mySites.guru

Sourcerer 14.0.0 fixes PHP execution from unverified content

Sourcerer, the Joomla extension, ran PHP from page content it could not trace to a verified source. CVE-2026-74253 scores 10.0 critical. Update to 14.0.0.

Phil E. Taylor | 17 August 2026

On 17 August 2026 Regular Labs released version 14.0.0 of Sourcerer, the Joomla extension that lets you place PHP, JavaScript and CSS inside your content. The changelog carries one entry tagged both `[SECURITY FIX]` and `[BC BREAK]`:

SOURCERER 14.0.0 CHANGELOG, THE SECURITY ENTRY IN FULL

```
[SECURITY FIX] [BC BREAK] Prevents reflected or otherwise unverified rendered Sourcerer code from executing by default while preserving verified article and Custom module code
```

It runs that code wherever you put it: articles, modules, components, the page head. To do that it inspects the finished page on every render and executes the tags it finds. The question 14.0.0 answers is one Sourcerer never used to ask about most of the page: where did this code come from?

TL;DR

- **Sourcerer 14.0.0 is the fix.** The CVE record puts the affected range at 1.0.0 to 13.1.1, so every 13.x build is in range, from 13.0.0 to 13.1.1
- Before 14.0.0, the only content whose origin Sourcerer verified was **article text**. Code that reached the page any other way simply ran
- From 14.0.0, a block executes only if the plugin can trace it to a trusted source: **article content, or a Custom module whose rendered text still matches what**

is stored

- **HTML-escaping is not a mitigation.** Sourcerer decodes HTML entities inside its own tags on purpose, so WYSIWYG-authored code works
- **PHP execution is enabled by default**, and the default forbidden-function list blocks shell functions, not file writes
- **Tracked as CVE-2026-74253, CVSS 4.0 10.0, critical**, published by the Joomla CNA on 17 August 2026. The earlier CVE-2026-64796 is a separate issue and stops at 12.2.8
- **This is a deliberate breaking change.** Some legitimate code will stop running until you allow it, which is the correct trade
- Sourcerer is installed on **thousands** of the Joomla sites we manage, and roughly **two thirds of those installs are on an affected 13.x build** today

What Sourcerer 14.0.0 changed

The fix is structural rather than a patched line. Version 14.0.0 adds a trust-marking step. Content Sourcerer can vouch for gets marked as it passes through the page, and only marked blocks may execute code. Two origins qualify. Article content does, as it did before. A Custom module does when the text being rendered still matches what is stored in the database for that module, which is how the plugin knows the code you are about to run is the code an administrator actually saved.

Everything else now counts as unverified. On the Free edition those blocks are removed rather than run. On Pro, each unverified CSS, JavaScript or PHP type has to be allowed explicitly before it will execute again.

Two halves of the same Joomla extension problem

This is the second Sourcerer security release in under a month, and the two are halves of one problem.

On 22 July 2026 Regular Labs shipped a coordinated hardening release across its whole catalogue. Sourcerer 13.0.0 was part of it, and the CVE that came out of it, CVE-2026-64796, describes the Free edition failing to "require both the article creator and last modifier to be Super Users before executing article PHP". That closed the article path: from 13.0.0, PHP in an article only runs if trusted people wrote and last touched it.

August's release closes the rest of the page. The article path had an author check; the module, component, head and final-rendering positions had no origin check at all. Getting the first half right is what made the second half visible.

The practical consequence is that patching one does not patch the other. CVE-2026-64796 gives its affected range as 1.0.0 to 12.2.8. A site sitting on 13.1.1 is outside that range, is not flagged by it, and is still affected by what 14.0.0 fixes. The record for the August issue, CVE-2026-74253, runs from 1.0.0 to 13.1.1 and covers the whole line up to the fix.

Why escaping the input does not save you

The natural assumption is that a site is safe as long as it escapes what users type, because escaped angle brackets cannot open a PHP block. With Sourcerer that assumption falls apart, and the reason is a documented feature, not a bug.

Code written in a WYSIWYG editor arrives with its angle brackets converted to HTML entities. So that code still runs, Sourcerer decodes entities inside its own tags before handling the contents. The decoding has no way to tell code an administrator typed into TinyMCE from text that arrived from somewhere else. Escaped text is not automatically inert, and "we escape our output" is not the control that protects a site here.

Two other defaults matter when you are weighing urgency. PHP execution is on out of the box. The default list of forbidden PHP functions covers shell-execution functions, so it stops a payload shelling out, but it does not cover writing files.

We are deliberately not publishing a working request or a payload. Nothing above is a recipe, and you do not need any of it to decide whether to update.

Which Joomla sites are affected?

Every site running a build below 14.0.0. The CVE record starts its range at 1.0.0, so there is no older version to fall back to, and the 13.x builds are the ones worth naming, because those sites took a security update a month ago and are still affected. Note that 13.1.1 is twelve days old at the time of writing, so recently updated sites are affected too. Being current is not the same as being safe when the fix is a major version bump.

One trap when you check versions by hand: Pro builds append a `PRO` suffix, so a Pro site reports `13.1.1PRO` rather than `13.1.1`. String comparisons and spreadsheet sorts handle that badly, and it is why a "less than 14.0.0" rule written the obvious way can misjudge a Pro build in either direction. Compare the numeric part and treat any 13.x build, Free or Pro, as affected.

Sourcerer is installed on thousands of the Joomla sites we manage, and roughly two thirds of those installs are on a 13.x build today. Only a handful had moved to 14.0.0 within hours of release.

What should I do now?

Update Sourcerer to 14.0.0. Then plan for the breaking change rather than being surprised by it: legitimate code Sourcerer was picking up from a module, component, head or final-rendering position will stop running until you allow that type explicitly on Pro. If the site depends on Sourcerer for anything structural, test on a staging copy first.

If you cannot update immediately, disabling the Sourcerer system plugin removes the exposure, because that is the plugin that renders Sourcerer tags. It also removes the functionality, so treat it as a stop-gap rather than a fix.

Disclosure and severity

This is not our find. The CVE record credits Łukasz Rybak as the finder; the vendor changelog credits nobody, and Regular Labs published no separate advisory beyond the entry quoted at the top of this post. We read the 13.1.1 and 14.0.0 packages side by side to work out what the entry means, and we confirmed the behavioural difference between the two versions on our own Joomla 6 test site.

Update: the official record arrived the same day

We first published this post with our own provisional scores, because no CVE covered the 13.x line. Hours later the Joomla CNA published CVE-2026-74253 and scored it CVSS 4.0 10.0, critical, with an affected range of 1.0.0 to 13.1.1. We said an official vector would be authoritative over ours, so the score below is the CNA's, with our own figures and the reason they were lower kept underneath it.

The record classifies the weakness as CWE-94, improper control of generation of code, which is the same call we made: code from an unverified origin was executed as PHP. It adds CAPEC-242, code injection. The impact once a block runs is complete: arbitrary PHP in the site's own process.

10.0
CVSS 4.0

CRITICAL **The Joomla CNA's score, and the maximum possible**

CVE-2026-74253 is scored AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H, which is every metric at its worst: reachable over the network, no login, no user interaction, nothing that has to be true first, and consequences that reach past the extension into the system around it. The record is titled "Unauthenticated RCE through unverified reflected user input in Sourcerer < 14.0.0".

CVE-2026-74253 | Arbitrary PHP execution | Escaping is not a mitigation | PHP enabled by default | No login needed

Our own provisional scores were lower, and the difference is reachability. We could not reach the flaw through Joomla core alone: core's input filtering stripped our test

payload out of a reflected search parameter, raw and entity-encoded alike, so we could demonstrate no anonymous route at all on a stock site. We scored what we had proved, treating a suitable delivery position as a requirement (`AT:P`) and putting the everyday case behind a login. The CNA scored the flaw itself, with the reflected route its finder demonstrated. Its record is the one to quote, and ours is here for completeness.

Our provisional variant	Vector (`CVSS:4.0/...`)	Score	Severity
Low-privileged account, raw-rendered position outside the article path	`AV:N/AC:L/AT:P/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`	7.7	High
Anonymous, where another extension renders raw request input	`AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`	9.2	Critical

Field	Detail
Component	Sourcerer for Joomla (`plg_system_sourcerer` and `plg_editors-xtd_sourcerer`), Free and Pro
Vendor	Regular Labs (regularlabs.com)
Type	Execution of code from an unverified origin
CVE	CVE-2026-74253, published 17 August 2026 by the Joomla CNA. CVE-2026-64796 covers the earlier, separate issue in 1.0.0 to 12.2.8
CVSS 4.0	10.0 (critical), `AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H` . Our own provisional figures, published before the record existed, were 7.7 and 9.2
CWE	CWE-94 (improper control of generation of code), CAPEC-242 (code injection)
Affected versions	1.0.0 up to and including 13.1.1 per the CVE record, Free and Pro. Every 13.x build is in range
Fixed in	14.0.0, released 17 August 2026
Finder	Łukasz Rybak, credited in the CVE record. Not a mySites.guru find, and the vendor changelog credits nobody
Vendor advisory	None beyond the changelog entry

Further Reading

- Sourcerer downloads and changelog at Regular Labs
- CVE-2026-74253, the official record for this issue (CVSS 4.0 10.0, CWE-94)
- CVE-2026-64796, the earlier Sourcerer issue fixed in 13.0.0
- CWE-94: Improper Control of Generation of Code at MITRE
- CVSS 4.0 specification at FIRST
- Joomla's guidance for developers handling a security report

Frequently Asked Questions

What did Sourcerer 14.0.0 fix?

Sourcerer 14.0.0, released on 17 August 2026, carries a changelog entry tagged [SECURITY FIX] [BC BREAK] that reads: 'Prevents reflected or otherwise unverified rendered Sourcerer code from executing by default while preserving verified article and Custom module code.' Sourcerer's job is to run PHP, JavaScript and CSS placed in Joomla content. Before 14.0.0 it checked where that code came from only for article content. Code that showed up anywhere else in the finished page simply ran. From 14.0.0 a block only runs if the plugin can trace it to a source it trusts, which means article content or a Custom module whose rendered text still matches what is stored in the database.

Which Sourcerer versions are affected?

The CVE record, CVE-2026-74253, gives the affected range as 1.0.0 up to and including 13.1.1, so every build below 14.0.0 is affected and 14.0.0 is the fix. The 13.x builds are the ones worth calling out, because those sites were patched a month ago and are still affected. An earlier, separate issue, CVE-2026-64796, covers 1.0.0 to 12.2.8 and was fixed in 13.0.0 on 22 July 2026, so being patched against that one does not help here. If you are on any 13.x build, including 13.1.1 released on 5 August 2026, you need 14.0.0.

Is there a CVE for this?

Yes. It is CVE-2026-74253, published by the Joomla CNA on 17 August 2026, hours after this post first went up. The record is titled 'Unauthenticated RCE through unverified reflected user input in Sourcerer < 14.0.0', gives an affected range of 1.0.0 to 13.1.1, classifies it as CWE-94, and scores it CVSS 4.0 10.0, the maximum. It credits Łukasz Rybak as the finder; the vendor changelog credits nobody. We first published with our own provisional scores of 7.7 and 9.2 because no record covered the 13.x line, and the official vector supersedes them.

Does HTML-escaping the input protect a site?

No, and this is the part most people would guess wrong. Sourcerer deliberately decodes HTML entities inside its own tags, because code written through a WYSIWYG editor arrives with its angle brackets escaped. That decoding is a feature for legitimate content, but it also means escaping attacker-supplied text does not neutralise a Sourcerer tag hidden in it. Escaping is not the control that saves you here. The version number is.

Will updating to 14.0.0 break anything?

It can, and Regular Labs marked it as a breaking change deliberately. On the Free edition, existing Sourcerer tags that were only generated during component, module, head or final rendering are now removed rather than run. On Pro, each unverified CSS, JavaScript or PHP type has to be allowed explicitly. If you have legitimate code that was being picked up from one of those positions, it will stop working until you allow it. That is the trade the vendor chose, and it is the right way round: code that cannot prove where it came from does not run.

How do I find every Sourcerer install across the Joomla sites I manage?

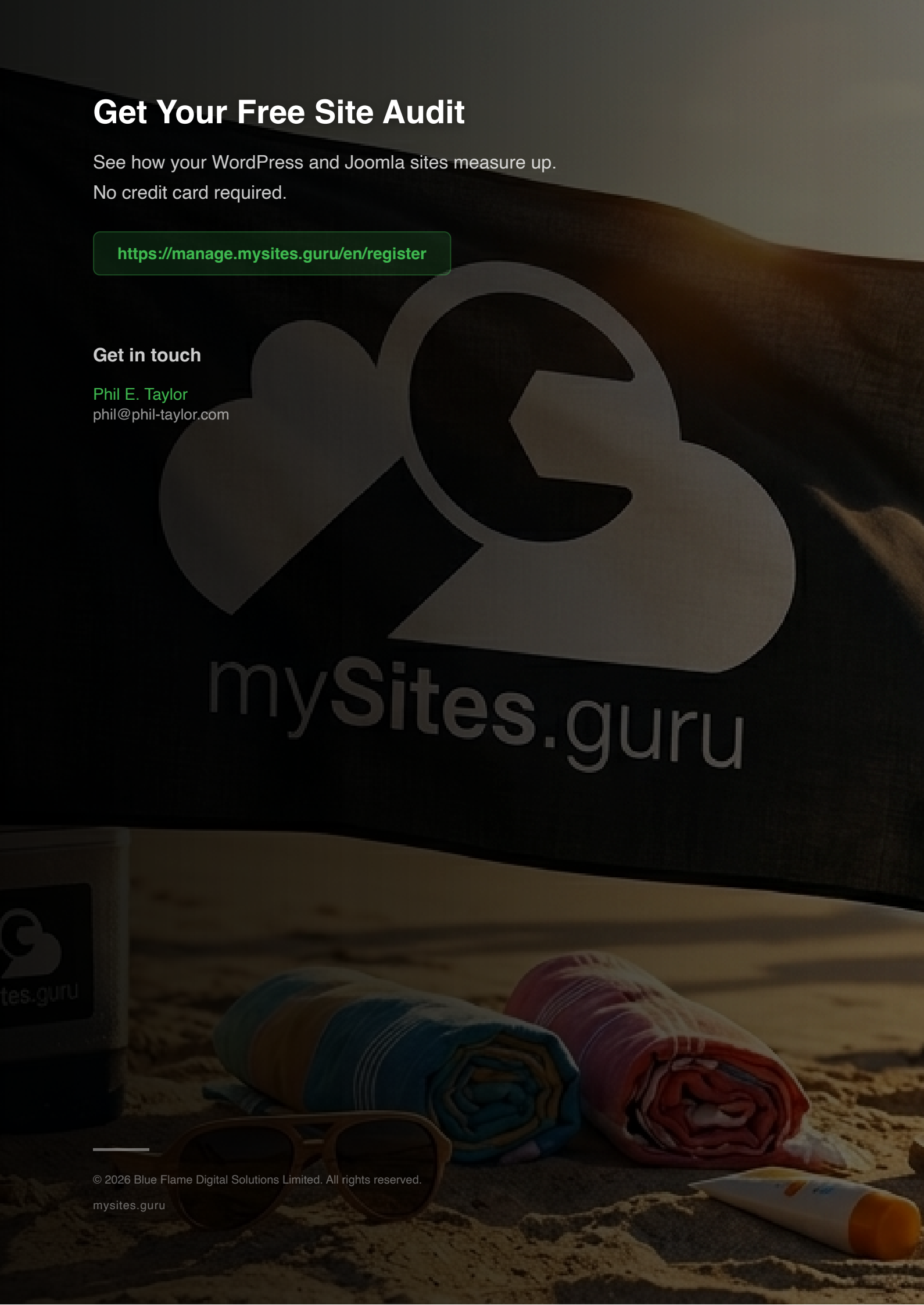
mySites.guru lists every extension on every connected site, so you can search for Sourcerer once and get back each site running it with its version number, rather than logging into every site in turn. Sites on an affected version are flagged automatically against our vulnerability rules.

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