mySites.guru

JoomGallery 4.4.2 Fixes an Unauthenticated File Upload

JoomGallery 4.0.0 to 4.4.1 accept uploads from anyone with no login. Install 4.4.2, not the 4.4.1 the CVE record names, which is affected too.

Phil E. Taylor | 19 September 2026

Install JoomGallery 4.4.2. Every 4.x release before it, from 4.0.0 up to and including 4.4.1, ships an upload service that accepts files from anyone on the internet with no login at all.

That is the short version, and if you only read one line it should be that one, because the published advisory will send you to the wrong release.

6.3
CVSS 4.0

MEDIUM **the Joomla CNA · CVE-2026-84048**

An anonymous visitor can write files into the site's temp directory, and read back or delete anyone else's upload in progress. Neither the file name nor the extension is under their control, which is what keeps this short of straightforward remote code execution on a normal server.

`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:N/VI:L/VA:N`

No login needed | Joomla 4, 5 and 6 | Fixed in 4.4.2

If you manage Joomla sites through mySites.guru, the work is already done for you. The vulnerability rule covering JoomGallery 4.0.0 to 4.4.1 is live, so every connected site running an affected version is flagged on its site card, and searching the mySites.guru extension inventory for JoomGallery gives you every site that has it with the installed version beside each one. We have also emailed the owner of every affected site directly, with their own sites listed.

Install 4.4.2, not the version the advisory names

Here is the trap, and it is a good example of why a machine-readable security record is not the same thing as security advice.

CVE-2026-84048 is titled `Joomla Extension – joomgalleryfriends.net – Unauthenticated arbitrary file upload via the TUS endpoint in JoomGallery <`

`4.4.1`. Read that the way everybody reads a CVE title and you conclude the flaw was fixed in 4.4.1.

The same record's structured affected list says something different:

```
"versions": [ { "status": "affected", "version": "4.0.0-4.4.1" } ]
```

That range includes 4.4.1. The two parts of one record disagree, and the version list is the right one. JoomGallery 4.4.1 was released on 8 September 2026 as an ordinary bugfix release with ten non-security changes in it, and the upload service it ships is unchanged from 4.4.0. The fix arrived ten days later in 4.4.2.

The vendor's own advisory gets this right: affected 4.0.0 up to and including 4.4.1, fixed in 4.4.2. But the CVE record has no remediation field at all, so anything downstream that wants to tell you what to install has to derive it from that title. As of 19 September 2026, one widely-read vulnerability database renders the record as "any release below 4.4.1 is considered affected", adds "no vendor fix or workaround currently provided", and recommends you "upgrade the JoomGallery extension to version 4.4.1 or later" - on the same page whose affected table lists 4.4.1 as vulnerable, the day after 4.4.2 shipped.

Even the vendor's own GitHub release notes for 4.4.2 quote the CVE description verbatim, so the release that contains the fix describes the flaw as being in versions "< 4.4.1".

If you already actioned this from a vulnerability feed, check what you installed

An automated feed or a scanner that derived its advice from the CVE title would have told you to install 4.4.1. That version is affected. Open one of the sites you updated and confirm the JoomGallery version reads 4.4.2 before you consider this closed.

What the JoomGallery upload service actually did

JoomGallery 4.x ships a drag-and-drop multi-upload widget for the gallery, and behind it an implementation of tus, an open protocol for resumable uploads. Resumable means the upload is split across several requests on purpose: one to create the upload and get back a handle, more to append bytes at an offset, another to check how far it got. A dropped connection resumes instead of starting over, which is the right thing to want when someone is uploading five hundred photographs over hotel wifi.

The implementation is a vendored fork of an open-source tus server, wired into the component and dispatched before anything else happens in the controller. In 4.4.1 the entire precondition on creating an upload is a check that the client sent the right protocol version header:

```
case 'POST':
  if(!$this->checkTusVersion())
  {
    throw new Request('The requested protocol version is not supported', 4
  }
  $this->buildUuid();
  $this->processPost();
    break;
```

There is no session token, no permission check and no check of who is asking. Search the whole request path at that release for `checkToken`, `getFormToken`, `authorise`, `checkACL` or `getIdentity` and you get nothing back, which is an unusually clean answer to the question “what was missing”.

What an anonymous visitor could do on a stock Joomla server

This is where the reporting on this flaw splits in two, and each side overstates in the opposite direction. The vendor’s page calls it critical. The record scores it 6.3 Medium. The useful answer is the list of what the endpoint permitted on a default install:

Write arbitrary bytes	Create an upload and append any content to it. The bytes are entirely attacker-chosen; only the file's name is not.
Read back someone else's upload	The server's GET handler streams an upload back to anyone who names its handle, and it is enabled by default in both releases.
Delete someone else's upload	The DELETE handler has the same absence of an ownership check as everything else.
Fill the disk	The 250MB cap was enforced against a header rather than against what was stored, and both guards in the write loop were skipped entirely by a chunked request that sends no Content-Length.

None of that needs an unusual server. It needs JoomGallery 4.3.0 or later installed and enabled.

What the endpoint would not do is give an attacker a file they could run. On creating an upload, JoomGallery generates the filename itself, as 32 hexadecimal characters with no extension and nothing appended from the request. The attacker-supplied `filename` in the upload metadata is only read much later, at the point the gallery finalises an image into the library, and that step runs the component's normal permission and file-type checks that an anonymous visitor cannot pass. So the artefact left on disk is a pair of files in the Joomla temp directory:

```
tmp/aabbccddeeff00112233445566778899
tmp/aabbccddeeff00112233445566778899.info
```

The first holds whatever was uploaded. The second is a small JSON record of the upload's state.

Why code execution needs an unusual server

Both the vendor and the CVE say code execution requires a non-standard server configuration, and having read the code, that is a fair statement rather than a hedge.

A default Joomla `tmp` directory is web-readable and ships only an `index.html`. The bundled `htaccess.txt` has no rules covering it. So one of those files will happily be served to a browser. It will not be executed, because it has no extension, and both Apache with a `.php` handler and nginx with a `location ~ \.php$` block decide what to execute by extension.

Getting one to run needs the server to have been told to execute things by something other than extension: a `SetHandler` bound to a directory rather than a suffix, an nginx location that passes everything to PHP-FPM, a temp path pointed somewhere a second stack executes, or a separate file-inclusion bug elsewhere on the site that will include a path the attacker can name. That last one is the realistic chain, and it needs a second vulnerability to exist first. The attacker does know the exact path, because the server hands the handle back in the response to the request that created the upload.

So: not a webshell drop on a normal LAMP stack, and the 6.3 is honest. Also not nothing, because everything in the table above happens on a completely ordinary install.

The front-end route opened in 4.3.0, not 4.0.0

The CVE and the vendor both date this to the beginning of the 4.x branch. The vendor's wording is that the flaw "has existed since the introduction of JoomGallery 4.x". We read the released packages for 4.0.0, 4.1.1, 4.2.0, 4.3.0, 4.4.1 and 4.4.2 to work out what changed and when, and the picture is slightly more specific than that.

The upload service has been unprotected since 4.0.0, which is what the vendor is describing. But reaching it without logging in is a separate question, and the answer changed partway through the branch:

4.0.0 to 4.2.0

Administrator only

The upload service exists with no access checks, but the only route to it is inside the Joomla administrator, behind the core.manage permission on the component.

4.3.0 to 4.4.1

Reachable by anyone

A site-side controller inherits the upload dispatch for the first time, so the same unprotected service answers requests from the public front end.

Read from the released packages at each tag. We did not exploit a running 4.2.0 install to confirm the negative, so treat the earlier band as the code's account rather than a tested one.

From 4.3.0, released in February 2026, a controller in the site half of the component extends the administrator controller that dispatches the upload service, and Joomla's own access gate does not help, because it only checks permissions when the request is for the administrator application. That gives three front-end routes into it, all reachable without a login.

This distinction is worth having if you are triaging a backlog of Joomla sites rather than updating one. A site sitting on 4.1.1 is in better shape than one on 4.4.1, because the unprotected endpoint there needs an administrator session to reach at all. It still wants updating, and mySites.guru still flags it, both because the vendor and the CNA describe the whole branch as affected and because an endpoint with no token check is reachable through a logged-in administrator's browser whatever else is true.

What JoomGallery 4.4.2 changed

The fix rewrites the upload service rather than patching one check into it, which is the right shape of response. Four things were added.

A session token is now required on every request that writes. The server reads an `X-CSRF-Token` header and compares it against Joomla's form token with `hash_equals`, rejecting anything that does not match with a 403. On its own that stops a third-party

page driving a logged-in administrator's browser, though it does not stop anonymous abuse, because a guest has a session and a valid token too.

Each upload is now owned. On creation the server records an owner derived from the logged-in user id, or from a hash of the session id for a guest, and every later request comparing that owner with `hash_equals` before it will touch the upload. That is what stops one visitor reading, resuming or deleting another visitor's upload.

Every upload must now name a target the requester is allowed to write to. A new authorisation step loads the gallery category or image being uploaded into and runs the component's own permission check against it, and rejects the request when no target is given. This is the control that actually shuts the anonymous attacker out, because an anonymous visitor has no permission to add an image to a category.

The size limit is now enforced against what is being stored rather than against a header the client supplied, and the write loop's bound no longer depends on a `Content-Length` that a chunked request simply omits.

One change in the same release that is easy to misread

The upload handle also changed, from an MD5 of `uniqid()` seeded with `mt_rand()` and the machine name, to `bin2hex(random_bytes(16))`. That looks like a weak-randomness fix and it is worth understanding as the opposite. In 4.4.1 the handle protected nothing, because anyone could create uploads freely and no request checked ownership, so a perfectly random value would have changed nothing. The patch gives the handle a security job for the first time, by pairing it with the owner check, and a predictable value would have been a real weakness under the new design. The fix is correct for what the code became, not a repair of what it was.

Three CVEs in one Joomla extension branch in under a month

This is the third CVE against the JoomGallery 4.x branch in four weeks, and the second of the three classified as improper access control.

CVE	Published	What it was	Fixed in
CVE-2026-66916	22 Aug 2026	Password-protected categories readable through the JSON view, which skipped the gate the HTML view enforced	4.4.0
CVE-2026-66917	22 Aug 2026	Stored cross-site scripting through image records, executing for every later visitor	4.4.0
CVE-2026-84048	15 Sep 2026	The unauthenticated upload service described here	4.4.2

Two different external researchers found them, which is a reasonable sign that people are now looking at this extension properly. If you updated in August and considered the matter closed, note that 4.4.0 fixed the first pair and is affected by this one.

The gap that stands out in the sequence is the three days between CVE-2026-84048 being published on 15 September and 4.4.2 shipping on 18 September. For those three days the flaw was public, complete with the endpoint's name, and no fixed release existed. That is not the vendor being slow, six days from report to release is a good turnaround for a volunteer project, but it is a reminder that a CVE appearing in your feed does not mean a patch is waiting for you.

How to check the Joomla sites you manage

By hand, the loop per site is: log in, open the extension manager, filter for JoomGallery, read the version, update if it is below 4.4.2, then open the temp directory over FTP or SSH and look for the file pair. On one site that is five minutes. On forty it is most of a day, and it is the kind of day where site thirty-one gets a less careful look than site three.

The version part of that is what an extension inventory is for. In mySites.guru you search the inventory once for JoomGallery and get back every connected site with it installed and the version on each, and any site below 4.4.2 is flagged as vulnerable on its own site card without you searching for anything at all.

The temp directory part is the check below, which looks for exactly the kind of leftover this flaw produces.

One caveat worth knowing about, because it affects a small number of sites. A handful of JoomGallery installs report an empty version string in their manifest. A version that cannot be ordered against a range cannot be judged against one, so mySites.guru declines to flag those sites rather than guessing, which is the right way round but leaves you to check them yourself. If such a site runs Joomla 4, 5 or 6 then it is on the 4.x branch by definition, because 3.x does not run there, and it needs a manual look. The alert emails we sent name those sites explicitly.

What the numbers look like across the sites we manage

Two days after 4.4.2 shipped, 94% of the JoomGallery 4.x installs we can see across the Joomla sites on mySites.guru are still on an affected version. That is not a criticism of anyone: a release from two days ago that was announced on a project news page and a GitHub tag has not reached most people yet, which is the entire reason we send the emails.

The larger population is more interesting. The clear majority of JoomGallery installs we can see are still on the 3.x branch for Joomla 3, and those are not affected by this flaw,

because 3.x has no resumable-upload server in it at all. Those sites have a different and larger problem, which is that they are on Joomla 3.

Updating extensions across a lot of sites

The honest lesson of this one is about the distance between a fix existing and a fix being installed, which for most agencies is measured in weeks and for some sites is measured in never. JoomGallery is incidental to it.

mySites.guru exists to close that distance. It tracks the installed version of every extension on every connected Joomla and WordPress site, matches those against a curated vulnerability database for Joomla extensions and the Wordfence feed for WordPress, and flags affected sites without anyone reading a mailing list. You can update extensions across many sites from one screen instead of logging into each, and set safe automatic updates for the extensions you trust to update themselves.

Some related reading, if this post found you because a site is already in trouble rather than because you are keeping ahead of it:

- Finding hacked files and backdoors, which is where to start if you found something in that temp directory you cannot account for
- Why your .htaccess will not stop a Joomla hack, which covers this exact class of problem and why server hardening is not a substitute for patching
- AJAX endpoints and the CMS security blind spot, the root-cause pattern this flaw is one more instance of
- How to clean up dangerous files on a Joomla server, including what accumulates in `tmp` and why Joomla does not always clear it

See the full pricing Unlimited sites on every plan, so a JoomGallery sweep across forty Joomla sites costs the same as across four.

If you would rather see it work before deciding, connect one site and run a free audit on it. It will tell you whether that site is running an affected JoomGallery, and a good deal else besides.

And if you have already found something you cannot explain in a temp directory, that is a different conversation. fix.mySites.guru is a flat £120 per incident: the site gets cleaned, audited for backdoors and handed back, usually the same day, and we screen it first so you are not charged in the rare case it cannot be fixed.

Timeline

2026-02-08 — **JoomGallery 4.3.0 opens the upload service to the front end**

A site-side controller inherits the resumable-upload dispatch for the first time. Before this release the same endpoint exists but only inside the Joomla administrator.

2026-09-12 — **Yugorin of Samar Group reports the flaw to the vendor**

The vendor's advisory credits Yugorin of Samar Group with a responsible disclosure on this date. The fix branch opens the same day.

2026-09-15 — **CVE-2026-84048 is published with no patch available**

The Joomla CNA publishes the record, scored CVSS 4.0 6.3 Medium. The title names 4.4.1 as the boundary. No fixed release exists yet, and none would for another three days.

2026-09-18 — **JoomGallery 4.4.2 ships the fix**

The release adds a token check, an ownership check and per-category permission checks to the upload service, and enforces the upload size limit it previously ignored.

2026-09-19 — **mySites.guru flags every affected site and emails the owners**

The vulnerability rule covering JoomGallery 4.0.0 to 4.4.1 goes live, and the owner of every connected site running an affected version gets a direct alert listing their sites.

Further Reading

- JoomGallery 4.4.2 Safety Release - the vendor's own advisory, which states the affected range and the fixed version correctly.
- JoomGallery 4.4.2 release notes on GitHub - the changelog, including the pull request that rewrote the upload service.
- CWE-284: Improper Access Control - the weakness class the Joomla CNA assigned to this flaw.
- The tus resumable upload protocol - the open protocol JoomGallery implements, and the reason the endpoint accepts a partial file across several requests.
- OWASP File Upload Cheat Sheet - the developer's checklist for accepting uploads safely, including why an upload endpoint needs its own access check.
- A month of Joomla security disclosures - the wider run of Joomla extension flaws this sits alongside.

Frequently Asked Questions

Which JoomGallery versions are affected by CVE-2026-84048?

Every JoomGallery 4.x release from 4.0.0 up to and including 4.4.1. The fix is in 4.4.2, released on 18 September 2026. Be careful here: the CVE record is titled "JoomGallery < 4.4.1", which reads as though 4.4.1 contains the fix. It does not. The same record's structured version list says 4.0.0 to 4.4.1 are affected, and the vendor's own advisory says 4.4.1 is affected and 4.4.2 is the fix. Install 4.4.2. The JoomGallery 3.x branch for Joomla 3 has no resumable-upload server and is not affected by this flaw.

What could an attacker actually do with it?

On a default Joomla server, an anonymous visitor could write arbitrary bytes into the site's temp directory, read back any upload still in progress, delete one, and keep writing until the disk filled. They could not choose the file name or give the file an extension, because JoomGallery generates the name itself and never appends anything the client sends. That is why the Joomla CNA scored it 6.3 Medium rather than a straight remote code execution. It becomes code execution only on a server configured to run files that have no extension, which is unusual but not unheard of.

Where do the uploaded files end up?

In whatever directory Joomla's Global Configuration has set as the path to the temp folder, which on a default install is the tmp directory in the site root. Each upload creates two files: one named as 32 hexadecimal characters with no extension, holding the uploaded bytes, and a sibling of the same name ending .info holding a small JSON record of the upload. If you find a pair like that and you did not put it there, delete both and then check the rest of the site properly.

Does this affect Joomla itself?

No. JoomGallery is a third-party gallery extension, not part of Joomla core, and this flaw is in the extension's own upload service. A Joomla site without JoomGallery installed is not affected by CVE-2026-84048. The Joomla CNA assigns CVE identifiers for third-party Joomla extensions as well as for core, so a Joomla-assigned CVE does not by itself mean a core flaw.

How do I find every JoomGallery site I manage?

By hand, you would log in to each Joomla site in turn and read its installed extensions list. With mySites.guru you search the extension inventory for JoomGallery once and get every

connected site running it, with the installed version next to each one, on a single screen. mySites.guru also flags any site running JoomGallery 4.0.0 to 4.4.1 as vulnerable automatically, and the flag clears itself once the site reports 4.4.2.

My JoomGallery version shows as blank or unreadable. What does that mean?

Some Joomla extensions report an empty or placeholder version string in their manifest, and JoomGallery is one where we see this. A version that cannot be ordered against a range cannot be judged against one either, so mySites.guru will not flag such a site as vulnerable on no evidence. If your site runs Joomla 4, 5 or 6 and has JoomGallery installed, it is running the 4.x branch, because 3.x does not run on those versions. Open the Joomla extension manager on that site and read the version yourself.

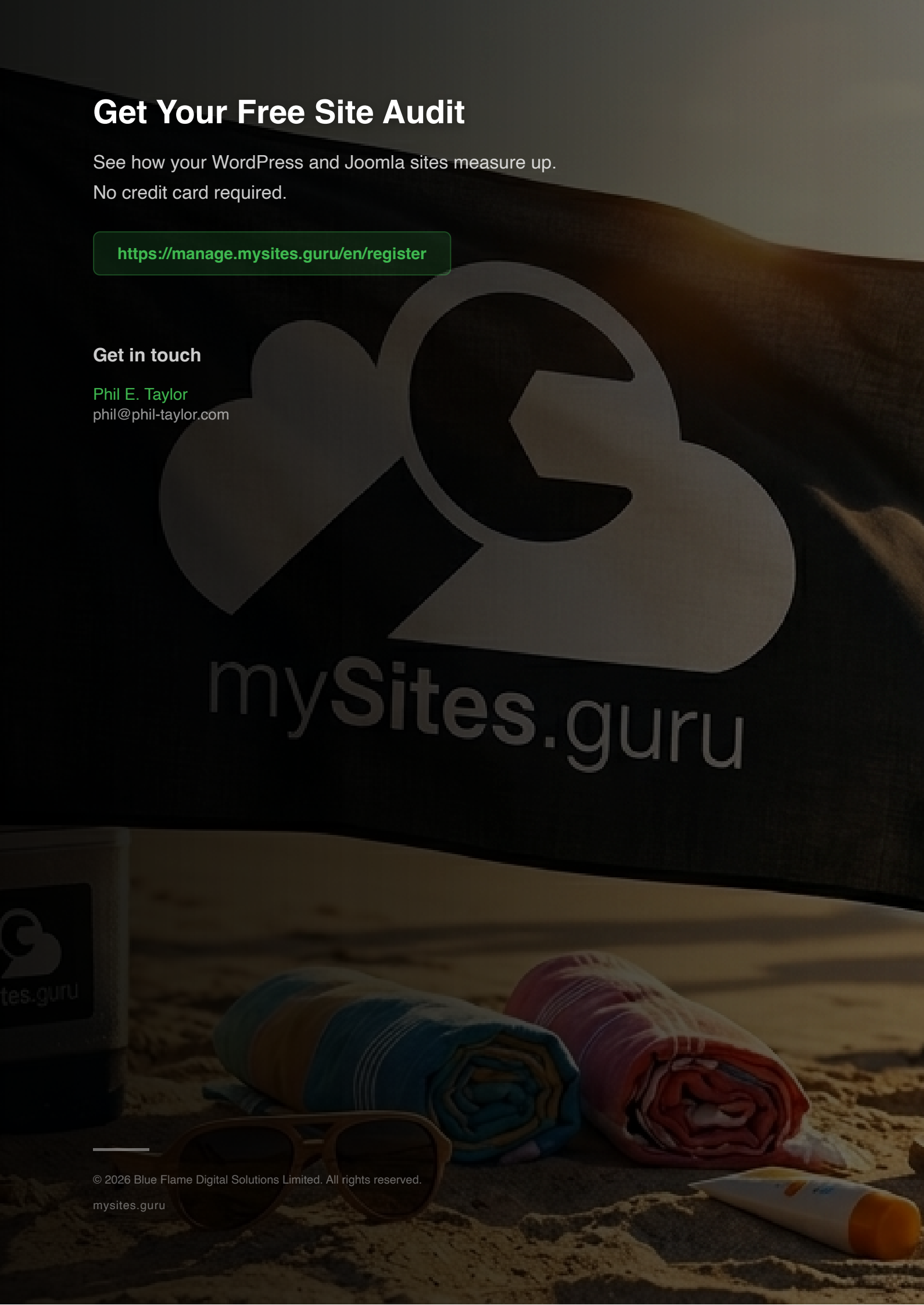
Written and edited by a human, with AI assistance. Our approach to AI

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