mySites.guru

jDownloads 4.1 Shipped an Unauthenticated Upload Endpoint, Now Fixed in 4.1.6

jDownloads 4.1.0 to 4.1.5 shipped a leftover test script that let anyone upload files to your Joomla site with no login. Update to 4.1.6, which removes it.

Phil E. Taylor | 26 August 2026

jDownloads is a long-established download-manager component for Joomla, installed as `com_jdownloads` and used to store, organise and serve files on business, membership and community sites. It has been on the Joomla Extensions Directory since 2007. Every 4.1.x release, from 4.1.0 to 4.1.5, shipped a file it should not: a standalone upload script with no login check, no CSRF token and no permission check, sitting at `administrator/components/com_jdownloads/assets/upload/upload-handler.php`. Anyone on the internet who knew the address could use it to upload files to your site.

jDownloads has now released 4.1.6 as a security update on 18 July 2026. It removes the leftover files from the package and clears any legacy files and `test_uploads` directory left behind by an earlier 4.1.x version when a site installs it. Update every affected site to 4.1.6. This post explains what the file did, which versions carried it, how serious it really was, and how to make sure it is gone, without publishing anything an attacker could copy.

> **Update to jDownloads 4.1.6**
>
> jDownloads 4.1.6 is the fix: it removes administrator/components/com_jdownloads/assets/upload/upload-handler.php and clears the leftover test_uploads directory on install. Update every jDownloads 4.1 site you run. If you cannot update immediately, delete that file by hand as a stopgap; it does not change how jDownloads works.

TL;DR

TL;DR: jDownloads 4.1.0 to 4.1.5 ship `administrator/components/com_jdownloads/assets/upload/upload-handler.php`, a standalone `move_uploaded_file` endpoint with no `_JEXEC` guard, no Joomla session, no CSRF token and no permission check. It ignores jDownloads' own upload blacklist and enforces only its own hardcoded allow-list, which excludes `.php` but permits archives, images, documents and executables like `.exe` and `.msi`. An unauthenticated visitor could write files of those types into a folder on your site and

read them back over HTTP. The file was introduced in 4.1.0 and is byte-identical across every 4.1.x build (MD5 `d5bbd57fd97c99890f6a5fdc9d79ec00`); it is absent from 4.0.52 and earlier. The developer confirmed it was a test component included by mistake, and jDownloads 4.1.6 (released 18 July 2026) removes it. It is tracked as CVE-2026-61900. Update every affected site to 4.1.6, and mySites.guru flags every jDownloads 4.1.0 to 4.1.5 site as vulnerable automatically so you can find them.

What Is Actually Wrong in jDownloads 4.1?

The problem is a single file that should never have shipped. `upload-handler.php` is a self-contained PHP script that handles a file upload on its own, outside Joomla's normal request flow. It has none of the checks a Joomla upload is supposed to have:

- No `_JEXEC` guard, so it runs when requested directly by URL rather than only through Joomla.
- No Joomla session or login check, so it does not care who you are.
- No CSRF token, so there is nothing tying the request to a real logged-in user.
- No `authorise()` permission check, so no capability is required.

It also ignores jDownloads' own configuration. The component has a proper upload area in the Joomla admin with a blacklist that blocks `.php` and other dangerous types. This script does not use any of that. It enforces its own hardcoded list of allowed extensions, and that list, while it excludes `.php`, includes archives, images, office documents, and executables such as `.exe` and `.msi`. Whatever an anonymous visitor sends, if the extension is on that list, the file is written to a folder on your site under a timestamped name.

We reproduced this on a local Joomla 5 install to be certain of the behaviour. As an anonymous request, with no cookie and no token, an allowed file uploaded successfully and was then readable back over HTTP. A direct `.php` upload was rejected by the script's own list. We are not publishing the request an attacker would send, the parameters, or a working proof of concept. The shape of the bug is enough to understand the risk and to remove the file.

Which jDownloads Versions Are Affected?

Every 4.1.x release has the file: 4.1.0, 4.1.1, 4.1.2, 4.1.3, 4.1.4 and 4.1.5. It is byte-for-byte identical in all of them, with the MD5 hash `d5bbd57fd97c99890f6a5fdc9d79ec00`, so you can verify a suspect file on any site by hashing it:

```
md5sum administrator/components/com_jdownloads/assets/upload/upload-handle
```

We pinned the version range by comparing the official jDownloads packages directly. The file first appears in 4.1.0 and is not present in 4.1.0's predecessor, 4.0.52, or any earlier release, and it is removed again in the patched 4.1.6. In practice that means the 4.0 line and the older 3.x line, which are still the most common jDownloads versions we see across the sites we manage, do not carry this particular file. It is the 4.1.0 to 4.1.5 sites that are exposed. If your sites are on any of those, they have the file, whether or not you use uploads, so update them to 4.1.6.

> **Note**
> A matching hash is a definite hit, but the absence of the exact hash does not clear a site: if you find a file at that path on a 4.1.x install, delete it regardless of its hash. The hash is a convenience for confirming the shipped file, not a scanner in its own right.

How Was This Found, and How Was It Fixed?

This one came to us through a mySites.guru subscriber. They found the flaw and reported it privately to the jDownloads developer, Arno Betz, who confirmed on 16 July 2026 that the file was experimental upload code left in the release package by mistake and said a fix was being prepared. jDownloads then released 4.1.6 on 18 July 2026, which removes the handler and clears the leftover `test_uploads` directory when a site installs the update.

We obtained the vendor's confirmation and then reproduced the flaw on a local Joomla install, so that we could be precise about what was exposed and which versions carried

the file before flagging it to the sites we manage. It is now tracked as CVE-2026-61900. jDownloads' own advisory thanks the community for reporting security issues responsibly but does not name the file or credit anyone by name, so if you only read the changelog you might miss what 4.1.6 actually closes.

> **Why we are not publishing an exploit**
>
> The developer acknowledged the flaw and has shipped a fix in 4.1.6. We have withheld the exact request, the parameters and any working proof of concept. The point of this post is to get the file off live sites, not to hand anyone a tool. The file path is included only because you need it to check for the file and delete it if you cannot update immediately.

How Serious Is It Really?

It is easy to file every "unauthenticated file upload" under the same maximum-severity panic, so it is worth being precise. This is a real, no-login flaw, but it is not a clean remote code execution on a default server, and saying otherwise would be dishonest.

The reason is the script's own extension list. It blocks a bare `.php`, so an attacker cannot simply upload a web shell and run it on a standard setup. What they can do without any login is real enough:

- Host malware, phishing pages, or other unwanted files on your domain. A trusted domain serving attacker content gets flagged by browsers and search engines, which drags down your reputation and your search rankings even if no code ever runs.
- Fill your disk. The script accepts large files, so it is a straightforward abuse and denial-of-service primitive.

It crosses into remote code execution in two situations. The first is a server that executes secondary file extensions, an Apache `AddHandler` or `mod_mime` misconfiguration where a name like `shell.php.jpg` is run as PHP even though it "looks like" an image. That is common enough on cheap shared hosting. The second is a site where an administrator has widened the allowed types to include an executable one. In

both cases an anonymous visitor gets to run code on your server. We assess the flaw as High severity for that reason: unauthenticated and reachable by anyone, gated on the default stack only by an extension list that several common configurations undo.

How Do I Fix My jDownloads Sites?

The proper fix is to update to 4.1.6. If you cannot do that on a site straight away, remove the file by hand as a stopgap. Either way it takes a couple of minutes per site.

1. **Update to 4.1.6.** Update jDownloads to 4.1.6 on every affected site through Joomla's own Updates screen or by installing the package. It removes `upload-handler.php` from the component and clears the leftover `test_uploads` directory on install, so getting onto the patched release is the right end state.

2. **Or delete the upload script by hand.** If you cannot update a site immediately, use FTP, SSH or your host's file manager to delete `administrator/components/com_jdownloads/assets/upload/upload-handler.php`. That one file is the whole problem. The `default.php` sitting next to it is part of the same leftover test code and can go too. Over SSH:

   ```
   rm administrator/components/com_jdownloads/assets/upload/upload-handle
   ```

3. **Clear the test_uploads folder.** Look for `administrator/components/com_jdownloads/test_uploads`. If it exists, that is where the script writes, so review it for anything you did not put there, especially files that are not obviously your own, and then delete the folder.

4. **Check for signs it was used.** If a site ran 4.1.0 to 4.1.5 with the file in place for a while, do not just assume it is clean. Look for unexpected files in web-served folders, unfamiliar administrator accounts in your Joomla Users list sorted by registration date, and recently modified PHP files across the site.

If you cannot get to a site immediately, you can also block the path at the web server as a stopgap, denying access to `upload-handler.php`, but updating to 4.1.6 is cleaner

and permanent.

How mySites.guru Flags This Across Every Site

If you run sites through mySites.guru, two things are already working for you. First, mySites.guru now has a rule that flags any connected Joomla site running jDownloads 4.1.0 to 4.1.5 as vulnerable, so the affected sites surface on their own rather than waiting for you to go looking. Second, the extension inventory lets you search for jDownloads once and see every connected site running it and the version each is on, so you know your exposure in seconds instead of logging into forty admin panels.

The monitoring also watches for the outcome, not just the extension. An unexpected file appearing in a public folder, or an unfamiliar administrator account, is caught by the suspect content and hacked-file detection regardless of which extension let it in. That is the same generic detection that catches Page Builder CK and RSFiles upload attempts, and it is why a new bug of this shape does not need a bespoke signature before it is visible.

Find every jDownloads install across your sites

Open your Extension Inventory

Search for jDownloads across every connected Joomla site and filter for anything on 4.1.0 to 4.1.5 to find the installs that need updating to 4.1.6. Not a subscriber? Sign up free and connect your sites.

File Uploads Are a Recurring Joomla Attack Surface

If you manage Joomla sites, treat every extension that touches file uploads as part of your attack surface. The underlying weakness here, a file with a dangerous type written where a check should have stopped it, is CWE-434, and it turns up across the whole category rather than in one vendor's code.

The recent run of Joomla disclosures makes the point. Balbooa Forms, iCagenda, Page Builder CK, RSFiles, the SP Page Builder zero-day and DJ-Classifieds were all variations on the same theme: an endpoint that accepted a file without a strict check on what actually got written. jDownloads' version is a little different, because the dangerous file is not part of the working feature at all, just a test script that was never meant to reach a live site. The lesson is the same one behind the AJAX and frontend endpoints blind spot: anything reachable over HTTP has to assume the request is hostile, and a stray script with no checks is reachable the moment it is installed.

To the developer's credit, jDownloads has a short security history and responded to the private report rather than ignoring it, shipping 4.1.6 within days. This is one file left in by mistake, not a pattern of problems, and the fix removes it properly.

Stay Ahead of the Next One

This is one extension on one day, and there will be another, because Joomla runs on thousands of third-party extensions and the ones that handle files keep producing bugs like this. The hard part is never the removal itself. It is knowing the problem exists, knowing which of your sites are affected, and getting to them before someone else does.

That is the job mySites.guru does for you. It keeps a live inventory of every extension on every Joomla and WordPress site in your account, flags the ones with a known problem, jDownloads 4.1.0 to 4.1.5 included, and lets you act across all of them from one screen. When something like this surfaces before there is a CVE or a clear public warning, you still see exactly which of your sites run it.

Get free email alerts when a Joomla vulnerability breaks

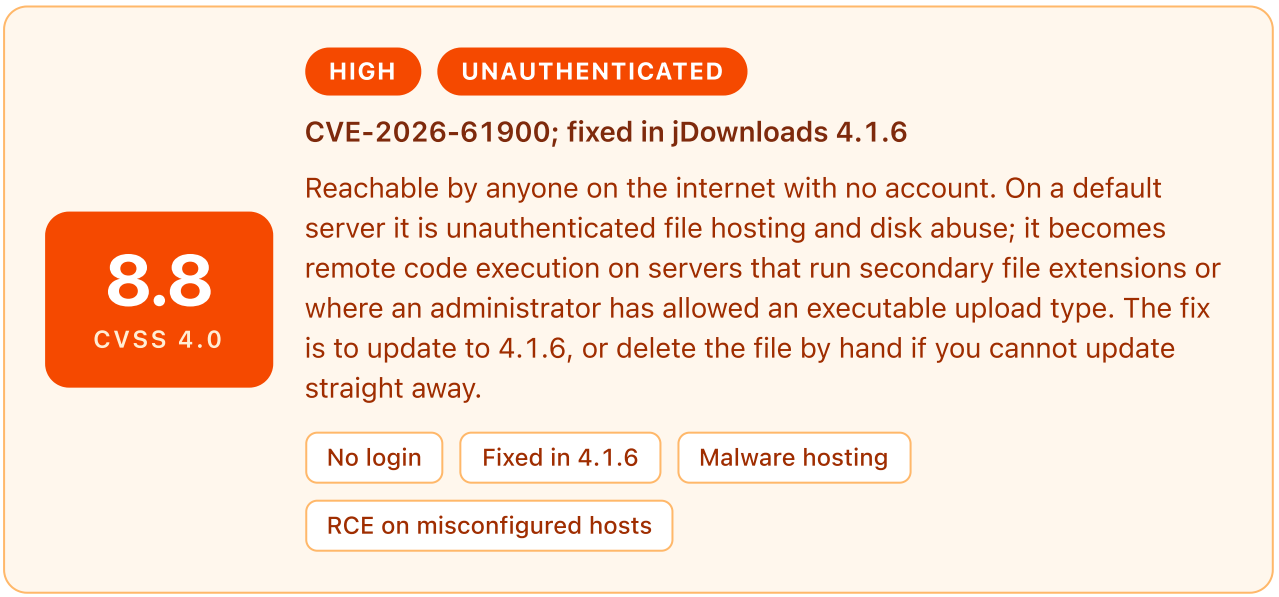
We email a plain-English alert the moment a serious flaw like this one is found, with the affected versions and what to do. No charge, unsubscribe any time.

Subscribe to security alerts

Want the alerts and the tooling to act on them? Start with a free audit on one site and see your full extension inventory, or sign up for mySites.guru to get vulnerability flags and one-click updates across every site you manage. If a site has already been touched, fix.mysites.guru is the done-for-you option.

Disclosure and Severity

This flaw is CWE-434, unrestricted upload of a file with a dangerous type. It is reachable over the network by an anonymous user, with no login and no user interaction, on any site running jDownloads 4.1.0 to 4.1.5. On a default server the script's extension list keeps it short of direct code execution, so the immediate impact is unauthenticated file hosting and disk abuse; it reaches remote code execution on servers that execute secondary extensions or where an administrator has allowed an executable type. We assess it as High severity, CVSS 4.0 8.8 (`AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`), scoring the guaranteed arbitrary file write rather than the host-dependent code execution. It is tracked as CVE-2026-61900. jDownloads published a security advisory and released the patched 4.1.6 on 18 July 2026.



Field	Detail
Component	jDownloads (`com_jdownloads`)

Field	Detail
Vendor	Arno Betz, jdownloads.com
Type	Unauthenticated arbitrary file upload (leftover test script)
File	`administrator/components/com_jdownloads/assets/upload/upload-handler.p` (MD5 `d5bbd57fd97c99890f6a5fdc9d79ec00`)
CWE	CWE-434 (Unrestricted Upload of File with Dangerous Type)
CVE	CVE-2026-61900
Severity	High (our assessment)
CVSS 4.0	8.8 (High), `AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N` (our own assessment)
Reported by	A mySites.guru subscriber, privately, to the vendor
Affected versions	4.1.0 to 4.1.5 (file introduced in 4.1.0; absent from 4.0.52 and earlier)
Fixed in	jDownloads 4.1.6, released 18 July 2026

The full timeline is at the foot of this post.

Timeline

2026-07-16

The vendor confirms the upload handler was shipped by mistake

Arno Betz confirms privately that upload-handler.php was experimental upload code included in the jDownloads release package in error, and says a fix removing the handler and the test_uploads directory is being prepared.

2026-07-17

We reproduce it, flag the affected range and publish

The flaw is reproduced on a local Joomla install and the range confirmed by comparing official packages. Sites running 4.1.0 to 4.1.5

are flagged. No proof of concept is released.

2026-07-18 ● **jDownloads 4.1.6 removes the leftover files**

A security update that strips the leftover upload files from the package and clears the test_uploads directory on install.

Further Reading

- jDownloads 4.1.6 Secure Update Released - the vendor's own advisory for the patched release, with its notes on the removed legacy files and cleanup on install.
- CWE-434: Unrestricted Upload of File with Dangerous Type - the canonical definition of this weakness class from MITRE.
- OWASP Unrestricted File Upload - why file upload flaws rate so highly and how far they can go.
- OWASP File Upload Cheat Sheet - the developer's checklist for accepting uploads safely, including why storing uploads outside the web root matters.
- Securing Joomla extensions - Joomla's own guidance for developers on access checks and safe file handling.
- A month of Joomla security disclosures - the wider run of Joomla extension flaws this sits alongside.

Frequently Asked Questions

What is the jDownloads upload-handler.php flaw?

jDownloads 4.1.0 to 4.1.5 ship a standalone PHP script at administrator/components/com_jdownloads/assets/upload/upload-handler.php that accepts file uploads with no login, no CSRF token and no permission check. It ignores jDownloads' own upload blacklist and enforces only its own hardcoded list of allowed extensions. Anyone on the internet can use it to write files of the allowed types into a folder on your site. The developer has confirmed it was experimental code left in the release package by mistake.

Which jDownloads versions are affected?

Every 4.1.x release: 4.1.0, 4.1.1, 4.1.2, 4.1.3, 4.1.4 and 4.1.5. The file was introduced in 4.1.0 and is byte-for-byte identical across all of them (MD5 d5bbd57fd97c99890f6a5fdc9d79ec00). It is not present in 4.0.52 or any earlier release, which we confirmed by comparing the official packages. If you are on the 4.0 line or older, this specific file is not on your site.

Is there a patched version of jDownloads yet?

Yes. jDownloads 4.1.6 was released on 18 July 2026 as a security update. It removes the leftover files from the package and, on install, clears out any legacy files and test_uploads directory left behind by an earlier 4.1.x version. Update every affected site to 4.1.6. If you cannot update straight away, delete the file yourself as a stopgap, which does not affect how jDownloads works for you or your visitors.

Can this flaw be used to take over my Joomla site?

The script's own list of allowed extensions blocks a direct .php upload, so on a default server it is an unauthenticated way to host malware, phishing pages or unwanted files on your domain, plus disk abuse. It becomes remote code execution on servers that run secondary file extensions (an Apache AddHandler or mod_mime misconfiguration), or if an administrator has widened the allowed types to include an executable one. Either way it should not be reachable by anonymous visitors, so remove it.

How do I delete the file if I cannot update straight away?

Updating to 4.1.6 is the proper fix, but if you cannot do it immediately you can remove the file by hand. Using FTP, SSH or your host's file manager, delete administrator/components/com_jdownloads/assets/upload/upload-handler.php on each affected site. The sibling default.php in the same folder is part of the same leftover test

code and can go too. Then check for an administrator/components/com_jdownloads/test_uploads folder and, if it exists, review it for anything you did not put there before deleting it.

How do I find every jDownloads site I manage?

Manually you would log in to each Joomla site and check its installed extensions. With mySites.guru you search your extension inventory for jDownloads once and get every connected site running it, with the installed version on each, on one screen. mySites.guru also now flags any site running jDownloads 4.1.0 to 4.1.5 as vulnerable automatically.

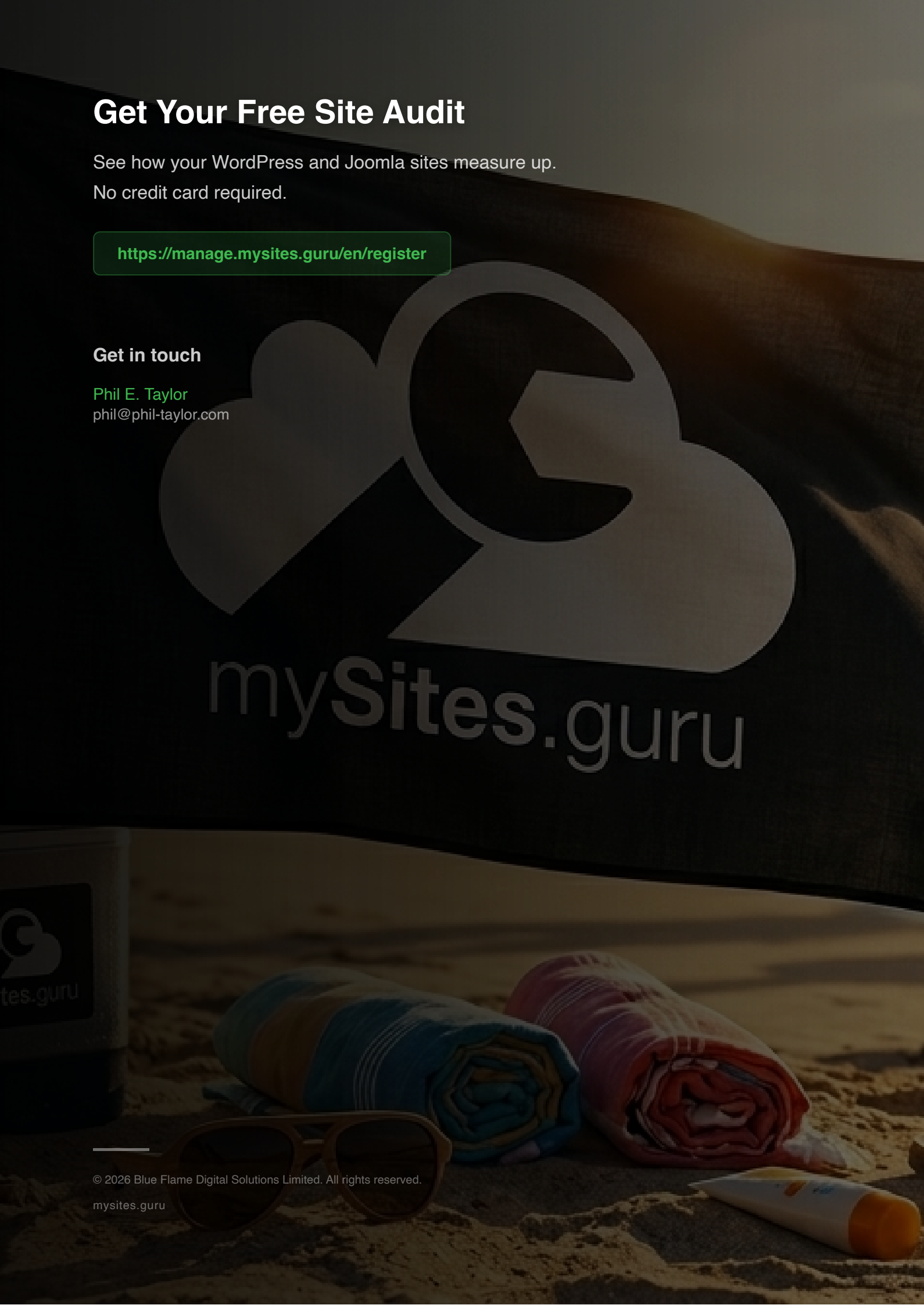
Written and edited by a human, with AI assistance. Our approach to AI

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

© 2026 Blue Flame Digital Solutions Limited. All rights reserved.