joomla 6	6.1.7	Yes
Joomla 5	5.2.4	Yes
Joomla 4	4.0.12	No
Joomla 3	3.5.8	No

The Joomla 4 and Joomla 3 packages carry the identical unquoted concatenation, and the Joomla 4 front-end model has the same unfiltered `a` and `s` binding. No fixed release exists for either branch.

CVE-2026-74251, and where its version range is wrong

The Joomla project's CNA published CVE-2026-74251 at 12:51 UTC on 16 August 2026, a little over two hours after the second package went out. It scores the flaw **CVSS 4.0 9.3, Critical**, on the vector `AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H`, and classifies it as CWE-89. That is the official severity, and it matches what the code does: reachable over the network, no privileges, no user interaction, and high impact on the confidentiality, integrity and availability of the database behind the shop.

The description confirms the mechanism independently of our diff. It names the `a[]` and `s[]` GET array parameters on the public shop items page, says they are

“concatenated raw into SQL WHERE clauses without parameterization or escaping”, and states the consequence as “full database extraction via time-based blind techniques”.

Two details in the record do not survive checking, and both make a site look safer than it is.

The affected range is given as 5.0.0 to 6.1.16, and there is no Phoca Cart 6.1.16. The highest tag ever published on the 6.x line is 6.1.7, which is the release carrying the fix. The last vulnerable 6.x build is 6.1.6. Read literally, the range flags the one release that contains the fix.

The 5.0.0 floor leaves out the two branches with no fix at all. We extracted 4.0.12 and 3.5.8 and found the identical unquoted concatenation in both, as the table above sets out. Those are the newest packages Phoca offers to Joomla 4 and Joomla 3 sites, so a range starting at 5.0.0 reads as an all-clear for exactly the shops with nowhere to update to.

Act on the 9.3, and treat the version range as narrower than reality. If you are running Phoca Cart below 5.2.4 on the 5.x line, below 6.1.7 on the 6.x line, or anything at all on 4.x or 3.x, you are affected.

Why Joomla’s updater will not offer the fix to some sites

This is where a routine extension update turns into an operational problem.

Phoca’s update feed publishes one entry per Joomla generation, each gated by a `targetplatform` regex:

```
<version>5.2.4</version>  <targetplatform name="joomla" version="5\.*"/>
<version>6.1.7</version>  <targetplatform name="joomla" version="6\.*"/>
```

Joomla’s extension update adapter matches that regex against the running Joomla version:

```
$product == $this->currentUpdate->targetplatform['NAME']
&& preg_match('/^' . $this->currentUpdate->targetplatform['VERSION'] . '/
```

A Joomla 5.4.7 site therefore only ever sees the 5.2.4 entry. The 6.1.7 entry is discarded before any version comparison happens. Then `Updater::findUpdates()` checks that the offered version is actually newer than what is installed, using a `gt` comparison:

```
if (version_compare($current_update->version, $data['version'], $operator)
    $current_update->extension_id = $eid;
    $retVal[] = $current_update;
}
```

With 5.2.4 offered and 6.1.5 installed, `version_compare('5.2.4', '6.1.5', 'gt')` is false. No update row is created. The site reports nothing to update.

So a Joomla 5 site running Phoca Cart 6.x is stranded on a vulnerable version while its update screen looks clean. The extension never appears in the update list at all, and an empty update list is indistinguishable from a patched site. No warning, no flag, no clue that anything is outstanding.

> **Warning**
> If you run Joomla 5 with Phoca Cart 6.x, an empty update list is not evidence you are patched. Check the installed Phoca Cart version directly under System, Manage, Extensions, and compare it to 6.1.7 by eye.

Phoca Cart 6.x installs and runs on Joomla 5 perfectly well, which is how these sites got into this state in the first place. There is no Joomla version gate in the installer, so the manual fix works: download the 6.1.7 package and install it through Extensions, Install, Upload Package File.

The Joomla 4 branch is still unpatched

Joomla 4 sites are offered Phoca Cart 4.0.12, which we confirmed still contains the vulnerable code. There is no 4.x release with the fix, so there is no in-branch remedy. The realistic options are migrating the site to Joomla 5 or 6 and taking the patched build, or removing the extension.

Phoca's own README is candid about which branches it considers finished. As of the 6.1.7 tag it labels the 3.x, 4.x **and 5.x** lines "no longer supported", listing 3.5.8, 4.0.12 and 5.2.4 against them, with only 6.x unqualified. So Phoca backported this fix to one branch it has already declared unsupported, and not to the two older ones. That is a defensible line to draw, and it is better than nothing, but it does mean the Joomla 4 answer is a migration rather than an update.

Joomla 4 itself has been out of support for a while: bug-fix support ended in October 2024 and security support in October 2025, and the 4.x series no longer appears on Joomla's roadmap at all. A Joomla 4 shop is therefore running an unsupported CMS and an unsupported, unpatched shopping cart at the same time.

How the affected Joomla sites break down

Of the Phoca Cart installs visible across the Joomla sites we manage:

- **Not one was on a fixed version.** Every install we can see predates 5.2.4 and 6.1.7, which is expected on release day and is exactly why the update path matters
- **About 55%** will be offered the fix normally by Joomla's updater and can be patched from the update screen
- **About 39%** are Joomla 5 sites running Phoca Cart 6.x. These get offered a lower version, so their update screens show nothing at all
- **About 6%** are Joomla 4 sites, where no fixed release exists

That means **roughly 45% of the Phoca Cart installs we can see cannot get this fix through Joomla's own updater.** If you manage Joomla shops and your process is "check the update screen, apply what is listed", that process misses close to half of the affected sites here and gives you a clean bill of health while doing it.

The SQL injection itself is an ordinary bug, competently fixed, in a tightly scoped release. Getting that fix onto the sites that need it is the harder half of the job.

How do I find every Phoca Cart install across my Joomla sites?

The check you need is not “what version of Phoca Cart is this” on its own. It is the extension version and the Joomla version together, because the pair is what decides whether the updater will do anything.

mySites.guru lists every extension across every connected site, so you can search for Phoca Cart once and get back every site running it with its version, alongside the Joomla version for that site. That turns a site-by-site login into a single search. It is part of the subscription rather than a free lookup tool, and if you already manage a batch of Joomla sites it is the difference between knowing which shops are stranded and guessing.

For this specific release, the triage is:

1. Find every site running Phoca Cart, with its Joomla version
2. Joomla 6 sites below 6.1.7, and Joomla 5 sites below 5.2.4 on the 5.x line: update from the Joomla update screen
3. Joomla 5 sites running any 6.x version: download 6.1.7 and install it manually, because the updater will not offer it
4. Joomla 4 and Joomla 3 sites: no fix exists, so plan the migration or drop the extension

If a site turns out to have been running a vulnerable version on a public shop for a long time, updating closes the hole but tells you nothing about what happened before. If you need someone to go through a site properly, fix.mysites.guru is a fixed fee of £120 per incident, screened first so you are not charged if it cannot be fixed.

Did mySites.guru find this one?

No. This is Phoca's own security release. We had no involvement in finding it, no involvement in reporting it, and we learned about it the same way anyone else could, by reading a published diff after the packages went out.

We say so because we have published a run of Joomla extension SQL injection disclosures this year that we did find, including AcyMailing, DPCalendar and EasyStore, and this post sits close enough to those to be mistaken for one. Credit here belongs to Phoca. We have reported a separate flaw in Phoca Download in the past, which Jan Pavelka fixed the same day, so our experience of Phoca acting quickly on a security report is first hand.

CVE-2026-74251 names no finder either, which fits Phoca having caught this internally. The Joomla CNA does credit researchers when there is one to credit, so an empty credits field is a data point rather than an oversight.

Severity

9.3
CVSS 4.0

CRITICAL The Joomla CNA's score for CVE-2026-74251

Reachable over the internet on a public shop page with no account, no token and no user interaction. The CNA scored confidentiality, integrity and availability all high, because the injection sits inside a WHERE clause the attacker controls outright rather than in a value the query merely reads. The 5.x and 6.x lines have a patched release. The 4.x and 3.x lines carry the same code with nothing to update to.

No login needed | Public shop page | Full database read | Password hashes exposed | No fix on Joomla 3 or 4

Field	Detail
Extension	Phoca Cart for Joomla (`com_phocacart`)
Vendor	Phoca, Jan Pavelka (phoca.cz)

Field	Detail
Type	Unauthenticated SQL injection (CWE-89) through the `a[]` attribute and `s[]` specification filter parameters
CVE	CVE-2026-74251, published by the Joomla CNA at 12:51 UTC on 16 August 2026
CVSS 4.0	9.3 (Critical), `AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N` (the Joomla CNA's score)
Vulnerable file	`admin/libraries/phocacart/search/search.php`, six injection points now passed through `$db->quote()`
Impact	Anonymous read of any table in the shop's database, including customer records and password hashes, by time-based blind extraction
Finder	Phoca, who found and fixed it themselves. The CVE record credits nobody, and this is not a mySites.guru finding
Affected versions	Every release below 5.2.4 on the 5.x line and below 6.1.7 on the 6.x line, plus the whole of the 4.x and 3.x branches. The CVE record states 5.0.0 to 6.1.16, which is wrong at both ends: there is no 6.1.16, and the 5.0.0 floor omits 4.0.12 and 3.5.8, which we extracted and confirmed unpatched
Fixed in	5.2.4 (Joomla 5 line) and 6.1.7 (Joomla 6 line), both released 16 August 2026. No fixed release on the 4.x or 3.x branches

Further reading

- CVE-2026-74251, the Joomla CNA's record, for the official CVSS 4.0 score and vector
- Phoca Cart 5.2.4 release and 6.1.7 release on GitHub, with the downloadable packages
- Phoca Cart project page for documentation and the vendor's own download portal
- Joomla's update server documentation explaining `targetplatform` and how update servers are matched

- How to enable Joomla extension auto-updates safely on getting patches applied without breaking sites

Frequently Asked Questions

What did Phoca Cart 5.2.4 and 6.1.7 fix?

Both releases fix an SQL injection in the product filter. The attribute and specification filter values, the `a` and `s` request parameters on a product listing page, were concatenated into the WHERE clause of the product query inside hand-written quotes instead of being passed through the database's quoting. Both versions now run every filter value through Joomla's `$db->quote()`. The component change is two files: the version manifest, and `admin/libraries/phocacart/search/search.php`.

Which Phoca Cart versions are affected?

Every release before 5.2.4 on the Joomla 5 line and before 6.1.7 on the Joomla 6 line. We also confirmed the same unpatched code in 4.0.12, the newest release offered to Joomla 4 sites, and in 3.5.8 for Joomla 3. There is no fixed release for the 3.x or 4.x branches, so Joomla 4 shops have no upgrade path inside the 4.x line.

Do I need to be logged in to exploit this?

No. The two filter parameters are read straight from the request by the front-end product listing model with Joomla's `array` filter, which does not sanitise the individual values. Any anonymous visitor can supply them on a public product listing or category page, with no account and no token.

Why does Joomla say no update is available for Phoca Cart?

If you run Joomla 5 with Phoca Cart 6.x, Joomla's updater will not offer the fix. Phoca's update feed maps Joomla 5 to the 5.x release and Joomla 6 to the 6.x release, so a Joomla 5 site is only shown Phoca Cart 5.2.4. That is lower than the 6.x version already installed, so Joomla discards it and reports nothing to update. Download the 6.1.7 package from GitHub or phoca.cz and install it manually.

Was this vulnerability found by mySites.guru?

No. This is Phoca's own security release and we had no part in finding or reporting it. We analysed the published diff after the fact, the same way anyone can. We have previously found and reported a separate flaw in Phoca Download, but this one is not ours.

Is there a CVE for the Phoca Cart SQL injection?

Yes. The Joomla project's CNA published CVE-2026-74251 on 16 August 2026, hours after the two packages shipped, scoring it CVSS 4.0 9.3 Critical under CWE-89. Two details

in the record do not hold up: it gives the affected range as 5.0.0 to 6.1.16, but Phoca Cart 6.1.16 does not exist and the last vulnerable 6.x build is 6.1.6, and the 5.0.0 floor leaves out 4.0.12 and 3.5.8, which we extracted and found carrying the identical unpatched code.

How do I find every Phoca Cart install across my Joomla sites?

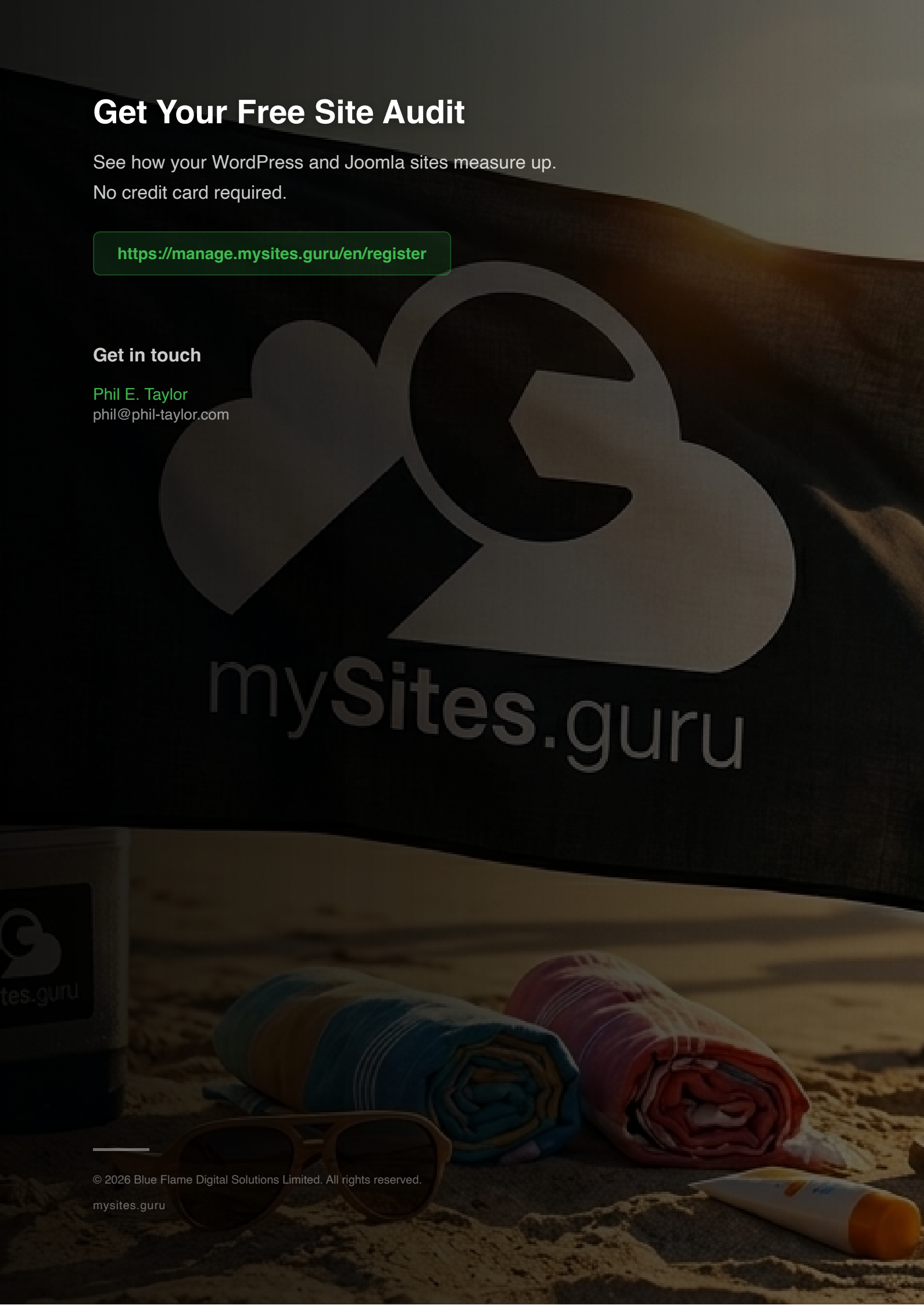
mySites.guru lists every extension on every connected site, so you can search for Phoca Cart once and get back the sites running it with their version numbers, instead of logging into each site to check. That matters here because the version alone does not tell you whether the update will reach a site: you need the Joomla version next to it.

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